A DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

by

Allen G. Greenwood

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Business Administration (Management Science)

APPROVED:

L. J. Moore, Chairman	L. P. Rees
E. R Clayton	M. E. Ridenour
E. C. Houck	B. W. Taylor, III

June, 1984

Blacksburg, Virginia

A DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

by

Allen G. Greenwood

(ABSTRACT)

Tuition and fees are a major source of income for colleges and universities and a major portion of the cost of a student's education. The university administration's task of making sound and effective tuition and fee policy decisions is becoming both more critical and more complex. This is a result of the increased reliance on student-generated tuition-and-fee income, the declining college-age student population, reductions in state and Federal funds, and escalating costs of operation.

The comprehensive computerized decision support system (DSS) developed in this research enhances the administration's planning, decision-making, and policy-setting processes. It integrates data and reports with modeling and analysis in order to provide a systematic means for analyzing tuition and fee problems, at a detailed and sophisticated level, without the user having to be an expert in management science techniques or computers.

The DSS with its imbedded multi-year goal programming (GP) model allocates the university's revenue requirements

to charges for individual student categories based on a set of user-defined objectives, constraints, and priorities. The system translates the mathematical programming model into a valuable decision-making aid by making it directly and readily accessible to the administration. The arduous tasks of model formulation and solution, the calculation of the model's parameter values, and the generation of a series of reports to document the results are performed by the system; whereas, the user is responsible for defining the problem framework, selecting the goals, setting the targets, establishing the priority structure, and assessing the solution.

The DSS architecture is defined in terms of three highly integrated subsystems - dialog, data, and models - that provide the following functions: user/system interface, program integration, process control, data storage and handling, mathematical, statistical, and financial computations, as well as display, memory aid, and report generation. The software was developed using four programming languages/systems: EXEC 2, FORTRAN, IFPS, and LINDO.

While the system was developed, tested, and implemented at Virginia Polytechnic Institute and State University, the concepts developed in this research are general enough to be applied to any public institution of higher education.

DEDICATION

To Jane - for her understanding and tolerance, encouragement and support, sacrifices and patience, and most of all her love.

ACKNOWLEDGEMENTS

I am most grateful to my chairman and mentor, Dr. Laurence J. Moore, whose help, encouragement, guidance, and support throughout my study at Virginia Tech has been greatly appreciated and valued. He has made my pursuit of the Ph.D. a rewarding, worthwhile, and enjoyable experience. I would also like to thank Dr. Moore for advising me to transfer into the Management Science program and for introducing me to the areas of financial planning and decision support systems.

I would like to thank Mr. Minnis E. Ridenour for the financial support for this dissertation, access to the data, as well as his review of the model and overall interest in the research.

I would also like to thank Dr. Bernard W. Taylor, III for first introducing me to the area of goal programming and for his support and encouragement throughout the doctoral program.

In addition, I would like to express my appreciation to Dr. Edward R. Clayton, Dr. Ernest C. Houck, and Dr. Loren P. Rees for their continued interest, help, and encouragement through both the cousework phase and the dissertation phase of my doctoral program.

Also, I would like to express my appreciation to Mrs. Katherine M. Johnston and Mr. Wayland H. Winstead for their assistance and support in all phases of this research, e.g. defining the problem, explaining the process, providing data, demonstrating the model, and implementing the system.

I wish to thank Gerry Chenault and Jane for their help in preparing this document.

TABLE OF CONTENTS

DEDICATION iv

ACKNOWLEDGEMENTS v

Chapter page

I. INTRODUCTION 1

Statement of the Problem 5
Purpose and Objectives 6
Purpose 7
Objectives 8
Methodology Overview 9
Modeling the tuition and fee process via goal programming 9
DSS as a means to translate the GP model into a valuable decision-making aid 11
The DSS architecture 14
Integration of the DSS into the policy-setting and decision-making processes . 15
Scope and Limitations 18
Review of Related Literature 19
The Process of Establishing Tuition and Fee Charges 20
Decision Support Systems in Academic Administration 23
The Use of GP Models for Resource Allocation Planning in Academic Administration . . 25
Plan of Presentation 30

II. THE TUITION AND FEE PROCESS 32

Environmental Factors 32
Demographic Changes 32
Fiscal Pressures 37
Political Influences 38
Evolution of the tuition and fee recovery rate 41
Impact of the Environmental Factors 45
Procedural Steps of the Process 47
Budget activities 48
Recovery rate activities 50

Establishing the tuition and fee charge structure 51
Student Charges 54
General Tuition 57
Student Fees 61
Dorm and Dining Charges 62
Summary 62

III. THE GP MODEL COMPONENT OF THE DSS 64

Modeling the Tuition and Fee Allocation Process 64
The Decision Variables 68
Mathematical Formulation of the Goals 72
Aggregate Goals 74
State-level tuition revenue requirement . 74
Supplemental funds revenue requirement (Yearly) 75
Supplemental funds revenue requirement (Overall) 76
Individual Goals 77
Tuition charges as a percentage of average total cost 77
Change in tuition charges over time . . . 78
University's position in relation to peer institutions 79
General individual goals 81
Relative Goals 82
Absolute Goals 83
Single-category absolute goals 83
Pairwise comparison absolute goals 85
Four-category absolute goals 86
The Priority Structure and Achievement Function 86
Priority Structure 87
Achievement Function 88
GP Model Solution Methodology 89
Sequential Linear Goal Programming 89
Integration of the Solution Routine into the DSS 90
The GP Solution Algorithm 94
Inputs to the Subroutine USER 95
Translation of the priority structure to the achievement function 100
The Solution routine 105
Output and reports 109
Summary 115

IV. THE OVERALL DECISION SUPPORT SYSTEM ARCHITECTURE 116

System Definition 117
Steps in a typical application of the DSS . 117

The design philosophy and concepts 120
The computer system and programming languages 123
Plan for the detailed presentation of the DSS 127
The Data Subsystem 134
The problem framework data file, SETUP DSS 139
The enrollment data files 141
The Dialog Subsystem 146
Executive Control Program DSS 150
Component Control Program ENROLL 158
Component Control Program FINANCE 162
Component Control Program SETUP 168
Component Control Program COMPARE 169
Component Control Program AGGR 172
Component Control Program PERCOST 176
Component Control Program CHGTIME 177
Component Control Program PEERINST 180
Component Control Program PEER 189
Component Control Program GENIND 191
Component Control Program PRIORITY 192
Component Control Program MPS 196
Component Control Program FORMAT 200
Component Control Program GP 200
Component Control Program REPORTS 202
The Models Subsystem 204
The linkage between the DSS and IFPS . . . 207
IFPS Command files 209
IFPS model ENROLL 210
Calculation of the decision-variable coefficients for the state revenue requirement goals. 211
Calculation of enrollment values 213
IFPS model FEES 216
IFPS models DD1 and DD2 220
IFPS Model COST 226
Tuition and fee recovery rate 226
State revenue requirement goal target value 231
Percentage-of-average-total-cost goals . 232
Change-in-tuition-charges-over-time goals 233
IFPS Model PEER 233
IFPS Model TUITION 239
Summary 243

V. IMPLEMENTATION, CONCLUSIONS, AND RECOMMENDATIONS 245

Implementation 245
Evolution of the system 246
Technical testing of the system's components 249

Demonstration of the system's use and capability 251
Application of the DSS to a real problem . 254
Conclusions 261
The use of goal programming to model the tuition process 262
The use of an automated system to formulate and solve GP models 263
Translation of the GP model into a decision-making aid 264
Recommendations 268

BIBLIOGRAPHY 271

Appendix page

A. PROGRAM LISTINGS FOR THE EXECUTIVE CONTROL PROGRAM (ECP) 280

B. PROGRAM LISTINGS FOR THE COMPONENT CONTROL PROGRAMS (CCP) 298

Component Control Program AGGR 299
Component Control Program CHGTIME 303
Component Control Program COMPARE 305
Component Control Program ENROLL 307
Component Control Program FINANCE 313
Component Control Program FORMAT 318
Component Control Program GENIND 319
Component Control Program GP 321
Component Control Program MPS 322
Component Control Program PEER 324
Component Control Program PEERINST 325
Component Control Program PERCOST 330
Component Control Program PRIORITY 332
Component Control Program REPORTS 334
Component Control Program SETUP 337

C. PROGRAM LISTINGS FOR THE FORTRAN PROGRAMS 338

FORTRAN Program AGRGMPS 339
FORTRAN Program CAPGEN 342
FORTRAN Program CAPIFPS 343
FORTRAN Program CHGTIME 345
FORTRAN Program COMPGEN 348
FORTRAN Program COMPMPS 350
FORTRAN Program COMPSET 355
FORTRAN Program CTIMEGEN 358

FORTRAN Program GENIND 359
FORTRAN Program GENINGEN 361
FORTRAN Program PCOSTGEN 362
FORTRAN Program PEERCALC 363
FORTRAN Program PEERGEN 366
FORTRAN Program PEERMCRO 368
FORTRAN Program PERCOST 369
FORTRAN Program PRIOR 371
FORTRAN Program PRIORCHK 374
FORTRAN Program ROWS 375
FORTRAN Program SETUP 376
FORTRAN Program SUPLMREV 378
FORTRAN Program USER 379

D. IFPS-RELATED PROGRAM LISTINGS 394
Linkage Program IFPSDSS 395
Linkage Program QUIT 397
IFPS Command File COSTCOM 398
IFPS Command File DDFEECOM 399
IFPS Command File ENRLCRED 400
IFPS Command File PEERCOM 401
IFPS Command File TUITCOM 402
IFPS Model COST 403
IFPS Model DD1 405
IFPS Model DD2 407
IFPS Model ENROLL 410
IFPS Model FEES 414
IFPS Model PEER 418
IFPS Model TUITION 419
IFPS Report Generator COSTINFO 422
IFPS Report Generator CREDIT 424
IFPS Report Generator CREDMPS 425
IFPS Report Generator DDFEEMPS 426
IFPS Report Generator DORMDING 427
IFPS Report Generator ENROLL 429
IFPS Report Generator FEES 432
IFPS Report Generator INDEXCHG 433
IFPS Report Generator PEERMPS 434
IFPS Report Generator PERCOST 435
IFPS Report Generator ROUND 436
IFPS Report Generator STATEREV 438
IFPS Report Generator TOTCOST1 439
IFPS Report Generator TOTCOST2 440
IFPS Report Generator TOTCOST3 441
IFPS Report Generator TOTCOST4 442

E. FILES NOT DISPLAYED IN THE TEXT 443

File CHGTIME IFPS 444
File COST IFPS 445
File DDBASE1 IFPS 447
File DDBASE2 IFPS 449
File ENROLOUT IFPS 451
File FACTOR IFPS 456
File FEEBASE IFPS 458
File FILE FT08F001 460
File GENIND IFPS 470
File PEER IFPS 471
File PEER_ IFPS 472
File PERCOST2 IFPS 473
File SETUP IFPS 474
File SUPLMREV IFPS 475
File TUITHIST DSS 476

F. R^2 VALUES, BY STUDENT CATEGORY, FOR TUITION CHARGES VERSUS 0, 1, AND 2-YEAR LAG IN THE CPI 477

G. DESCRIPTION OF THE GOALS INCLUDED IN THE DSS . . . 479

H. HARDCOPY VERSION OF THE COMPUTER SESSION USED TO DEMONSTRATE THE DSS 483

VITA 543

LIST OF TABLES

Table — page

2.1. Components of student charges and their relationship to student category. 56

3.1. Translation of the decision-variable notation from the general model to the computer solution routine. 97

3.2. Representive goal structure and its corresponding numbering scheme for the model's variables. . . . 98

4.1. Using the DSS involves dual and shared responsibilities between the DSS and the decision maker. 119

4.2. Summary of the tasks that each of the programming languages perform in the DSS. 128

4.3. Summary of where an example of each data file is displayed[1]. 137

4.4. Naming convention for the peer institution data files. 182

4.5. R^2 values for tuition charges versus 0, 1, or 2-year lag in the CPI. 240

5.1. Summary of the implemented model's solution . . . 258

5.2. Summary of the implemented model's level of goal attainment. 259

LIST OF FIGURES

Figure page

1.1. Establishing the tuition charge structure - allocating revenue requirements to student categories. 3

1.2. The tasks performed by the DSS. 13

1.3. Expected interaction between the user and the system. 16

2.1. Projected percentage change in the number of high school graduates from the 1979 level. 33

2.2. Projections and possibilities for total enrollment in postsecondary education to 1990. 35

2.3. Tuition-and-fee revenue percentage of adjusted E&G appropriation. 40

2.4. The effect of increased fiscal pressures and political influence. 46

2.5. Timing of key activities in the tuition and fee process. 49

2.6. Establishing the tuition and fee charge structure. . 53

2.7. The "per-credit-hour-to-plateau" tuition charge function, for a typical student category. 60

3.1. Allocating revenue requirements to student categories. 65

3.2. Overview of the GP Solution approach used in the DSS. 92

3.3. Translation of the priority structure to an achievement function. 101

3.4. Steps in the GP solution process. 108

3.5. Display of the GP solution - value of the decision variables 110

3.6. Goal achievement report of the GP solution. . . . 113

3.7. Trace of the decision variable values at each priority level. 114

4.1. The software architecture that is the basis for the DSS design. 122

4.2. Flow-chart diagram of the DSS - Part 1. 129

4.3. Flow-chart diagram of the DSS - Part 2. 130

4.4. Flow-chart diagram of the DSS - Part 3. 131

4.5. Flow-chart diagram of the DSS - Part 4. 132

4.6. The problem framework data file, SETUP DSS. . . . 140

4.7. Primary enrollment data file format (file AENROLL1 IFPS). 142

4.8. Secondary enrollment data file format (file CAENROLL IFPS). 144

4.9. The relationship between the executive control program (ECP) and the component control programs (CCPs). 148

4.10. Display for specifying which relative and absolute goals to include in the model (file COMPGEN). . 155

4.11. Capital projects data files: CAPGEN for user input and CAPIFPS in IFPS-readable format. 167

4.12. Example of a goal represented in its four forms. 171

4.13. Example of a peer institution data file (file NUONLGTO). 183

4.14. The display for providing the peer institution information (file PEERGEN). 184

4.15. The display for setting the priority structure (file PRIOR). 193

4.16. The decision variables in MPS format (file DSSXS). 198

4.17. A portion of the GP model in equation form (file FORMAT). 201

4.18. The display for selecting which reports to print (file REPORTS). 203

4.19. Calculations are performed in the DSS by accessing IFPS directly from a CCP. 208

4.20. Summary of the credit hour data used in the state revenue requirement goal (file CREDOUT). . . . 214

4.21. Process for calculating the quarterly charge for the various fee categories. 217

4.22. Summary of the student fee charges generated by the DSS (file DDFEEOUT). 221

4.23. Summary of the dorm-and-dining charges generated by the DSS (file DDFEEOUT). 225

4.24. Summary of the cost data used in the DSS (file COSTINFO). 227

4.25. Display of key indicators used to establish the change-over-time goals (file INDEXCHG). 234

4.26. Regression analysis of tuition charges versus the CPI. 236

4.27. Distribution of tuition and fee charges. 238

4.28. Management report that summarizes the annual tuition and fee charges (file TOTCOST). 242

4.29. Summary of the DSS's three subsystems and their functions. 244

Chapter I

INTRODUCTION

Tuition and fees are a major source of income for colleges and universities and a major portion of the cost of a student's education. As a result of the continued increase in tuition and fee charges, much attention is being focused on the tuition and fee process and the many issues surrounding the establishment of student charges. This growing awareness and concern - shared by educators, administrators, legislators, students, and parents - is precipitated by the fact that tuition and fee charges have increased faster than the Consumer Price Index over the past two decades [72, p. 10]. In fact, according to the College Board, tuition costs at public four-year colleges and universities in the Fall of 1983 increased 12% over the previous year [16, p. 5]. By comparison, the Consumer Price Index (CPI) increased by 2.9% during the same period. Interest and concern are further stimulated by the impending question of how future demographic, political, and economic changes will effect tuition and fee policy.

Tuition charges, the student's share of the institution's operating expense, are primarily used to balance the institution's budget - to offset the difference

between all of the other sources of revenue and the costs of operating the institution. Student charges are often increased in order to bridge the financial gap resulting from escalating operating costs and reduced government support. As a result, public institutions of higher education have become more dependent upon student-generated tuition and fee revenue to finance larger proportions of their operations.

Tuition and fees must be regarded as a compromise or balance between stark financial considerations and university philosophical considerations. For example, a university's policy on how much more to charge an out-of-state student as compared to a state resident depends on charges at peer academic institutions, future enrollment levels, anticipated operating costs, and policy decisions at the state level. As Van Alstyne indicates, "like a pricing decision in a business firm, the tuition levied must be considered in relation to overall demand, tuition levels at competing institutions, net revenue available to cover costs, and cash flow" [91, p. 68].

The process of establishing tuition and fee charges at colleges and universities is illustrated in Figure 1.1. This process can be defined in terms of three steps, the determination of: (1) the institution's planned operating

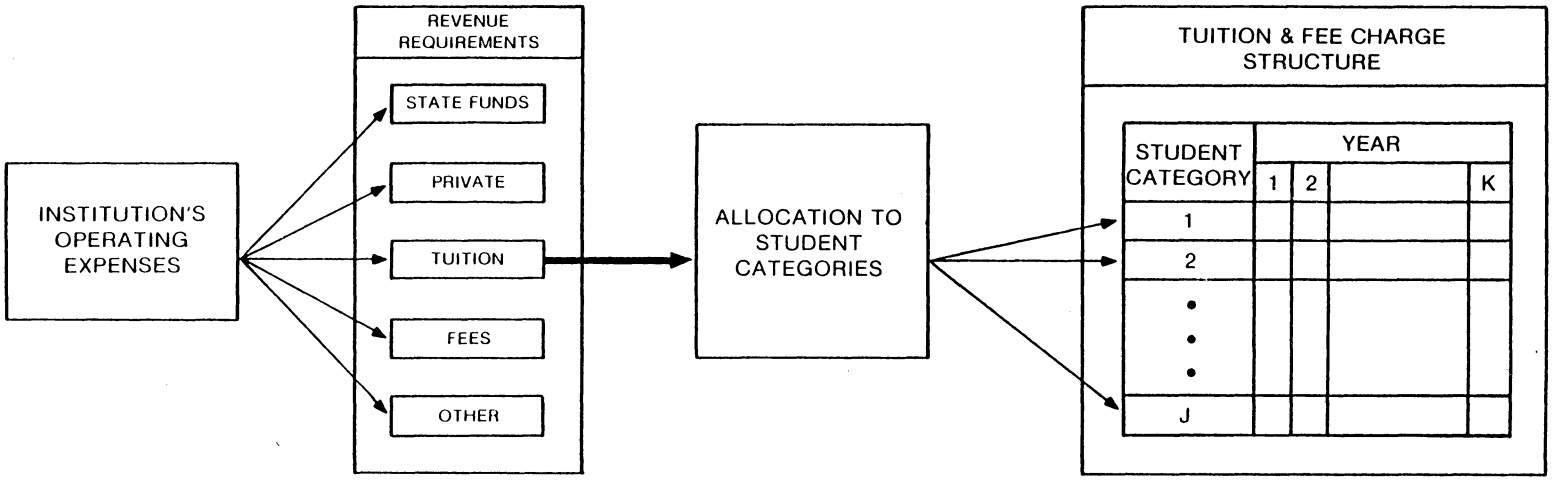


Figure 1.1: Establishing the tuition charge structure - allocating revenue requirements to student categories.

expenses, (2) the proportion of the operating expenses that are required to be offset by tuition revenue, and (3) the tuition charges for each of the student categories. The revenue necessary to cover the costs of operating a public university is primarily obtained from state funds and tuition (other possible sources of revenue include: Federal and local appropriations, gifts, endowments, grants, and contracts).

As shown in Figure 1.1, once the amount of revenue that must be generated from tuition and fees is determined, it must be allocated to individual charges for different student categories. These categories are the combination of various student classifications which include:

1. academic status: undergraduate versus graduate,
2. residency status: in-state versus out-of-state,
3. participation status: full-time versus part-time,
4. location: on-campus versus off-campus, and
5. session: regular versus summer.

The allocation of revenue requirements to each student category is based on the constraints, objectives, and priorities of the institution. The schedule of all of the student category charges is called the "tuition and fee charge structure".

1.1 STATEMENT OF THE PROBLEM

The university administration's[1] task of making sound and effective tuition and fee policy decisions is becoming both more critical and more complex. This is a result of the increased reliance on student-generated tuition-and-fee income, the declining college-age student population, reductions in state and Federal funds, and escalating costs of operation. Therefore, the administration must design or modify the institution's tuition-setting policies and the resulting tuition charge structure in response to the changing demographic, political, and economic conditions.

The process of analyzing and setting tuition and fee charges involves projecting future values, evaluating alternatives, making policy decisions, allocating resources, performing sensitivity analyses, and addressing "what-if" questions. The process requires the assimilation of large amounts of information and consideration of a large number of variables - many of which are interrelated, span multiple time periods, and have uncertain values. Such factors make an automated system including the application of certain analytic methods a necessity.

---

[1] Administration refers to those administrators directly involved in the tuition and fee process: the governing board, chief executive officer, chief academic officer, chief business and finance officer, etc.

State colleges and universities need a model of the procedure that is followed to establish tuition and fee charges. The model needs to integrate all of the variables, factors, and criteria that effect the tuition-and-fee policy-setting and decision-making processes. Such an approach should provide the administration with a means to systematically consider all of the relevant factors in the problem. It also should provide a framework for considering alternative charge structures, assessing the impact of policy changes, and measuring the effect of changes in factors that influence tuition charges. Ideally the system should incorporate a range of criteria for allocating the institution's revenue requirements to student categories; the choice of the criteria and the way that they are combined should be left to the decision maker.

1.2 PURPOSE AND OBJECTIVES

The purpose of this dissertation is to enhance the planning, policy-setting, and decision-making processes associated with establishing tuition and fee charges. The research's objective is to develop, test, and implement a comprehensive computerized system that integrates the necessary data, models, and reports into an on-line interactive user-friendly tuition-and-fee analysis package.

The decision support system is intended to be used by middle- and upper-level administrators - those that are directly involved in the decision-making activities of the institution - to "design" tuition charge structures based on university goals, constraints, and priorities.

1.2.1 Purpose

In order to adequately plan, make appropriate decisions, and set effective tuition policies, the university administration must perform certain activities, including: (1) estimate future financial needs and requirements, (2) determine the effect of changes in variables and parameters that impact tuition and fee charges, and (3) test present and alternative policies under different scenarios. In order to facilitate and enhance the administration's capability for performing these activities, a decision support system with an imbedded goal programming model has been developed through this research. It provides the administration with a means to address tuition and fee problems, at a detailed and sophisticated level without directly handling large amounts of data, structuring the problem, formulating and solving complex analytical models, and composing a series of reports to document the results.

1.2.2 Objectives

The objectives of this research are to model the tuition and fee process and to develop a computerized system that provides a scientific approach for setting tuition charges. This research establishes a framework for the administration to explore tuition-related issues by providing a means to analyze and test present and alternative policies under various conditions and by providing a means to measure the effect of changes in the factors that influence tuition charges.

The decision support system that results from this research is expected to be a valuable and useful part of the policy-setting and decision-making processes. It is a comprehensive package that integrates data and reports with modeling and analysis, but in a manner that does not require the user to be an expert in computers and management science. The system operates interactively, thereby providing immediate feedback, facilitating what-if analyses, and satisfying the need to explore and test alternatives.

1.3 METHODOLOGY OVERVIEW

The basis for this dissertaion is the application of two management science techniques - mathematical programming and decision support systems - to tuition and fee policy analysis. Over the past few years there has been a significant increase in the use of management science techniques in university management, as indicated by Dartmouth College President, John G. Kemeny:

> A century of expansion for higher education has come to a halt. The result has been the raising of consciousness about the necessity for optimizing the use of institutional resources. ... An era of philosophical discussion on the priorities of a college has been replaced by an era characterized by the necessity of tight management by central administration and careful long-range planning. Many institutions independently came to the conclusion that the traditional intuitive judgements must be replaced by sophisticated management methods. [34, p. ix]

1.3.1 Modeling the tuition and fee process via goal programming

The tuition and fee process can be viewed as a resource allocation process with competing and conflicting goals; as Van Alstyne indicates, "... tuition policies ultimately reflect the combination of political choices and economic constraints" [91, p. 69]. Therefore, the administration attempts to establish a tuition charge structure that reflects the institution's philosophy on the level of

tuition that each of the student categories should be assessed, but not without considering revenue requirements, political climate (e.g., percentage of cost paid by in-state students), and market conditions (e.g., charges at peer institutions).

This research represents the tuition and fee process as a multi-year goal programming (GP) model. The university's revenue requirements are allocated to the various student categories based on a set of objectives or goals. These goals include: meeting the state revenue requirements, remaining competitive with peer institutions, maintaining a specified difference in charges between categories, etc. Since it is usually not possible to establish a tuition charge structure that meets all of the goals, the university must priortize the goals and settle for a charge structure that "best" reflects their goals, limitations, and parameters. As Lee and Clayton point out, "... (t)he goal programming approach appears to be the most appropriate technique in developing a model to attain multiple, competitive, and often conflicting goals with varying priorities" [57, p. B-396].

The use of goal programming provides the administration with a means to easily experiment with various charge structures, trade-off alternative policies, and test

policies under various demographic and economic conditions. It provides insight into the effect of including certain goals in the model and lets the decision maker consider various goal-ranking schemes or priority structures. As Lee and Moore note,

> "(d)eveloping and solving the goal programming model points out where some goals cannot be achieved under the desired structure of goals and hence where tradeoffs must occur due to conflicts among the goals. Furthermore the model allows the administration to review critically the priority structure in view of the solution derived by the model. Indeed, the most important property of the goal programming model is its great flexibility which allows model manipulation with numerous variations of constraints and goal priority structure" [59, p. 713].

1.3.2 DSS as a means to translate the GP model into a valuable decision-making aid

Once the tuition and fee process was modeled using a goal programming approach, this research focused on the development of a comprehensive decision support system for tuition and fee policy analysis. The system combines the latest computer technology with management science modeling so that the effectiveness of the mathematical programming technique is maximized, by making it directly and readibly accessible to the administration. As Peter Keen, a leading authority on decision support systems (DSS), states:

"... the most important aspect of the DSS approach is that it translates complex analytic models into usable and useful techniques for decision makers" [46, p. 41].

The term decision support system is used in this dissertation in the context of Moore and Chang's definition [68, p. 8-9], where: "decision" emphasizes the focus on decision making in problem situations rather than on simply information processing; "support" clarifies the computer's role in aiding rather than replacing the decision maker; and, "system" highlights the integrated nature of the overall approach - implying a wider context of man, machine, and decision environment.

As shown in Figure 1.2, the core of the DSS is the GP model. The DSS, through its three subsystems (dialog, data, and models), is used to translate the GP model into a usable and useful technique for the administration. The DSS handles all of the tasks directly associated with the GP model, as shown in Figure 1.2; these include: formulating the goals, calculating the parameter values, establishing the goal targets, specifying the priority structure, solving the model, and analyzing the solution. Figure 1.2 also shows that the DSS performs certain supporting tasks, such as: processing and storing the data, interfacing between the user and the system, providing displays and memory aids,

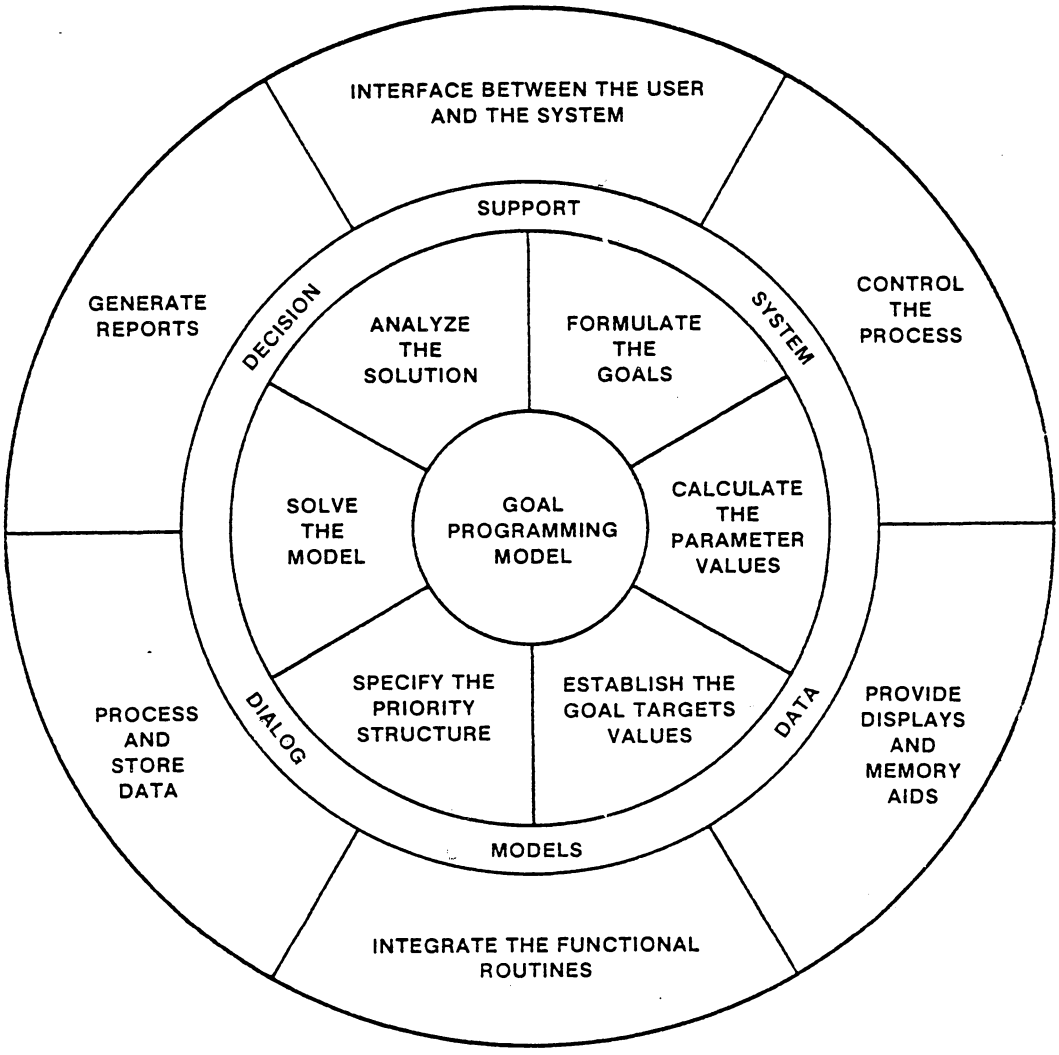


Figure 1.2: The tasks performed by the DSS.

generating reports, integrating the functional routines, and controlling the overall process.

1.3.3 The DSS architecture

The architecture of the DSS is defined in terms of three highly integrated subsystems: (1) dialog, (2) data, and (3) models. The dialog subsystem controls most of the operations and provides the interface between the user and the system. The memory and information handling functions are provided by the data subsystem. The analytic function for calculating the parameter values and solving the GP model is provided by the models subsystem. The software for the DSS was developed using four programming languages/systems: EXEC 2, FORTRAN, IFPS,[2] and, LINDO.[3]

The dialog subsystem is probably the most important since it provides the interface between the user and the system. It is linked to the models subsystem to control the models' operation and data sources. The dialog subsystem is linked to the data subsystem to provide the user with

---

[2] Interactive Financial Planning System licensed from EXECUCOM Systems Corporation in Austin, TX. [20, 27]. Cost to the university: a one-year university support license fee of $2800 and an annual license renewal fee thereafter. The annual renewal fee is currently 50% of the commercial maintenance fee ($6000).

[3] Linear INteractive Discrete Optimizer licensed from Linus E. Schrage, University of Chicago [80, 81]. Cost to the university: a one-time license agreement of $530.

displays, reports, and memory aids and to store the user's input values in data files.

The dialog subsystem is comprised of two types of programs: the executive control program (ECP) and a series of component control programs (CCPs). The ECP serves several functions: it is the primary link between the user and the DSS; it controls and manages the user's session through the use of menus, displays, and control mechanisms that initiate operations when certain conditions exist; and, it provides the user with direct access to common and shared data files. Most of the interface between the models and data is provided by the CCPs; they provide five basic functions: data control, goal formulation, priority structure definition, problem solution, and report generation.

1.3.4 Integration of the DSS into the policy-setting and decision-making processes

Figure 1.3 illustrates conceptually how the user is expected to interact with the system in a typical application. Since the data and the user are influenced by exogenous factors - economic, political, and demographic conditions - the specific actions that are taken will vary from application to application, but the overall process should be similar to that shown in Figure 1.3.

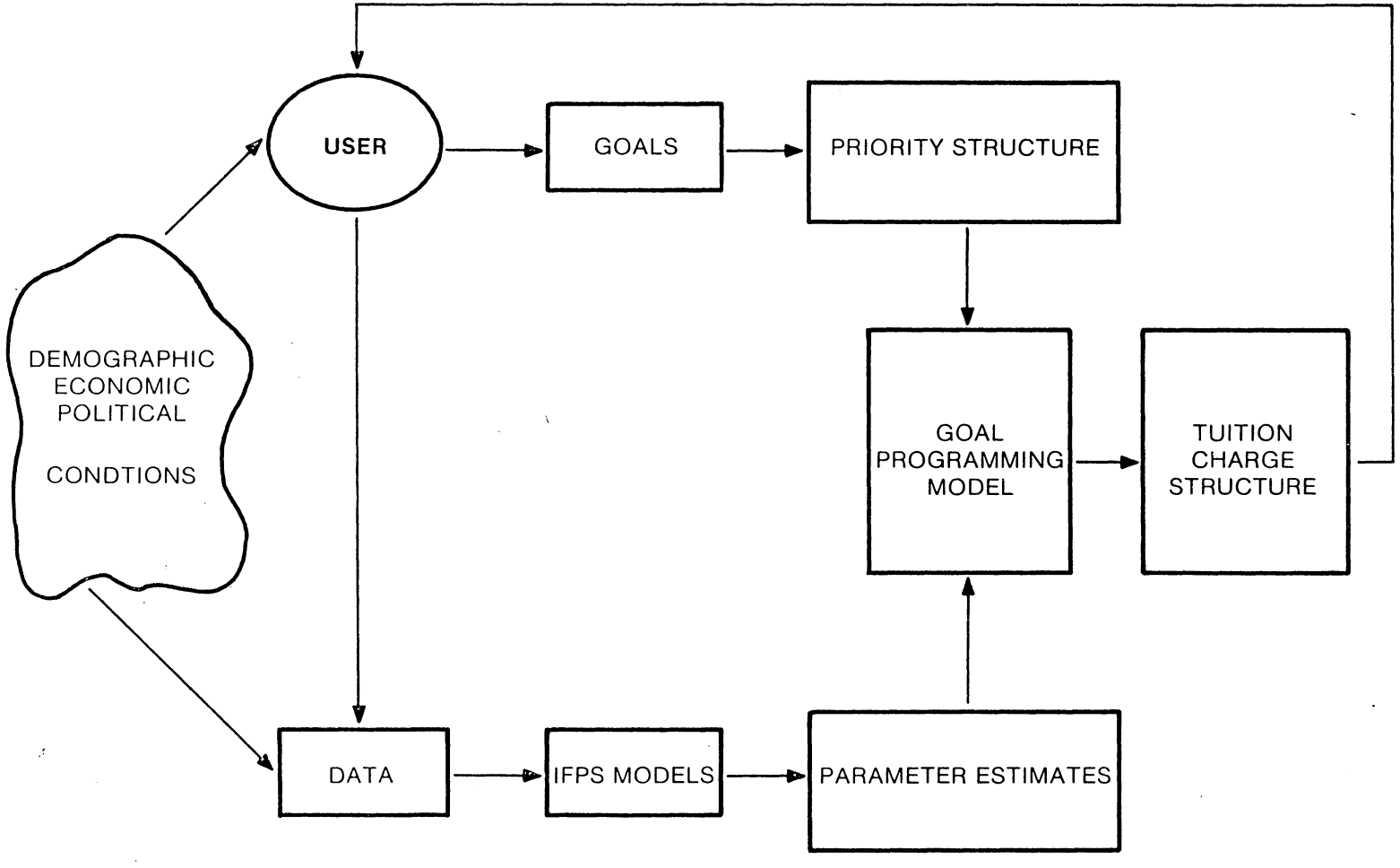


Figure 1.3: Expected interaction between the user and the system.

As previously described, tuition charges are established by a goal programming model imbedded in the DSS that allocates the revenue requirements to student categories. There are two basic inputs to the GP model - goals and parameter estimates - both of which are controlled by the user, as illustrated in Figure 1.3. From a predefined set, the user decides which goals to include in the analysis and ranks them in the order of importance to the university, thereby constructing a priority structure. The parameter values are either obtained directly from the user or calculated by supporting models. The user controls the data (enrollment levels, cost information, inflation rate, fee charges, etc.) that are used by the IFPS models to calculate the parameter values.

The DSS automatically formulates the goals, translates them into a form required by the optimization routine, estimates the appropriate parameter values, solves the problem interactively, and generates a set of reports. Once this has been completed and the charges have been established (the model has been solved), control is returned to the user, as illustrated by the feedback loop in Figure 1.3, for the purpose of exploring and testing alternatives and to perform sensitivity analyses.

1.4 SCOPE AND LIMITATIONS

As stated in the objectives, this research is intended to provide the university with a system that will support tuition and fee analysis through projecting future values and providing a means to evaluate alternatives and perform sensitivity analyses. This is accomplished by modeling the existing process and developing an integrated system of data, models, and reports. This dissertation incorporates various existing techniques (enrollment projections, budget formulas, etc.) into a unified and integrated structure. Although improved methods may be a result of this research effort, the search for such improvements is not within its planned scope.

As with any management science technique, the DSS and its imbedded GP model does not make the policy or decide what the tuition charges should be - the university administartion must still do this - but it does facilitate and enhance the process.

While the system is tested at Virginia Polytechnic Institute and State University, the results of this research are general enough to be used by many of the state-supported institutions of higher education. The overall concept of modeling the tuition and fee process and building a decision support system to analyze the process is applicable to any

college or university. Likewise, the goal programming approach to solving the problem of assigning tuition and fee charges to student categories has widespread potential for application. Some of the models incorporated within the system, especially the "recovery rate model", are specific to Virginia institutions. The data sets and report formats are university-specific; but the substitution of another institution's data would not be a major task.

The system also has potential for application at the state level for the purpose of aggregate analyses. This would simply require the consolidation of all of the institutions' data bases. The use of goal programming may be a useful approach to developing, or at least testing, state policy.

1.5 REVIEW OF RELATED LITERATURE

Three areas of research, relevant to the development of a DSS for tuition and fee policy analysis, are examined: (1) the process of establishing tuition and fee charges, (2) the use of DSSs in academic administration, and (3) the use of GP models for resource allocation planning in academic administration.

1.5.1 The Process of Establishing Tuition and Fee Charges

Tuition issues have been at the forefront of discussions concerning the financing of public higher education since 1973, when two study groups - the Carnegie Commission on Higher Education and the Committee for Economic Development - recommended raising tuition at public institutions and increasing student aid for low-income students [91, p. 66]. Most of the literature concerning tuition policy has been descriptive rather than prescriptive - the focus has been on identifying and discussing the factors and issues that surround tuition policy decisions.

Much of the literature has focused on state policy rather than on the tuition policy of individual colleges and universities. For example, Van Alstyne [91] claims that setting tuition policy at the state level is fundamentally a political decision, resting on questions of priorities and budget tradeoffs. Van Alstyne [91] presents six rationales (social/individual benefits, ability to pay, public service, user tax, education as an investment, tuition as a share of educational cost) for setting tuition levels.

A survey, conducted by McKeown [63], found that 22 states had established tuition and fee policies. The survey also identified three distinct methods for determining tuition charges: indexing to educational cost, adjustments

based on changes to an economic index, and benchmarking to peer institutions. The policy of indexing tuition to educational costs is discussed in detail by Viehland, et. al. [93]. They examine both the factors that must be addressed by policy makers, at the state and institutional level, in order to initiate such a system, and the implications of such a system on planning, budgeting,and enrollment.

Saupe and Blagg [78] focused on the individual institution. Their survey demonstrates the difficulty of collecting and gathering tuition data and, more important to this researach, describes the characteristics of existing procedures that are used to establish tuition charges. Based on 20 major public institutions, the survey revealed that (1) wide variations exist in the terminology used to define student charges, (2) differential charges are assessed to a number of different student categories, and (3) a variety of charge functions are used to relate the number of credits for which a student enrolls and the amount of charges that are assessed.

Using regression and correlation analysis, Rusk and Leslie [76] assessed the relative contribution of 21 variables to predicting tuition prices at major public universities. They view the setting of tuition charges in

higher education in terms of price theory and define the variables as descriptive measures of: prices charged by competitors, market share held by competitors, costs, net income of potential buyers, and quality of the institution.

Troutt [88], through the use of a linear programming model, proposed the only quantitative approach to establishing tuition charges. The single-period model determines, for each student category, the change in tuition charges - both on a per student and on a per credit hour basis. The objective is to maximize marginal revenue, subject to wage and price guidelines (those that were in effect in 1979) and a set of policy constraints (out-of-state tuition three times the in-state rate, part-time tuition rate at least as great as the full-time rate, excess-load rate should not exceed the full-time rate, maximum increase per student and per hour). Troutt [88] mentioned the use of goal programming as a possible approach to establishing the tuition charge structure.

While Troutt [88] presents a systematic and scientific approach to establishing tuition charges, his model: (1) is not sensitive to the multiple goals of the institution, (2) does not include many of the factors that the decision maker must consider, (3) does not provide a sufficiently detailed formulation of the problem, and (4) does not

provide an efficient way to test, evaluate, and analyze alternative policies or measure the effect of changing environmental conditions.

1.5.2 Decision Support Systems in Academic Administration

Even though the concept of DSS began in the late 1960's [46, p. 33] and the conceptual work of the field was mainly carried out between 1970 and 1976 [46, p. 43], DSS is a relatively new information systems technology. For example, the First International Conference on DSS was not held until June 1981 and, according to Sprague and Carlson [85], it was not apparent until late 1980 "that the technical developments, the information needs of managers and professionals, the organization mechanisms, and marketplace products and services were converging to create a viable environment for the development of effective DSS." [85, p. xvi]

Keen [46] listed the following areas where DSS has been successfully applied: portfolio management, merger and acquisition analysis, design of police force beats, redesign of school districts, market planning, corporate planning, manpower planning, state government policy analysis, R&D management, product planning, media planning, and budgeting [46, p. 39]. The development of DSSs for specific

application to problems in academic administration has been quite limited.

Franz, Lee, and Van Horn [22] developed a DSS for academic resource planning based on a single-period goal programming model of an academic department. The model is used to establish the department's personnel requirements and corresponding budget, based on a set of rank-ordered goals, and to investigate the impact of policy changes on staffing and budget requirements. The model's goal formulations consider: faculty distribution by rank, support staff, credit-hour demand, level of salary increases, advising activities, contractual commitments, and budget levels.

The GP model is only one of three subsystems included in their DSS. The second subsystem facilitates the determination of the decision maker's priority structure; the third subsystem provides feedback on the model's results and the level of goal attainment, providing an evaluation of the current solution and a basis for exploring alternative solutions.

1.5.3 The Use of GP Models for Resource Allocation Planning in Academic Administration

In this research, the process of establishing tuition charges is viewed as a resource allocation problem - the allocation of a revenue requirement to individual charges for each student category. Linear programming is one method used to allocate scarce resources to competing activities and Troutt [88], as previously described, proposed the use of linear programming to establish the tuition charge structure. However, this approach does not recognize the multidimensionality of the university's objective. A goal programming model is more appropriate since it allows for the inclusion of multiple, competitive, and often conflicting goals and for experimentation with various goal structures or objectives of the university [60, p. 218]. Goal programming has been applied to a wide variety of problems in academic administration but most have been used as a means to allocate faculty and dollars within the university.

The use of goal programming models for the purpose of university budget planning and resource allocation was first presented by Lee and Clayton [57]. Their model determines the personnel requirements for a single academic unit, i.e. the number, rank, and compensation for an academic department or college. The single-year allocation is based

on a set of goals that includes the following: accreditation guidelines, level of salary increase, credit-hour demand, faculty distribution by rank, faculty/student ratio, faculty/staff ratio, faculty/graduate assistant ratio, and payroll budget.

Schroeder [83] also presented a goal programming model for determining the number of faculty, staff, and graduate assistants, but for several academic units and over multiple time periods. The personnel allocations are established based on a set of budget and faculty flow constraints and a series of goals that consider: faculty distribution by rank, faculty instruction loads, faculty/staff ratio, and faculty/graduate assistants ratio.

Lee, Van Horn, and Brisch [60] developed a goal programming model for analyzing resource allocation patterns and decision processes in academic departments. The earlier Lee and Clayton [57] model was extended to include differences between undergraduate and graduate activities, undergraduate advising, graduate supervisory responsibilities, departmental operating budgets, and a professional development fund. More importantly, constraints and goals were added to compare the model's "optimal" solution with actual departmental data, allowing the model to be used as an analytical tool rather than a

descriptive or prescriptive technique for academic planning. Prior to using the model to simulate the effect of various policy decisions, the "revealed preferences" of previous resource allocation decisions must be established. This was accomplished by testing various priority structures until a "reasonable fit" to actual departmental data was found. This is a somewhat unique approach to assessing the effect of changes in the model's parameters as well as changes in managerial preferences.

A goal programming model was developed by Lee and Moore [58] to assist in admissions planning. The model establishes the number of applicants that should be admitted into each student category, based on a set of prioritized institutional and societal goals. The set of goals formulations included the following considerations: state residency policy, admission standards, physical plant, student mix, and residence hall capacity.

A mixed-integer goal programming model for capital budgeting within a university was developed by Keown, Taylor, and Pinkerton [50]. The model is used to select projects based on a set of rank-ordered goals that consider: capital budgets, operating expenses, building construction, equipment purchases, accreditation, political-social demands, and improved departmental performance.

The problem of allocating library funds for books and standing orders is modeled by Hannan [29] using goal programming. The goals include the total number of units purchased, total cost of expenditures, teaching worth, research worth, and percentage of departmental budget used for books.

Kendall and Luebbe [49] developed a GP model to assist in the management of college student recruiting activities. The model identifies both the type and number of activities that must be completed each quarter in order to reach an enrollment goal for a given year. Factors such as manpower, time, budget, and marketing strategies are included in the goal formulations.

The models just described, as well as the model developed in this research, all use goal programming with preemptive priorities. But, it is not the only multiple criteria decision making technique that has been applied to problems in academic administration.

Said [77] used goal programming with a weighted objective function in his budgeting and financial planning model. The model establishes the level of the institution's budget request for each expense item and allocates financial resources simultaneously to both academic activities (instruction, approved research, academic administration,

and professional development) and academic administrative units. The model also includes a set of relationships that link the application of budget funds to specific financial sources (state appropriations, research contracts, and gifts.)

Walters, Mangold, and Haran [95] also used a weighted objective function in their multiperiod model for establishing faculty staffing levels across multiple academic units. The goal formulations considered faculty staffing objectives, career constraints, teaching and course load requirements, tenure constraints, and budget constraints.

Geoffrin, Dyer, and Feinberg [24] presented a model that allocates an academic department's faculty effort among such activities as: teaching, departmental service, research, and student counseling. The model is solved interactively by a technique that uses a large-step gradient algorithm to move toward solutions that have improved utility function values. Improved solutions are based on successive tradeoffs between objectives at a specific achievement level, without explicitly defining the utility function.

A modified version of the solution approach described above is used by Wehrung, Hopkins, and Massy [96] in their

long-range planning procedure that is used to identify new directions for the university to follow. Their model is an aggregate representation of a university, defined in terms of 12 planning variables (i.e., faculty size, student body size, tuition level, tenure ratio, staff/faculty ratio, growth rates) and a set of financial constraints. The administrator's implicit preferences, rather that a standard objective function, are used by an interactive mathematical programming algorithm to guide the search process toward improved solutions.

1.6 PLAN OF PRESENTATION

Chapter II, entitled The Tuition and Fee Process, describes the procedure for establishing tuition and fee charges in terms of three steps: (1) determining the institution's operating expenses, (2) determining the proportion of those expenses that must be offset by tuition revenue, and (3) allocating the revenue requirements to specific student categories. The chapter also describes the environmental factors that influence the tuition and fee process, i.e. demographic changes, fiscal pressures, and political influences.

Chapter III, entitled The GP Model Component of the DSS, includes the formulation of the goal programming model

developed to allocate the institution's annual revenue requirements to various student categories. In addition to the general model formulation, the chapter describes the methodology that was developed in order to solve the model.

Chapter IV, entitled The Overall Decision Support System Architecture, describes the comprehensive system that was developed to formulate and solve the GP model described in Chapter III. The system's architecture, logic, and software are defined through the detailed description of the system's three highly integrated subsystems - dialog, data, and models.

Chapter V presents the conclusions and recommendations of the research as well as a discussion of the implementation of the system.

Chapter II

THE TUITION AND FEE PROCESS

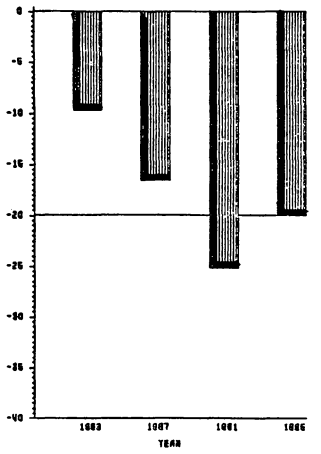
The objectives of this chapter are to describe the primary environmental factors that influence the tuition and fee process and to describe the procedural steps involved in setting tuition and fee charges.

2.1 ENVIRONMENTAL FACTORS

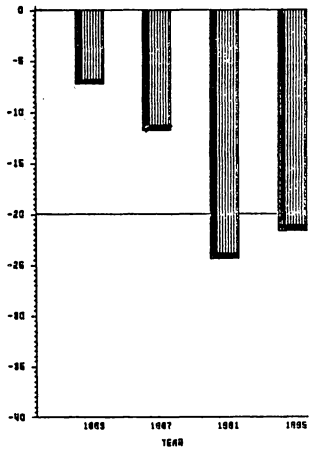
Each of the environmental factors (demographic changes, fiscal pressures, and political influences) and their effect on tuition and fee policies and charges are discussed in this section.

2.1.1 Demographic Changes

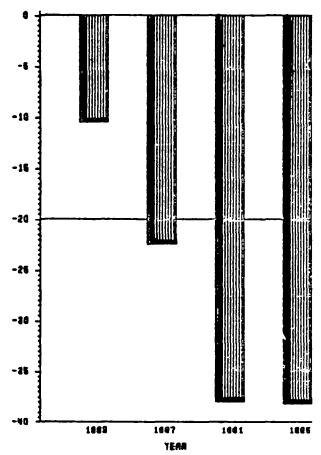
One of the most significant demographic changes that will effect colleges and universities is the projected decline in the number of high school graduates. The number of high school graduates per year peaked in 1979 and is expected to decline dramatically in the 1980s and 1990s [14, p. 19]. In fact, nationally there is expected to be 20% fewer high school graduates in 1995 than there were in 1979, as shown in Figure 2.1. Figure 2.1 compares the projected high school graduate population for the United States



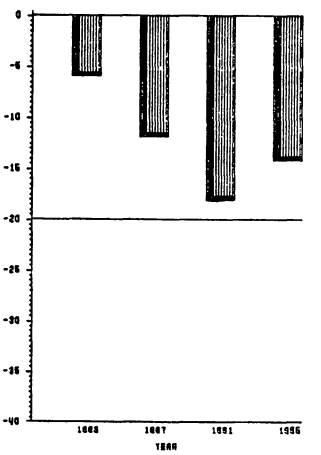
A. United States



B. Virginia



C. Northeast



D. Southeast

Figure 2.1: Projected percentage change in the number of high school graduates from the 1979 level.

Source: H. S. Graduates: Projections for the Fifty States. WICHE, 1979, p. 23-31.

overall (Panel A) to Virginia (Panel B) and to two select regions - the Northeast (Panel C) and the Southeast (Panel D). As illustrated by Figure 2.1, there is expected to be 12% fewer high school graduates in Virginia in 1987, compared to 1979, and 24% fewer in 1991. But even more alarming is the expected decrease in the Northeast, 22% fewer graduates are expected in 1987 and 38% fewer in 1991, compared to the 1979 level. The projections in Figure 2.1 are for public high schools only and are based on estimated birth rates, migration, persistence rates, and the interplay of public and private schools [30].

Most forecasters agree that the college-age population will decline in the 1980s and 1990s, since the number of people in the various age groups are fairly accurately known and most projections use the Bureau of the Census data as a base [25, p.370]. Where there is a wide difference in forecasts is in the degree of attendance or the enrollment rate in higher education, as evidenced by Figure 2.2. Centra [14], Glenny [25], and Leslie [61] all provide summaries of the various projections shown in Figure 2.2, including the forecaster's rationale and assumptions.

There are a wide variety of factors and assumptions that have affected these predictions; it is the diversity in views regarding the future that accounts for the wide

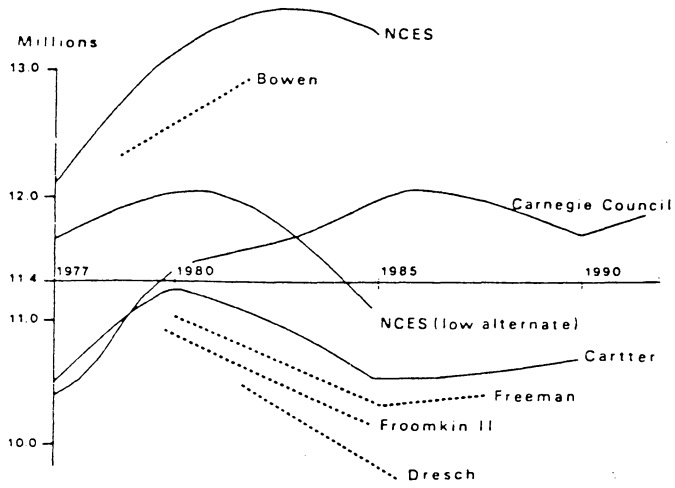


Figure 2.2: Projections and possibilities for total enrollment in postsecondary education to 1990.

Notes: 1) Relative to actual 1977 total enrollment of 11.4 million.

2) The lines for Bowen, Freeman, Dresch, and Froomkin's second scenario are not based on actual figures but instead are an approximate illustration of their views.

Source: Centra, J. A. "College Enrollment in the 1980s, Projections and Possibilities" Journal of Higher Education. 1980, vol. 51, no. 1, p. 24.

variation in the forecasts. Some of the studies considered changes in the fertility and birth rates and changes in the high school graduation rate, college attendance rate, and dropout rate by sex, race, and income level. Other studies addressed market-related conditions, such as: shifts in the labor market, demand for public school teachers, number of academic openings, level of Research and Development spending, the diminishing income gain from a college degree, and the stability of enrollments in graduate and professional programs. Even broader conditions were considered by some forecasters, i.e growth of the U.S. economy, level of student financial aid, social and cultural changes, and the expanding service sector. The status of the military draft, military recruiting policies, and the level of veteran benefits were also considered. In addition, some of the studies accounted for expanding non-degree program enrollments, higher attendance by adults, women, and minorities, and the increase in other-than-collegiate postsecondary training and educational opportunities. Finally, as Leslie [61, p.10] indicates, some of the effects of the decrease in the student-age population will be offset by the fact that during hard times

collegiate institutions will seek innovations, similar to business entreprenuers, in order to maintain their position in the marketplace.

Even though the predictions range from a drastic decrease to a moderate increase in enrollment, most foretell decline or as Centra concludes "all indicators point to a reversal in the growth spiral in higher education for at least the next decade" [14, p.37]. One thing that is certain from all of the predictions is that the level of enrollment in higher education in the future is very uncertain.

2.1.2 Fiscal Pressures

Fiscal changes, both declining state revenues and increasing institutional budgets, have combined to form the second primary factor influencing tuition and fee policies. Colleges and universities have seen their operating budgets escalate while many public institutions have not received comparable state-government support; the difference has had to come from tuition and fees. Like the demographic changes, the future economic conditions are at best uncertain. The institution must certainly consider its impending cost of operation, the level of Federal and state support, and the level that the students will be asked to pay.

Institutions of higher education have faced a period of rapidly rising costs for such items as fuel, books and periodicals, salaries, and equipment. For example, between 1978 and 1982, the cost of utilities more than doubled, the cost of fringe benefits increased 63%, and the cost of books and periodicals increased more than 50% [31, p. 2].

There have been other economic factors, besides double-digit inflation, that have strongly influenced these costs. For example, market pressures, caused by a shortage of new Ph.D.s and salary competition from industry, have driven up faculty salaries in such areas as engineering, computer science, and business. Also higher contributions to social security, escalating major medical costs, increased governmental demands for data and accountability, purchase of computer hardware, and new expenses for the handicapped and implementation of affirmative action plans have increased the cost of operating a college or university [48, p. 11]. There is little indication that these pressures will subside in the near future.

2.1.3 Political Influences

The final exogenous factor influencing tuition and fee policy is the attempt by state governments to allocate their scarce resources "fairly" or uniformily to all institutions.

There is a desire, on the part of state legislatures, to allocate funds based on student academic level and residency status, to link tuition charges with institutional costs and mission, and to reduce the sensitivity of the budgeting process to enrollment fluctuations.

In the last few years, in Virginia, there has been a dramatic shift in the payment responsibility for postsecondary education from the Commonwealth's General Fund revenue to student tuition-and-fee revenue. The continued change in the percentage of operating expenses that must be recovered from tuition and fee revenue and the change in the methodology used to determine this percentage make the planning and decision-making processes considerably more uncertain and complex.

To illustrate how volatile the tuition and fee recovery rate has been during the last decade, Figure 2.3 compares the 1976-77 and 1984-85 recovery rates for each senior institution in Virginia. While the rate decreased for some of the institutions, it increased significantly for many others. This shift in payment responsibility coupled with increasing costs have caused a precipitous rise in the tuition revenue requirements at many institutions. As an aside, it is interesting to note that the recovery rate for all senior institutions in Virginia had converged to 30% in

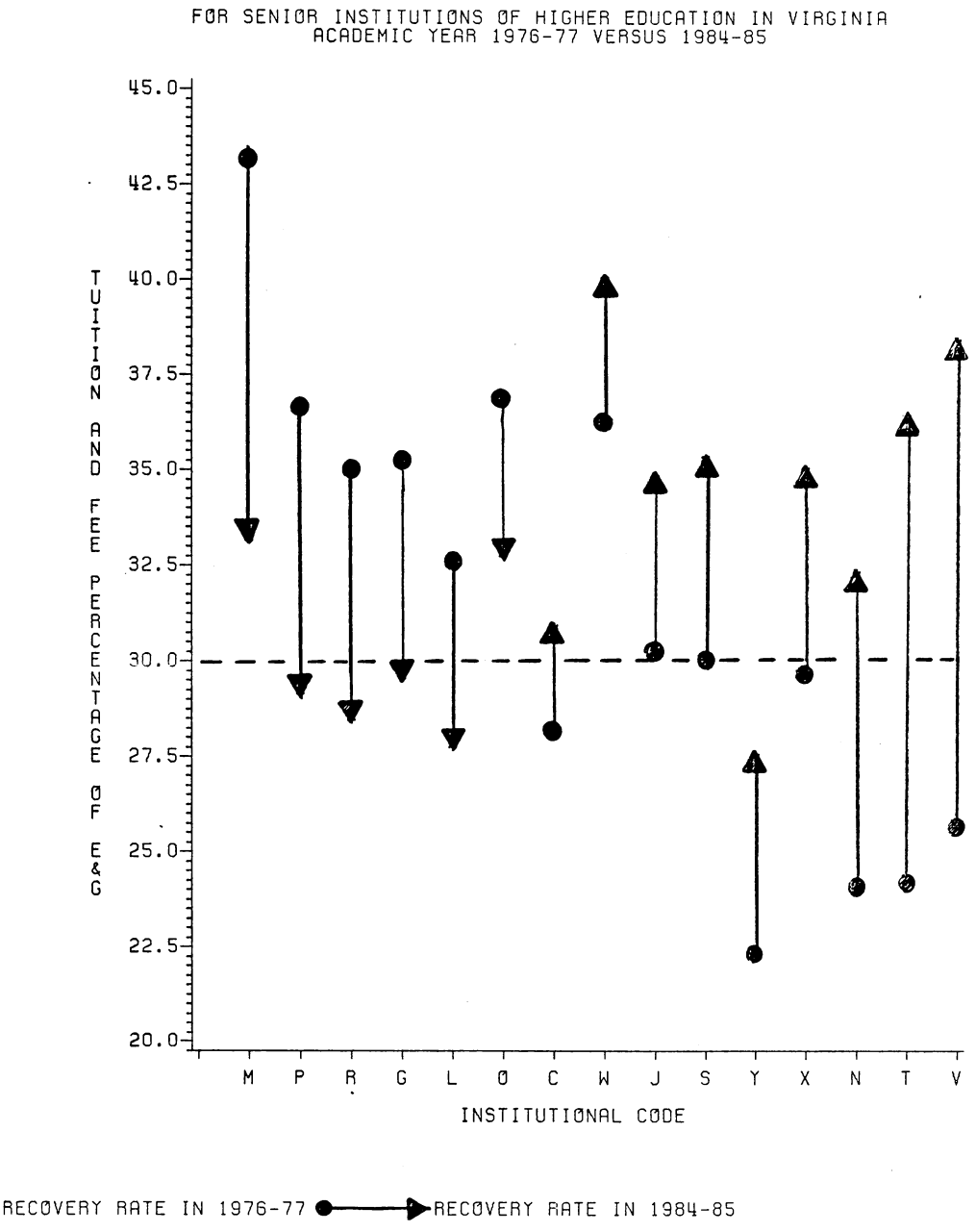


Figure 2.3: Tuition-and-fee revenue percentage of adjusted E&G appropriation.

1981-82. It is obvious from Figure 2.3 that they have begun to diverge again; the recovery rates for 1984-85 will range from 27.5 % to 40.0%.

2.1.3.1 Evolution of the tuition and fee recovery rate

The legislature of the Commonwealth started to provide guidance on tuition and fee policies to the state-supported institutions of higher education in 1976 via the "70/30 Plan". The plan's goal was to narrow the tuition gap between state institutions by requiring that the student bear thirty percent of the educational costs; the remainder was to be provided by the State's General Fund.

At the time of the 70/30 Plan's inception, in 1976-77, there was a wide diversity in recovery rates, from 22.3% to 43.2%. The State's higher education system encompasses a wide variety of institutions that differ significantly in size, composition, quality, program offerings, and mission. Recognizing this wide diversity of institutions, the plan was intended to balance the proportion of instuctional costs charged to students, not attempt to create uniform tuition charges across the state system.

The general philosophy of the state government was that, overall, the student should provide one-third of the cost of public education. In 1971 the Virginia Advisory

Legislative Council (VALC) recommended that tuition rates be set at one-third the estimated Educational and General (E&G) cost of undergraduate education; this was similar to the results of the 1973 Carnegie Commission study. The VALC also recommended that in-state graduate tuition be set at 150% of the undergraduate rate.

The basic 70/30 Plan has undergone considerable changes in its relatively brief history. The definition of the cost of instruction was changed in 1978 to include the cost of fringe benefits. All colleges and universities had completed their phase-in to a 30% rate by 1981-82; but, by this time the 70/30 Plan was found to be inadequate. The inadequacy was the result of (1) continuing fiscal pressures, both in terms of state revenues and institutional budgets, (2) the desire to distinguish between student level and residency status, (3) the desire to link tuition charges closer to institutional-specific costs and mission, and (4) the desire to reduce the sensitivity of the budgeting process to enrollment-driven fluctuations. Some institutions also felt that for all senior institutions to recover the same percentage of their budget in tuition and fees created a competitive advantage for the institutions with large out-of-state and graduate student enrollments, since they could shift the burden of paying higher fees onto

these students rather than onto in-state undergraduate students.

For the 1982-84 biennium, Virginia's tuition policy changed again. The new approach, the "65/35 Plan", called for student contributions to increase to 35% and the state's portion decrease to 65%. The 65/35 Plan, unlike the 70/30 Plan, was institutional-specific; it was dependent upon the institution's student enrollment mix. The plan called for Virginia undergraduates to pay 30% of the cost of instruction while non-resident undergraduates and all graduate students to pay 50%. A cap was placed on the rate so that no institution had to recover more than 32.5% in 1982-83 and no more than 35% in 1983-84.

The State Council for Higher Education in Virginia (SCHEV) formed a tuition-and-fee task force in 1982 to investigate other methods for calculating the amount of tuition and fees that each institution should be required to collect. The mission of the task force was to find an approach that provided a comprehensive and equitable tuition and fee policy in accordance with the legislature's mandate to :

> encourage institutions of higher education to offer the best possible education to students who are residents of the Commonwealth at the lowest

> possible tuition and fees, and to maintain and improve the quality of the Commonwealth's nationally recognized graduate degree programs.[4]

After considering a wide range of alternatives, the task force adopted a formula approach, called the "tuition and fee recovery rate".

The recovery rate formula, containing state-directed and institutional-specific variables, specifies for each public institution of higher education in Virginia the percentage of the state appropriation that must be generated from tuition and fees. It indexes the level of tuition-and-fee revenue to the institution's General Fund appropriations, apparent historical costs, student enrollment mix by academic level and residency, level of out-of-state graduate student employment, and the Commonwealth's policy that in-state students should pay for 25% of the full cost of instruction and non-resident students should pay three times the amount paid by in-state students.

As a result, the Board of Visitors, at each institution, must subsequently set the actual tuition and fee charges considering both the state's recovery rate requirement and the university's philosophy on how tuition and fee charges should actually be structured.

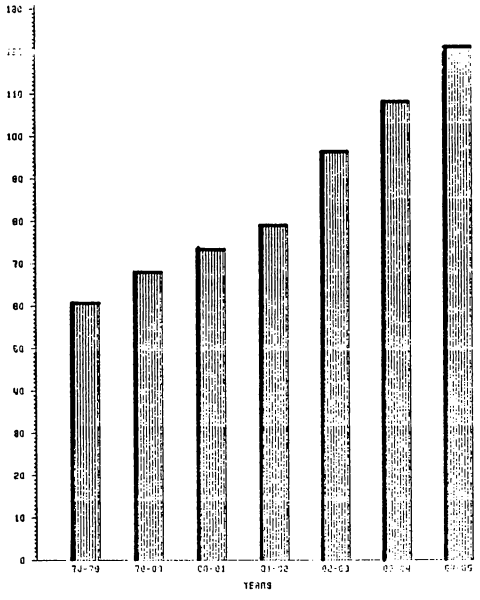
---

[4] 1982-84 Appropriations Act, Commonwealth of Virginia.

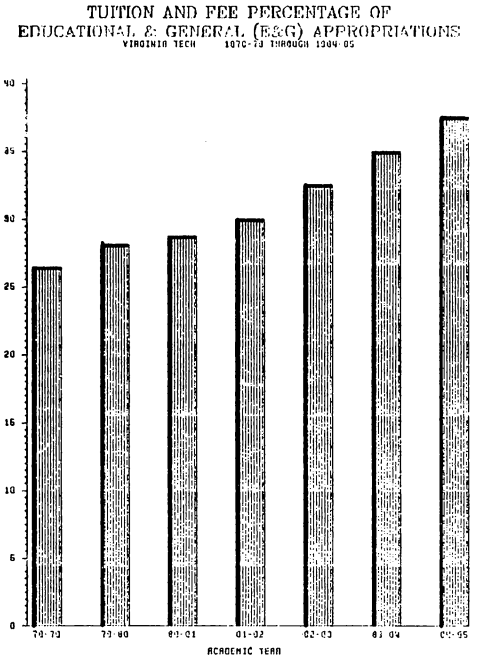
2.1.4 Impact of the Environmental Factors

There are two major outcomes from the increased influence of the demographic, economic, and political factors. First, they are causing educational institutions to become more reliant on tuition and fees as a revenue source, thereby causing a dramatic increase in tuition and fee charges. Second, they are significantly affecting the budgeting, policy-setting, and decision-making processes at each institution.

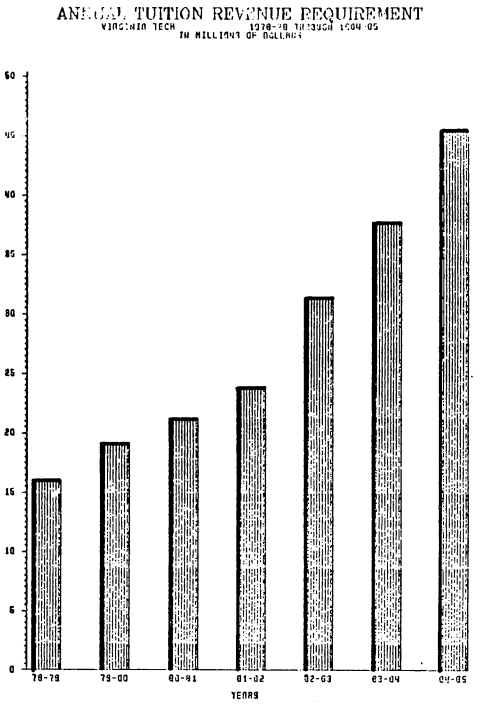
Using Virginia Polytechnic Institute and State University as a case example, Figure 2.4 illustrates the combined effect of increased cost and increased government involvement on tuition and fee charges. Panel A of Figure 2.4 shows the historical and projected growth in Educational and General appropriations necessary to operate the university. The increase in the tuition and fee recovery rate, shown in Panel B, illustrates the direction taken by state government to shift more of the payment responsibility for instructional costs from the State's General Fund revenue to student-generated tuition and fee revenue. Therefore, the increase in student charges is not only the result of higher operating costs at the university but is also the result of the state's decision to have the student bear a higher proportion of these costs.



A.



B.



C.

Figure 2.4: The effect of increased fiscal pressures and political influence.

The combination of these two effects over time results in the tuition-and-fee revenue pattern, shown in Panel C of Figure 2.4. While operating expenses, adjusted E & G appropriations, doubled between 1978 and 1984 (an annual rate of 12.1%), the tuition revenue requirement nearly tripled (an annual rate of 18.9%). As a point of comparison, the CPI increased by 59% over the same period (an annual rate of 8.0%).

Had Virginia Tech also been faced with a period of declining enrollment, the tuition and fee charges per student would have been higher. As enrollments decrease, the effects of the diseconomy of scale, aside from increasing costs, drive up the cost per student. Also a decrease in the non-resident student population causes the university to shift a significant portion of the payment burden to in-state students in order to meet revenue requirements.

2.2 PROCEDURAL STEPS OF THE PROCESS

The process of setting tuition and fee charges is composed of three interrelated steps that span a considerable period of time. These steps include: (1) the determination of the institution's planned operating expenses, (2) the determination of the proportion of revenue

that must be derived from tuition and fees in order to offset those expenses, and (3) the allocation of the tuition and fee revenue requirement to specific student categories. The steps are outlined below, using the 1984-86 biennium in Virginia as an example. Figure 2.5 shows the timing of key activities within each step of the tuition setting process.

2.2.1 Budget activities

The institution's planned operating expenses are determined through the biennial budget process. This first step is shown in the top section of Figure 2.5 labeled Budget Activities; it covers a four-year cycle, from initial planning through final funding. The 1984-86 budget process started in the Fall of 1982, with each institution supplying revenue projections to the state (tuition and fee revenue, Federal appropriations, grant and contract support). The institution's budget submission or appropriation request for planned operating expenditures undergoes a series of reviews and revisions before it is finalized (in mid-1984).

Several of the key budget activities are indicated in the upper portion of Figure 2.5. Initially each institution submitted a Program Proposal detailing its objectives and associated financial needs for the next three biennia. The state subsequently determined a target budget for each

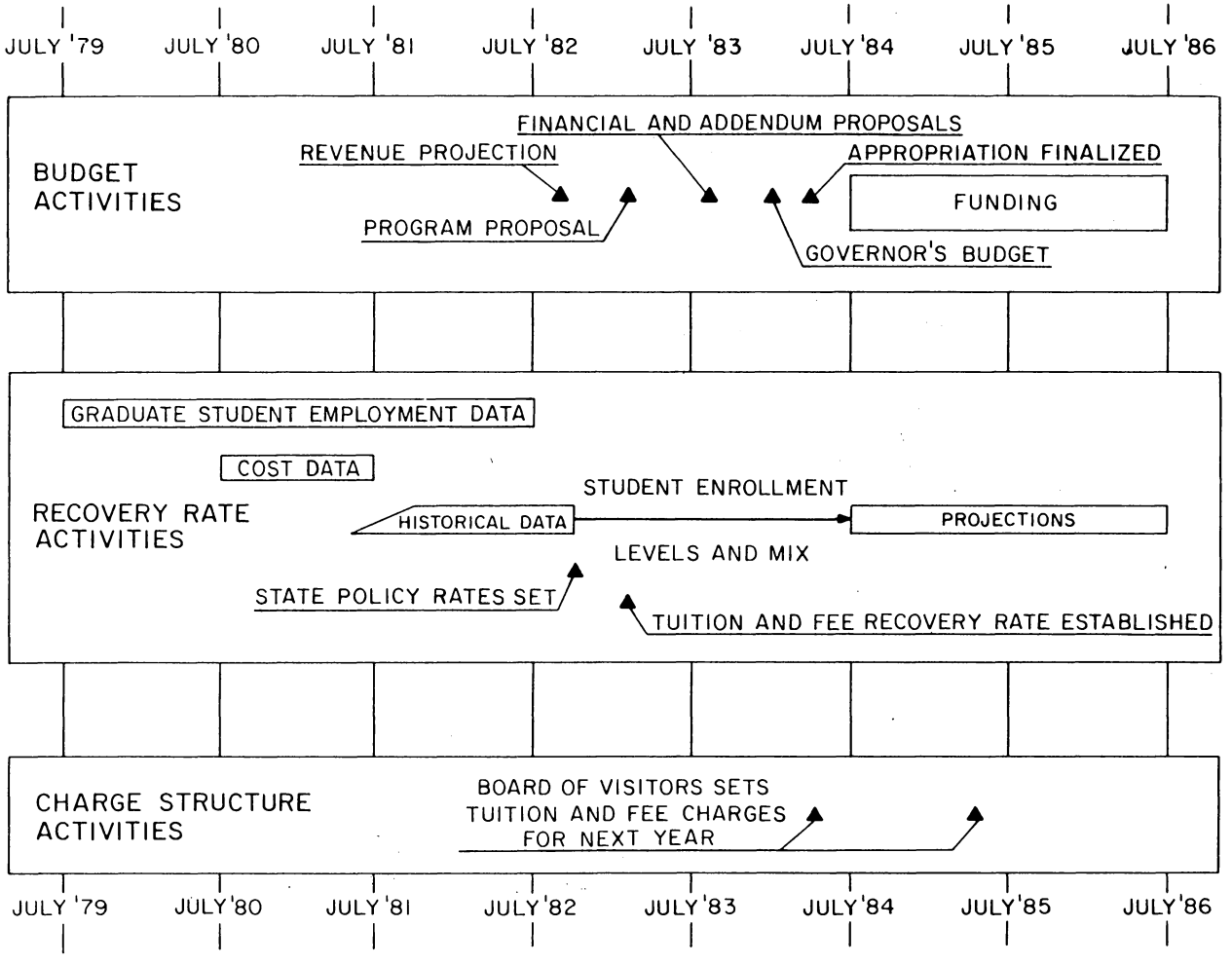


Figure 2.5: Timing of key activities in the tuition and fee process.

institution. The institution responded to the target by submitting the Financial and Addendum Proposals which were again reviewed by the Executive Branch; the Governor's recommendation was presented to the General Assembly in January 1984. The Legislature reviews and alters the Governor's Budget until an agreement is reached and a final appropriation is determined. Actual expenditure against this funding will occur between July 1984 and June 1986. The budget cycle is such that the university is always involved in the preparation and analysis of budgets for two different biennia.

2.2.2 Recovery rate activities

A certain proportion of the revenue that is required to meet the budgeted expenses must be derived from tuition and fee income. The second step in the tuition and fee process involves the determination of this proportion. In Virginia it is determined by a state-defined formula, the tuition and fee recovery rate. As discussed earlier, the recovery-rate formula is a function of institutional-specific and state-directed variables and is primarily dependent upon enrollment, cost, and state policy.

Key activities pertaining to this step in the process are shown in the middle section of Figure 2.5, labeled

Recovery Rate Activities. The tuition and fee recovery rate used in the 1984-86 biennium budget was established in early 1983 and the values used in the formula were from both historical data (from as early as the 1979-80 academic year) and from future projections for the 1984-86 Biennium. As illustrated in Figure 2.5, the level of graduate student employment was based on 1979-82 data; the cost ratios were based on 1980-81 reported cost data; and student enrollment level (number of students) and mix (academic level and residency status) estimates for 1984-86 were based on actual enrollment data through Fall Quarter 1983.

2.2.3 Establishing the tuition and fee charge structure

The amount of tuition and fee revenue that must be generated by the institution in a given year is determined by multiplying the results of the first two steps of the tuition and fee process - planned operating expenses and recovery rate. This is the amount of the total annual cost of operating the university that must be paid by the students. It subsequently must be allocated to different student categories and combined with other university charges to form the tuition and fee charge structure.

The final step in the tuition and fee process is the establishment of the charge structure by the Board of

Visitors of each institution. The Board sets three types of charges - general tuition, student fees, and room and board - for various student categories: undergraduate and graduate, resident and non-resident, full-time and part-time, on-campus and off-campus, regular session and summer session. As shown in the bottom portion of Figure 2.5, the tuition and fee charges for the 1984-85 academic year was set in the Spring of 1984; the charges for 1985-86 will be set in the Spring of 1985.

The charge structure is established, as shown by Figure 2.6, by considering fiscal needs, state requirements, anticipated enrollments, and the university's philosophy and policy concerning differential charges between various student categories. One of the purposes of this research is the development of a mathematical model that allocates the institution's revenue requirements, in each year, to the various student categories based on these considerations. This process is illustrated in Figure 2.6 by the box labelled "goal programming model". The allocation to the various student categories must consider other charges to the student (i.e. health service, student center, room and board) and how the "total" cost compares to other institutions in similar student markets. In addition, it must consider the difference in charges between successive

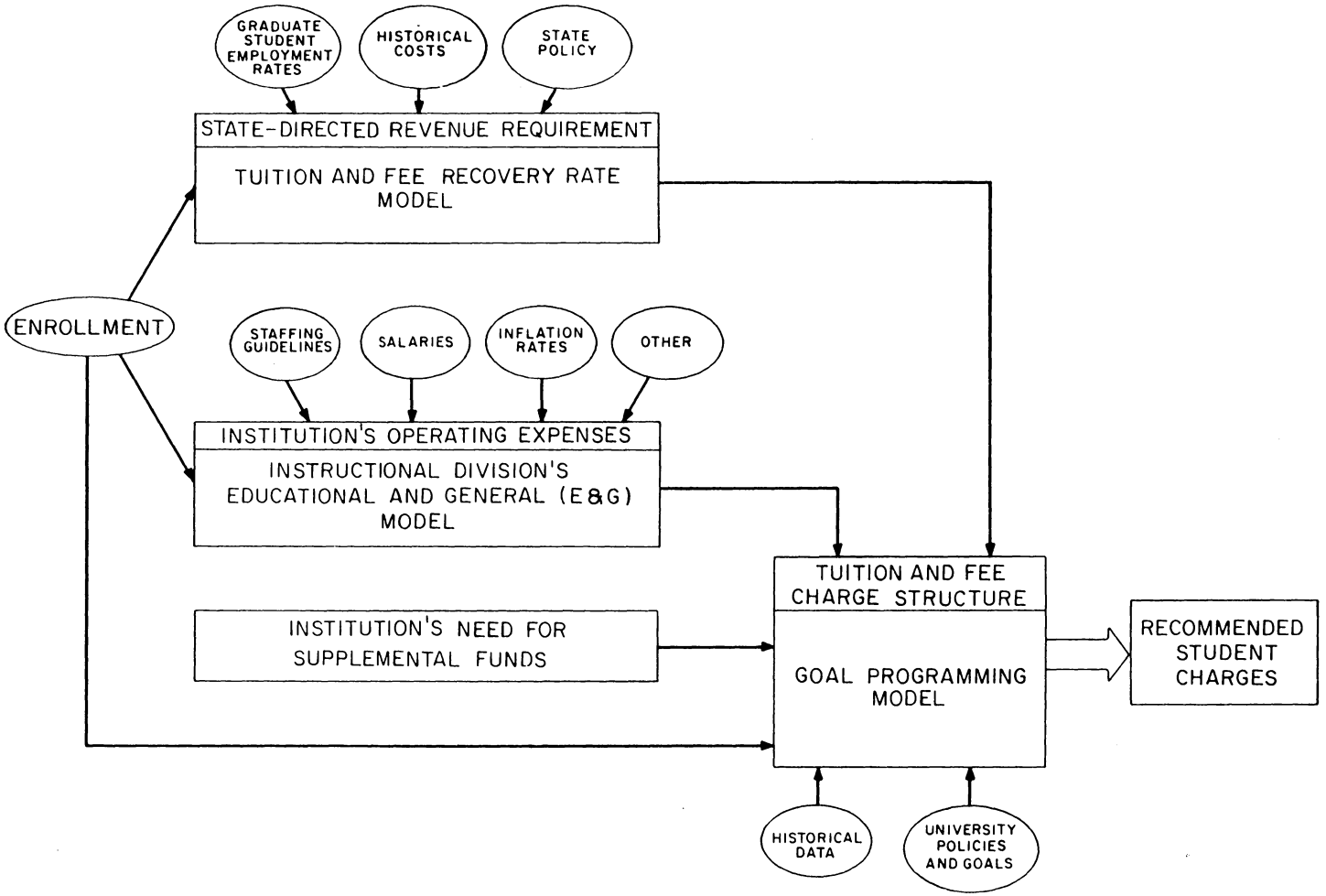


Figure 2.6: Establishing the tuition and fee charge structure.

years and the differences between student categories, either in terms of absolute dollars or percentages. This leads to the second purpose of this research - the integration of the goal programming model into a comprehensive system that not only facilitates the process of setting tuition charges, but provides a means to analyze and test alternative policies under various scenarios.

2.3 STUDENT CHARGES

This section defines the three types of student charges that are used in this research and their role in establishing the tuition charge structure. The terminology and definitions associated with student charges are based on the Virginia Polytechnic Institute and State University system since no "universal" student charges vocabulary exists.

The terminology associated with student charges varies widely from institution to institution. For example, Saupe and Blagg [78] found in a survey of twenty universities that the "basic, comprehensive, multipurpose, undesignated as-to-any-specific-purpose charge" was referred to as tuition at only nine universities. Six of the institutions used the term tuition for an additional charge that was assessed to non-resident students and five did not use the term tuition at all in their statement of student charges [78, p.59].

For the purpose of this research, the following terms will be used:

1. General tuition - the portion of the institution's instruction-oriented operating expenses that is charged to the students.
2. Student fees - all other required charges, such as athletic fee, bus fee, student center fee, and health service fee.
3. Dorm and Dining - charges to students who room in University residence halls and board in University dining halls.

A further definition of these terms will be provided below. Other student costs, such as books, transportation, living and dining in non-University facilities, etc. are not included. Also all miscellaneous charges and fees for admission, graduation, special courses, etc. are excluded.

Table 2.1 summarizes the components of student charges considered in this research - general tuition, four types of student fees (athletic, bus, student center, and health services) and dorm and dining. Table 2.1 also indicates how the specific charge is affected by student category. Basically, general tuition is different for each student category, fees are fixed (the same for each student category), and dorm and dining is an optional charge

TABLE 2.1

Components of student charges and their relationship to student category.

Student Category	Components of Total Student Charges					
		Fees				
	General Tuition	Athl	Bus	St Cntr	Hlth Serv	Dorm & Dining
Full-time students						
Undergraduate						
Resident	V	F	F	F	F	O
Non-Resident	V	F	F	F	F	O
Graduate: On-Campus						
Resident	V	F	F	F	F	O
Non-Resident	V	F	F	F	F	O
Graduate: Off-Campus						
Resident	V	-	-	-	-	-
Non-Resident	V	-	-	-	-	-
Part-time students						
Undergraduate						
Resident	V	-	-	F	O	O
Non-Resident	V	-	-	F	O	O
Graduate: On-Campus						
Resident	V	-	-	F	O	O
Non-Resident	V	-	-	F	O	O
Graduate: Off-Campus						
Resident	V	-	-	-	-	-
Non-Resident	V	-	-	-	-	-

V = Variable charge according to student category.
F = Fixed charge, same for all student categories.
O = Optional charge.
- = Charge does not apply (is not assessed) to that student category.

Note: Student Center and Health Service Fees are different for full-time and part-time students; Room and Board charges are different for graduate and undergraduate students.

depending upon whether or not the student lives in university housing.

The main emphasis of this research is on setting the general tuition charges; but student fees, and dorm and dining charges are included in the model since they influence the amount of general tuition that is allocated to each student category. For example, consider the case of a university that is trying to remain competitive, costwise, for in-state students but has relatively high room-and-board charges. One alternative for the institution is to allocate less of its general tuition charge to in-state students, thereby remaining competitive in terms of total student charges.

2.3.1 General Tuition

All student categories are assessed a general tuition charge. It is the student's payment for use of the educational facilities and services provided by the university. The charge varies by student classification; i.e., non-resident students are expected to pay more than in-state students and graduate students are expected to pay more than undergraduate students. General tuition is the

revenue that is transfered to the state government in order to offset the university's operating expenses. The university must set its charge structure so that it generates enough tuition revenue to meet the state's requirement.

The revenue requirement is allocated to charges for each student category based on academic level, residency status, location, and the number of credit hours taken during the academic session. Presently, at Virginia Tech, there is a different per-credit-hour general tuition charge for each of the following student categories.

1. resident undergraduate, on-campus (regular and summer session)
2. non-resident undergraduate, on-campus (regular and summer session)
3. resident graduate, on-campus (regular and summer session)
4. non-resident graduate, on-campus (regular and summer session)
5. resident graduate, off-campus (regular and summer session)
6. non-resident graduate, off-campus (regular and summer session)

There are a variety of functions that are used to describe the relationship between the number of credit hours for which a student enrolls and the amount of general tuition charge assessed. Saupe and Blagg [78] identified seven different functions that were in use at the twenty universities that they surveyed. Some universities charge strictly by the number of credit hours enrolled (per-credit-hour function); others charge a flat rate, regardless of the number of credit hours (step or plateau function). The other functions that are used are combinations of the per-credit-hour, step, and plateau functions.

The most common, in the Saupe and Blagg survey [78, p. 61] and the one that is used in this research, is the "per-credit-hour-to-plateau" function. Charges are assessed at a per-credit-hour rate up to a specified number of hours, after which the charge is the same regardless of the number of hours enrolled (plateau). The point at which the plateau begins (definition of full-time status) and the slope or dollar rate per credit hour varies by student category.

The functional form for a typical student category is illustrated in Figure 2.7 . For the regular session, the function is linear between three and ten credit hours for undergraduate students (nine hours for graduate students). Therefore, part-time students pay either the minimum

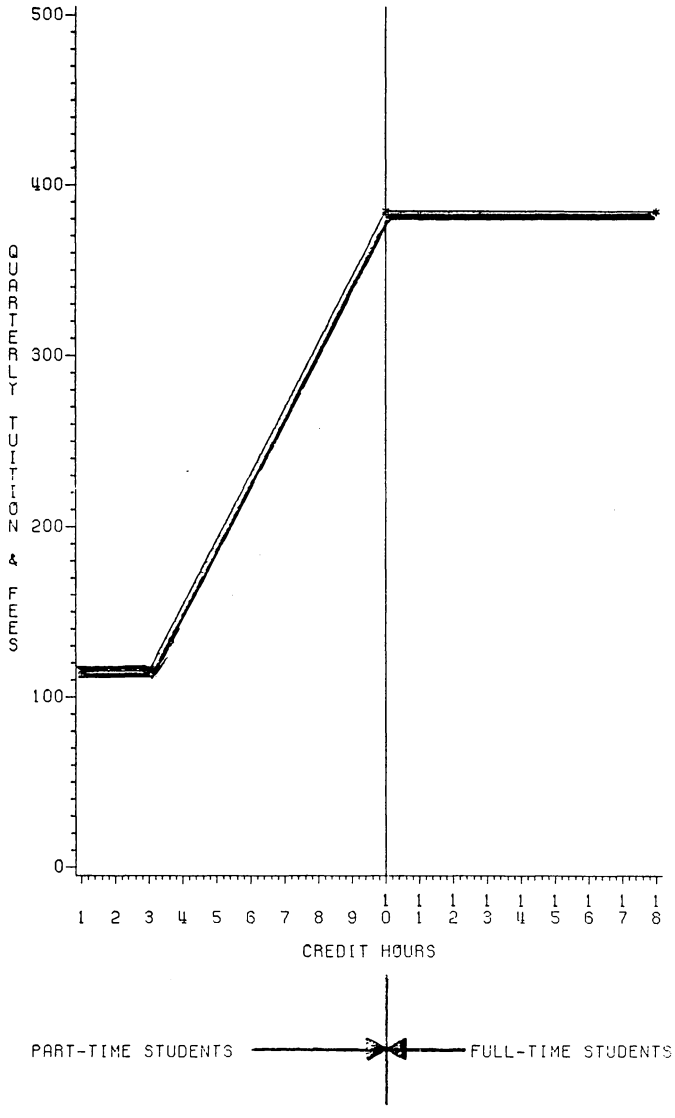


Figure 2.7: The "per-credit-hour-to-plateau" tuition charge function, for a typical student category.

quarterly charge (equivalent to the three credit-hour rate) or by the credit hour. Full-time undergraduate students pay a flat rate equivalent to ten credit hours (nine for graduate students) regardless of the actual number of hours for which they are enrolled.

2.3.2 Student Fees

The second type of charge, student fees, is for student services and activities. These charges are based on planned operational, maintenance, and debt service expenditures. Fees vary widely from institution to institution, in terms of definition and amount. Again the fee categories defined below are those that are currently in effect at Virginia Polytechnic Institute and State University.

The health service fee, for medical and nursing attention, is required of all full-time on-campus undergraduate and graduate students. There is one quarterly charge regardless of student category. The service is optional for part-time students.

The student center fee is required of all on-campus students. Presently there are two charges, one for students enrolled for seven or more hours, another for students enrolled for six or less hours.

The athletic fee, for tickets to all in-season home athletic events, is a required charge for all full-time on-campus students.

The payment of a bus fee is required of all full-time on-campus students for the unlimited use of the city transit system.

2.3.3 Dorm and Dining Charges

Average room and board charges for the different dormitory groups will be used to set the basic tuition charge. The charges are based on planned operational, maintenance, and debt service expenditures. Presently, there are four different quarterly residence hall charges (dorm groups) for undergraduate students, depending upon the type of dormitory, and two graduate student housing charges.

2.4 SUMMARY

Demographic changes, fiscal pressures, and political influences were identified as the primary factors that influence the tuition and fee process. A discussion of each factor and its effect on the tuition and fee process was discussed in this chapter.

The process of establishing tuition and fee charges at colleges and universities was also presented and discussed.

The process was defined in terms of three steps; the first two steps involve determining the institution's planned operating expenses and determining the proportion of those expenses that must be offset by tuition revenue. The third step, and the primary focus of this research, is allocating the annual revenue requirements to specific student categories, i.e., setting the tuition and fee charges by student category for a specified planning period.

Chapter III

THE GP MODEL COMPONENT OF THE DSS

The heart of the decision support system to establish tuition and fees is a goal programming (GP) model that is used to allocate the institution's annual revenue requirements to various student categories. The objectives of this chapter are to: (1) discuss the modeling of the tuition and fee allocation process using GP, (2) present the general GP model by: defining the decision variables, formulating the goal functions, and describing the priority structure and achievement function, and (3) present the methodology employed by the DSS to solve the GP model.

3.1 MODELING THE TUITION AND FEE ALLOCATION PROCESS

Tuition charges are established, as shown in Figure 3.1, by allocating the institution's total annual revenue requirement to tuition charges for each individual student category. The total annual revenue requirement, shown at the top of Figure 3.1, is composed of two components - the state revenue requirement and the supplemental revenue requirement. Recall that the state revenue requirement is the product of the institution's expected annual operating expenses and the state-defined proportion of operating

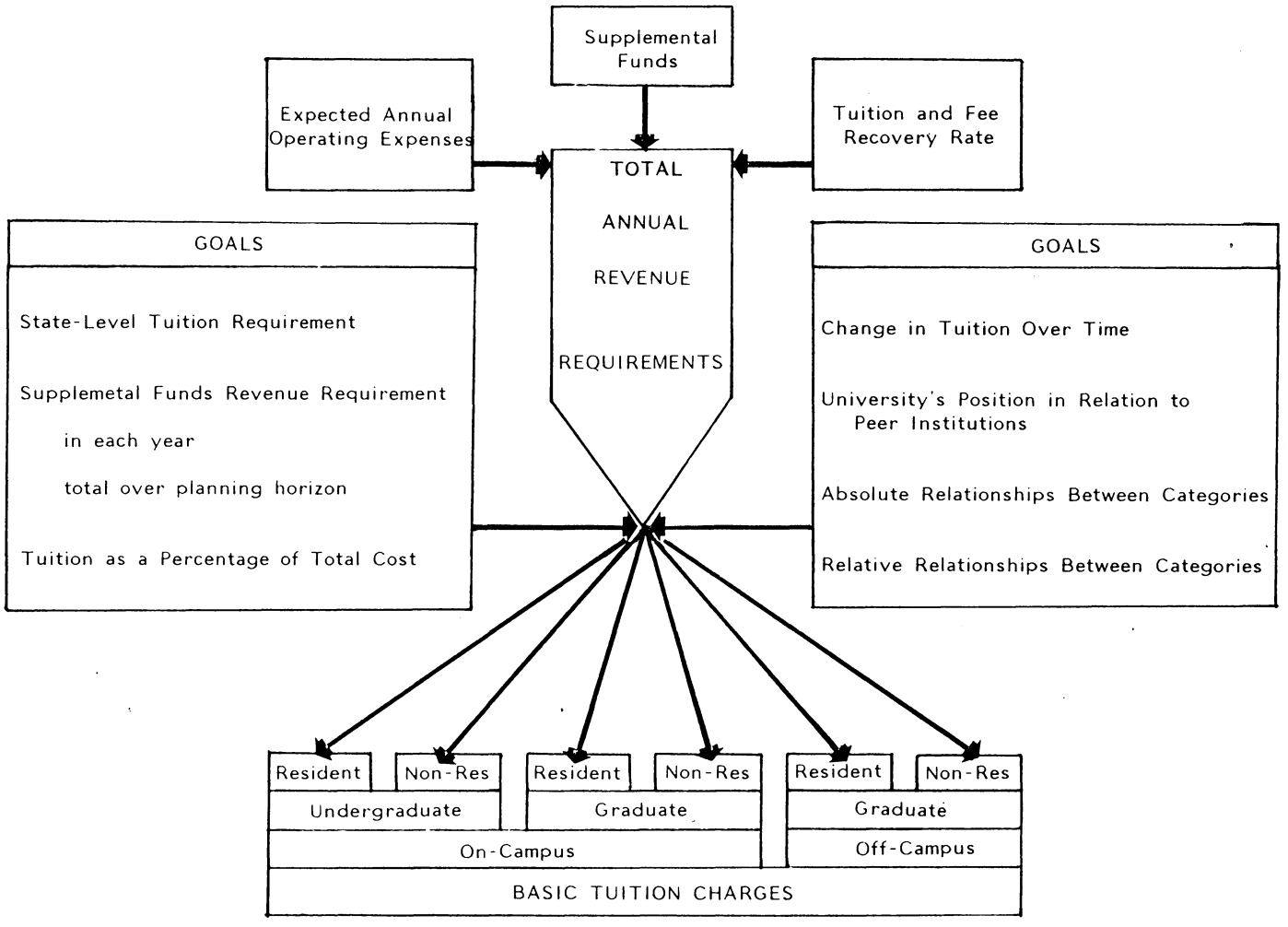


Figure 3.1: Allocating revenue requirements to student categories.

expenses that must be derived from tuition and fee revenue (the tuition and fee recovery rate). Once the total revenue requirement has been established, it must be allocated to the different student categories based on a set of university objectives, constraints, and priorities.

The result of the process, as shown at the bottom of Figure 3.1, is a set of tuition charges for all of the student categories - called the tuition charge structure. Each student category is assessed a different tuition charge. The distinction between student categories is based on academic level, residency status, campus location, etc.

The university's revenue requirements are allocated to various student categories based on a set of objectives. As shown in the middle of Figure 3.1, these objectives, hereafter called goals, include: meeting the state requirements, remaining competitive with peer institutions, maintaining a specific difference in charges between categories, etc.

Usually all of the goals cannot be achieved because of the limitations imposed by resource constraints, policy directives, political influences, past or standard practices, the nature of the competitive market, and the fact that often times the objectives themselves are conflicting. Since it is usually not possible to establish

a tuition charge structure that meets all of the goals, the university must priortize the goals and settle for a charge structure that "best" reflects their goals, limitations, and priorities. The resulting tuition charge structure is therefore a compromise between the institution's financial considerations and its philosophical concerns.

A goal programming (GP) model is used to establish the university's tuition charge structure by allocating annual revenue requirements to various student categories. GP is a mathematical programming technique that allows for the consideration of multiple incommensurable and conflicting objectives, called goals. It is especially useful for solving problems that contain multiple goals and multiple subgoals. The multiple goals are incorporated through a multidimensional objective function, referred to as an achievement function. Goal programming attempts to "satisfy" a set of rank-ordered objectives, rather than "optimize" a single objective, as in linear programming.

The algorithm minimizes the deviation from each goal's target value in the order of importance to the decision maker (a preemptive priority structure). The value of the decision variables in a GP solution (in this case, tuition charge for each student category) is the one that provides the best combination of goal achievements for a given

priority structure. All of the goals are usually not achieved in the final solution because GP only considers low-order goals after the higher-ordered goals have been reached or are at a level where no further improvement is possible. In other words, many goals are achieved only at the expense of other goals.

Since the topic of goal programming has been previously described in detail, [37], [38], [40], [56], [86,], and [100], a general discussion of this topic will not be presented.

The general GP model of the tuition allocation process is presented in the following three sections:
(1) definition of the decision variables, (2) formulation of the goals, and (3) description of the priority structure and achievement function.

3.2 THE DECISION VARIABLES

Each decision variable is defined as the quarterly charge per credit hour for a given student category. Since revenue requirements are allocated to each of the various student categories, it follows that each decision variable represent a student category. A student category is defined as the combination of the classifications listed below; included in the following list of possible student

categories are the set of assumptions that were made for each classification in order to limit the size of the problem and to reflect the current policies of the institution being modeled.

1. Academic level of instruction - Two levels of instruction are considered: undergraduate versus graduate.
2. Academic discipline, program, or major - There is no differential charge for discipline, program, or major.
3. Residency status - Two residency types are considered: state resident versus non-resident.
4. Session - No session differential is included; regular session and summer session charges are based on the same rate (same dollars per credit hour).
5. Location - A different charge is considered for two locations: on-campus (main campus) versus off-campus (satellite locations).
6. Participation status - Although two levels of participation are considered, full-time versus part-time, separate decision variables are not necessary since it is assumed that both are based on the same credit-hour rate.

7. Academic year - a planning horizon of three academic years is used.

The use of the most general set of student categories - the combination of all of the student classifications defined above - results in a sizeable model formulation. For example, if each of the seven classifications had only two levels, then there would be 128 different combinations or student categories ($2^7 = 128$), resulting in 128 decision variables. Most universities would find this general formulation and the resulting tuition charge structure too tedious and cumbersome to manage.

A variety of functions are used to describe the relationship between the number of student credit hours and the amount of general tuition charge assessed. Saupe and Blagg [78] identified seven different functions that were in use at the twenty universities that they surveyed. Some universities charge strictly by the number of credit hours enrolled (per-credit-hour function); others charge a flat rate, regardless of the number of credit hours (step or plateau function). The other functions that are used are combinations of the per-credit-hour, step, and plateau functions. The most common, in the Saupe and Blagg survey [78] and the one used in this research, is the "per-credit-hour-to-plateau function".

Since the tuition charge function being modeled is a "per-credit-to-plateau" function, the decision variables are defined in terms of a rate, dollars per credit hour. Both full-time and part-time students are charged the same rate 'r', thereby eliminating the need to include participation status (full-time and part-time) as a category in the decision variable definition. Charges are assessed at 'r' dollars per credit hour up to n-1 credits and at $(n*r) for n or more credits, where 'n' is the number of credit hours that must be carried in order to be considered a full-time student.

The decision variables are defined as :

X_{hijk} = quarterly general tuition charge per credit hour for a student in category hij, in year k.

where h = 1,2 (academic level: undergraduate, graduate)

i = 1,2 (residency status: resident, non-resident)

j = 1,2 (location: on-campus, off-campus)

k = 1,2, ... ,KK (academic year)

For example, X_{1213} is the tuition charge per quarter hour for an undergraduate (h=1) non-resident student (i=2) enrolled on-campus (j=1) in the third year (k=3) of the planning horizon.

3.3 MATHEMATICAL FORMULATION OF THE GOALS

The goals that comprise the GP model are descriptions of the university's tuition philosophy and policy, expressed in mathematical form. Each goal is formulated as a mathematical relationship between the decision variables and a target, as shown below:

$$f_g(X_{hijk}) + n_g - p_g = b_g$$

where f_g is the linear functional relationship, in terms of the decision variables, that describes the g^{th} goal. Also, n_g and p_g are deviation variables, where n_g represents a negative deviation or under-attainment of the g^{th} goal, and p_g is a positive deviation variable or over-attainment of the g^{th} goal. The right-hand side, b_g, is the target that the goal function is attempting to satisfy. Each goal is linked to the specified target by the deviation variable and, depending on the problem, each goal may be exactly achieved, exceeded, or underachieved relative to the specified target.

All of the goals that are used to set the tuition charge structure have been identified and formulated. The formulations presented in the following sections include, by type:

1. Aggregate Goals
   a) State-level tuition revenue requirement.
   b) Supplemental funds revenue requirement.
2. Individual Goals
   a) Tuition charges as a percent of average total cost.
   b) Change in tuition charges over time.
   c) University position in relation to peer institutions.
   d) General individual goal.
3. Relative Goals
4. Absolute Goals
   a) Single-category
   b) Pairwise comparison
   c) Four-category

Each goal type is discussed below and presented in general form. Since the formulations presented below are general and the solution of a specific problem may not include all of the goals, the deviation variables are indicated, in general terms, as n_g and p_g, for all goal types.

3.3.1 Aggregate Goals

Aggregate goals are defined as those that include all of the student charge categories in the goal formulation. There is a separate goal equation for each year in the planning horizon. There may also be a single goal that spans the entire planning horizon.

3.3.1.1 State-level tuition revenue requirement

Many states require that each state-supported academic institution generate enough tuition revenue to offset a specific proportion of its expected annual operating costs. Thus the goal, for year K, the goal is formulated as:

$$\sum_{h=1}^{2}\sum_{i=1}^{2}\sum_{j=1}^{2}[CH_{hijK}*X_{hijK}] + n_g - p_g = REVENUE_K \quad (3.1)$$

The parameter CH_{hijK} is the expected total number of "fee" credit hours[5] taken by all students in category combination HIJ, in year K. For example, CH_{1212} is the expected number of on-campus (j=1) out-of-state (i=2) undergraduate (h=1) credit hours in year 2 (k=2) of the planning period. The goal level value, $REVENUE_K$, considers both the expected

---

[5] Since a "per-credit-hour-to-plateau" tuition charge function is assumed, part-time students are charged only for the number of hours in which they are enrolled; whereas, full-time students are charged a flat rate regardless of the number of hours enrolled.

annual operating cost for the university in year K and the proportion of operating expenses that must be offset by tuition and fees in year K. To avoid numerical problems in the solution routine, it is necessary to scale some of the parameters; therefore, CH_{hijK} is expressed in thousands of credit hours and $REVENUE_K$ in thousands of dollars.

The institution must minimize the negative deviation variable n_g (set it to zero) to at least meet the state requirement. The institution would usually want to also minimize p_g so that it does not collect more revenue than it needs; but, under certain circumstances it may want to raise additional revenue to generate supplemental funds for salaries, equipment, etc. This goal is considered in the next section.

3.3.1.2 Supplemental funds revenue requirement (Yearly)

There are two components of the supplemental revenue goal - yearly and overall. The yearly component represents the desire to generate a certain amount of additional revenue in <u>each</u> year of the planning horizon; the overall component represents the total amount of revenue that is desired, in total, by the end of the planning horizon. For example, a university may want to generate at least $500,000 in additional revenue in <u>each</u> of three years, with the <u>total</u> at the end of the planning period being at least $2,000,000.

The yearly component is formulated in Equation (3.2).

$$p_g + n_g^* - p_g^* = SUPFUND_K \qquad (3.2)$$

p_g is the positive deviation variable from Equation (3.1), n_g^* and p_g^* are an additional set of deviation variables, and $SUPFUND_K$ is the desired level of supplemental funds or additional revenue in year K (in thousands of dollars).

3.3.1.3 Supplemental funds revenue requirement (Overall)

The second component of the supplemental revenue goal considers the total amount of funds desired over the entire span of the planning horizon; it is formulated as:

$$\sum_{g=1}^{MM} p_g + n_g^{**} - p_g^{**} = SUPFUND \qquad (3.3)$$

p_g is again the positive deviation variable from goal Equation (3.1), n_g^{**} and p_g^{**} are another set of deviation variables, and SUPFUND is the desired level of supplemental funds to be generated over the entire planning horizon (in thousands of dollars).

3.3.2 Individual Goals

Individual goals are used to consider individual student categories, apart from the others. There is a separate goal equation for each student category in each year.

3.3.2.1 Tuition charges as a percentage of average total cost

One way to differentiate tuition charges between student categories is to base the charges on a specified proportion of average total cost. Each category is assessed, through tuition charges, a share of the institution's average cost of instruction. For example, it may be desirable to set the tuition charges such that a resident undergraduate pays no more than 30% of the average total cost, a non-resident undergraduate pays no less than 50% of the average cost, etc. Another way of stating this is that a resident undergraduate student should pay no more than 30% of his or her education (the state pays the other 70%), a non-resident undergraduate is expected to pay at least half of the cost of education (the state pays the other half), etc. The goal for year K is expressed below:

$$\frac{PSCH_h * X_{hijK}}{AVGCOST_K} + n_g - p_g = PERCOST_{hijK} \qquad (3.4)$$

The basic per-credit-hour charge for any student category, X_{hijk}, is converted to an annual full-time charge by multiplying it by $PSCH_h$ - the minimum number of credit hours that a student must carry per academic year to be classified as full-time for fee purposes. The proportion of average total cost is obtained by dividing the annual full-time charge for each category by the university's average total cost of instruction per full-time student in year K, $AVGCOST_K$. The target value that each goal is attempting to achieve is specified by $PERCOST_{hijK}$.

3.3.2.2 Change in tuition charges over time

This goal is used to control the rate of change in tuition charges over time for each student category. It is not uncommon for the administration to establish the tuition and fee charge structure so that the charges for the various student categories increase at different rates.

The index often used to increase tuition charges from year to year is the overall expected change in the institution's revenue requirement. But, other measures such as various prices indices, expected change in the institution's instructional costs, expected changes at other institutions, etc. are also used to index the change in tuition charges for the various student categories. For

example, if the overall revenue requirement for the institution is increasing faster than the CPI (Consumer Price Index), it may be desirable to limit the increase in resident student charges to the change in the CPI and let the non-resident students absorb the difference required to meet overall revenue requirements.

For any student category, hij, and for any measure of change between two years, the goal would be formulated as:

$$\frac{X_{hijK}}{X_{hij(K-1)}} = MEAS_{hij}[K,K-1]$$

The intent is to set the ratio between the tuition charge in year K, X_{hijK} , and the previous year's charge, $X_{hij(K-1)}$, equal some measure of change.

The goal, rearranged in a more convenient form, is expressed as:

$$X_{hijK} - MEAS_{hij}[K,K-1]*X_{hij(K-1)} + n_g - p_g = 0 \qquad (3.5)$$

3.3.2.3 University's position in relation to peer institutions

In order to remain competitive in different student markets, the university often attempts to maintain a certain tuition position relative to similar schools within its peer group. The way that peer-institution goals are used will

differ by student category. For example, in setting tuition charges for in-state students, the university may be concerned with the average total charge for full-time students at other institutions within the state. For out-of-state undergraduates, the university may compare its charges to other Land-Grant colleges and universities, charges at neighboring state universities, etc. If most of the off-campus students are part-time, then it would be more appropriate to consider charges for a single course, rather than full-time charges, at institutions near the off-campus location(s). As another example, consider a college that has relatively high room-and-board costs. In order to be competitive with other schools within the state it must keep its total charges at a certain level. One way to accomplish this is to allocate less of the basic tuition charges to the in-state student category.

The general peer institution goal, formulated on an annual basis, is expressed as:

$$PSCH_h * X_{hijk} + OCHG_{hijk} + n_g - p_g = PEER_{hijk} \quad (3.6)$$

$PSCH_h$, as previously defined, converts the tuition charge from a per-credit-hour basis to an annual full-time basis. All other applicable charges, i.e. health service fee, student center fee, athletic fee, room and board, are

included in the parameter $OCHG_{hijk}$. The goal level, $PEER_{hijk}$, is a measure of the total cost for student category hij, in year k, for a group of comparable institutions. It may be, for example, the mean of the in-state charges at all of the colleges and universities in the state, the third-quartile of the out-of-state charges for all Land-Grant institutions, etc.

As indicated earlier, sometimes the total annual cost for a full-time student is not the most appropriate measure. For example, if most of the off-campus students are part-time, then it may be better to consider charges for a single three-credit course at institutions near the off-campus location(s). For a three-credit course, $PSCH_h=3$ in Equation (3.6).

3.3.2.4 General individual goals

The direct specification of a particular student category's per-credit-hour tuition charge is made through a General Individual Goal (GIG). A particular value, a lower limit, or an upper limit can be included for each student category in each year of the planning horizon. For example, a goal may be included in order to limit the resident undergraduate charge to no more than $60 per credit hour, or set the non-resident undergraduate charge to a level of at least $100 per credit hour, etc.

The DSS automatically includes a GIG for each student category in the model formulation. It will not influence the solution if the goal target value is set to 0 (the default value) or if the priority level is set to 0 (also the default value). GIGs are basically the same as single-category absolute goals, which are defined in a later section. That section includes a detailed discussion of how the GIGs can be used in conjunction with single-category absolute goals.

The general individual goal is formulated as follows:

$$X_{hijk} + n_g - p_g = VALUE_{hijk} \quad (3.7)$$

3.3.3 Relative Goals

Tuition charges may be established by considering the ratio between any two student categories. For example, the university may set its tuition charge structure so that non-resident undergraduates pay at least twice the in-state rate, resident graduate students pay at least 25% more than resident undergraduates, etc.

In general, the goal to achieve the c^{th} targeted relative difference, $RATIO_k^c[hij, hij^*]$, between any two decision variables, X_{hijk} and X^*_{hijk}, in year k, is expressed as:

$$\frac{X_{hijk}}{X^{*}_{hijk}} = RATIO^{c}_{k}[hij, hij^{*}], \quad \text{with } hij \neq hij^{*}$$

or expressed in a more convenient form:

$$X_{ijk} - RATIO^{c}_{k}[hij, hij^{*}]*X^{*}_{hijk} + n_g - p_g = 0 \quad (3.8)$$

3.3.4 Absolute Goals

In establishing the tuition charge structure it is often desirable to specify a goal in absolute terms, dollars per credit hour, either directly or as a comparative difference. The model includes three types of absolute goals: (1) single-category, (2) pairwise comparison, and (3) four-category.

3.3.4.1 Single-category absolute goals

A single-category absolute goal (SCAG) is used to directly specify a particular student category's charge, its lower limit, or its upper limit. Single-category absolute goals are used in the same way as General Individual Goals (GIGs); they provide an additional source of goal statements if needed.

Their inclusion in the model is actually related to how the DSS is structured. The user explicitly selects which

SCAGs to include whereas, a GIG is automatically included in the formulation. An example will clarify how the two types of goals are expected to be used.

Suppose the user has found an acceptable tuition charge structure (solution) by including no GIGs and only one SCAG in the problem formulation, e.g. maintain non-resident graduate student charges below $75 per credit hour. Also assume, as a matter of university policy, that the charges must be rounded (to the nearest $1.00, nearest $0.50, etc.) before they can implemented. To be sure that the implemented charge structure does not differ greatly from the GP model solution and that the proper amount of revenue is generated, the model could be rerun, constraining the student charges for certain categories to their rounded value. This would probably be done sequentially, with the most influential student category constrained first. The post-solution rounding constraints could easily be included in the problem by utilizing a General Individual Goal constraint. A GIG for each student category has automatically been included in the model, with a goal target value of 0, even if it was not directly specified or used (a design feature of the DSS). To force the particular charge to its rounded value, the user would simply: (1) change the appropriate GIG target value from 0 to the rounded value,

(2) set that goal's priority to one, and (3) re-solve the model.

The general formulation for a single-category goal is given in Equation (3.9). It is the same as the General Individual Goal previously formulated in Equation (3.7).

$$X_{hijk} + n_g - p_g = VALUE_{hijk} \qquad (3.9)$$

3.3.4.2 Pairwise comparison absolute goals

This goal is used to specify dollar per credit-hour differences between pairs of student categories. For example, the tuition charge structure may be set with the intent of maintaining at least a $10 per credit-hour differential between on-campus and off-campus programs, maintaining the difference between resident and non-resident graduate students at between $10 and $15 per credit hour, etc. There is a separate goal for each year in the planning horizon.

The general formulation for a per-credit-hour difference between two categories is given as:

$$X_{hijk} - X^{*}_{hijk} + n_g - p_g = DIFF^{a}_{k}[hij, hij^{*}] \qquad (3.10)$$
$$\text{where } hij \neq hij^{*}$$

X_{hijk} and X^{*}_{hijk} are any two different decision variables and the goal level, $DIFF^{a}_{k}[hij, hij^{*}]$, is the a^{th} desired per-credit-hour difference in year K.

3.3.4.3 Four-category absolute goals

This goal, while not especially intuitive, is used to model a situation where four student categories are simultaneously involved in a goal formulation. For example, this goal type is used to maintain the same difference in tuition charges between resident and non-resident graduate students in both the on-campus and off-campus programs (four student categories). There is a separate goal for each year in the planning horizon.

The four-category absolute goal, on a per credit-hour basis, is formulated in Equation (3.11).

$$(X_{hijk} - X^{*}_{hijk}) - (X^{**}_{hijk} - X^{***}_{hijk}) + n_g - p_g = VALUE_k \qquad (3.11)$$

$$\text{where } hij \neq hij^{*} \neq hij^{**} \neq hij^{***}$$

3.4 THE PRIORITY STRUCTURE AND ACHIEVEMENT FUNCTION

There is no general priority structure or corresponding achievement function since they are both problem-specific. The university administration's decision as to which goals to include and the order in which they should be ranked depends on the economic, political, and demographic climate at the time of the analysis.

3.4.1 Priority Structure

The priority structure is a reflection of the institution's philosophy and policies on how tuition charges should be apportioned to the various student categories given a set of economic, political, and demographic conditions. The priority structure is established in two steps: first, decide which goals, described in the previous section, to use in order to set the tuition and fee charge structure; and second, rank the goals in order of importance to the university.

Each goal that is included in the model is ranked and assigned to a corresponding priority level. The importance of fully satisfying each goal is reflected in its ranking - goals of lower rank are satisfied only after the higher ranking goals are satisfied or further improvement is not possible. Several goals may be included at the same priority level as long as they are commensurable (have the same units of measure).

Goals may be given different emphasis within the same priority level by using subgoal weights - numerical weights that reflect the relative importance of the deviation variables within the same priority level. The use of subgoals is not that common since it is often difficult to assign the appropriate weight. For example, it is often not

clear as to whether a particular goal is twice as important as another, 2.5 times, three times, etc. In most cases if the achievement of one goal is considered to be more important than another, then that goal should probably be treated in a preemptive manner - included in another priority level.

3.4.2 Achievement Function

Once the priority structure has been established, the achievement function (multidimensional objective function) is formulated. It contains the deviation variables that are to be minimized, and measures the attainment of the goals at each priority level. Goal programming attempts to minimize the deviation from the desired goal level in a sequential manner. The general form of the achievement function, that is to be minimized, is given below:

$$\bar{a} = \{g_1(\bar{n},\bar{p}), g_2(\bar{n},\bar{p}), \ldots , g_p(\bar{n},\bar{p})\}$$

The achievement function, $\bar{a}$, is represented as a vector since the GP model is formulated using preemptive priorities. It contains a linear function of the deviational variables, $g_p(\bar{n},\bar{p})$, for each of the P priority levels. The $g_p(\bar{n},\bar{p})$ functions are ordered in the achievement function according to their respective priority.

3.5 GP MODEL SOLUTION METHODOLOGY

The methodology used to solve the GP model is presented in three parts: (1) the definition of the general solution approach, (2) a discussion of how the solution routine is integrated into the DSS, and (3) a description of the specific solution routine.

3.5.1 Sequential Linear Goal Programming

The GP model, imbedded in the DSS, is solved using the Sequential Linear Goal Programming (SLGP) method - an approach that treats the GP model as an "interrelated sequence of conventional linear programming problems" [39, p. 142]. The GP model is decomposed, by priority level, into a series of single-objective problems and then solved by a standard commercial single-objective mathematical programming code (using the basic simplex algorithm) and a "user-written buffer code" to control the operations. A detailed description of the SLGP method is presented in [18], [36, p. 65-69], [38, p. 1115-1117], and [39].

Another approach to solving linear GP problems is the Multiphase method. This modified simplex method is an extension of the two-phase method of conventional linear programming and is discussed in detail in [37] and [56]. The Multiphase method was developed especially for linear GP problems and in theory it is preferred to SLGP because:

(1) it approaches the entire GP problem as a single model and does not require additional constraints during solution [38, p. 1116], (2) the Multiphase formulation is predetermined or static , (3) SLGP usually requires more pivots, and (4) duality and sensitivity analysis are more straightforward with the Multiphase formulation [39, p. 143].

Despite its theoretical superiority, the Multiphase method is presently dominated by the SLGP technique in terms of computational effeciency. SLGP's efficiency is the result of its ability to utilize the extremely powerful commercial mathematical programming packages that are available for traditional single-objective models. Computer codes for the Multiphase method have not been developed, as of yet, to a comparable level of sophistication. For other than modest size problems, the rather unsophisticated pivoting operations employed by the Multiphase computer codes leads to round-off problems [38], [39].

3.5.2 Integration of the Solution Routine into the DSS

The solution of the GP models in the DSS uses a modified SLGP approach. The basic SLGP is modified so that the GP model can be solved interactively in one execution. The model is solved sequentially using an existing

interactive commercial computer code, LINDO,[6] and a user-written subroutine that communicates with the LINDO system. A new solution routine was developed for the DSS because an interactive GP solution package is currently not available. MPS (Mathematical Programming System) III and MPS X have been used to solve SLGP problems but they are not as conducive as LINDO to performing interactive operations and for solving the sequence of linear programs in one execution.

Figure 3.2 provides an overview of how the GP solution routine is integrated into the DSS. Once the model has been formulated, the parameter values estimated, the goal target values set, and the priority structure established, then the model is solved interactively and a series of solution reports are generated. This process is automatic and once the user indicates that the model is ready to solve, no further information is required from the user until the solution appears on the terminal's display screen. As shown in Figure 3.2, when the model is ready to be solved control is passed from the Executive Control Program, DSS EXEC, to a Component Control Program, GP EXEC.

---

[6] The package requires 448K on the IBM 4341 Group 2 machine. The maximum size of the input matrix is 499 columns (variables) by 239 rows (objective function plus constraints).

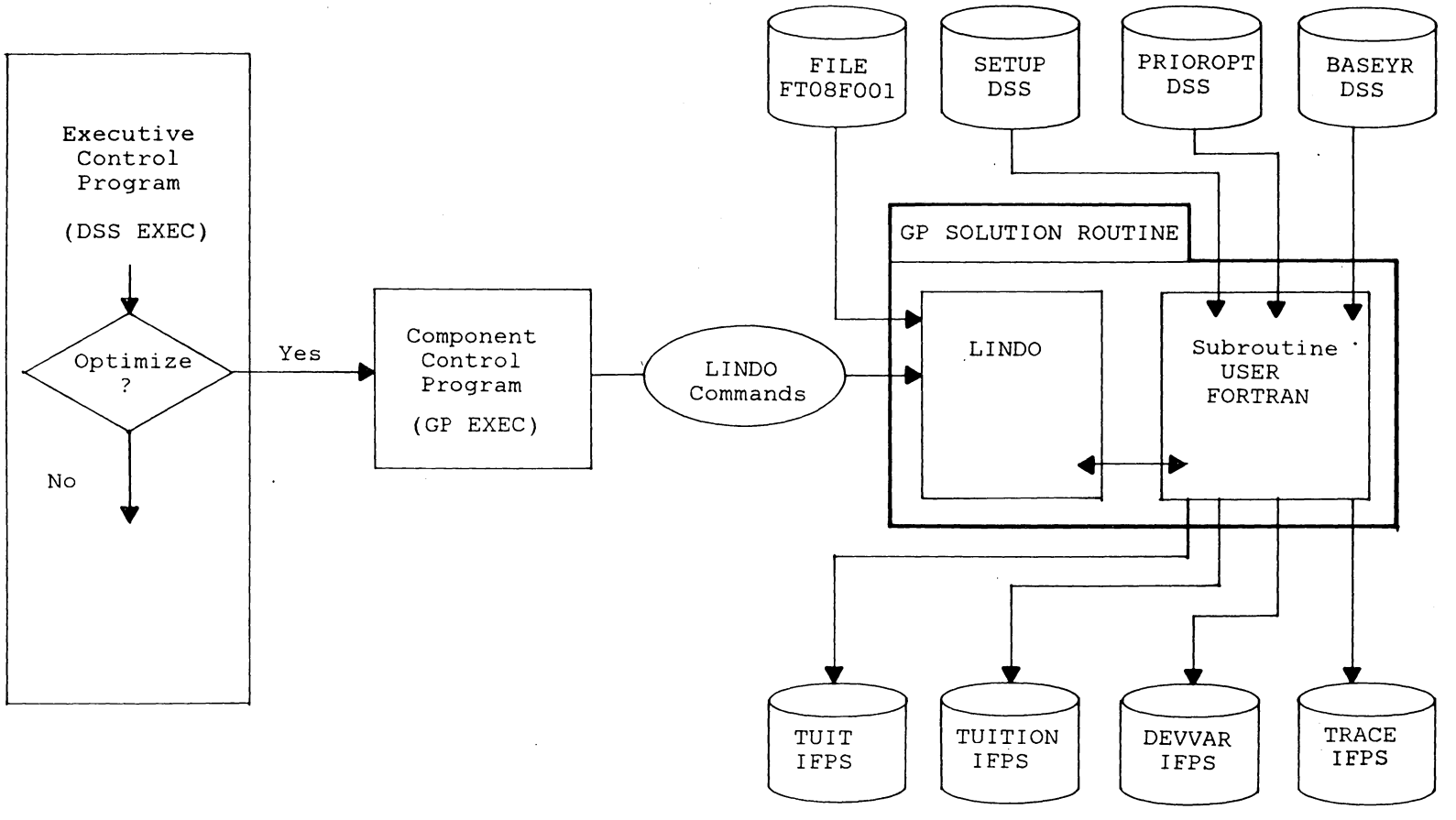


Figure 3.2: Overview of the GP Solution approach used in the DSS.

The Component Control Program GP EXEC has three functions: (1) "stack" a set of LINDO commands, (2) identify the necessary input/output files, and (3) link the USER subroutine with the LINDO system and start execution.

The set of commands that are passed to LINDO are listed below:

```
TERSE
RMPS
&UNO
MIN
USER
QUIT
```

Once the LINDO system is accessed it executes each of these commands in order. TERSE suppresses the printing of the entire solution, RMPS indicates that the problem is stored in a MPS-formatted[7] datafile, and &UNO provides the number of that file. LINDO assumes the problem is stored in a file named FILE FTXX001 where &UNO is the two-digit number XX. There is no way of specifying within MPS format whether the objective function is to be minimized or maximized; the command MIN defines the problem as a minimization. LINDO has a FORTRAN user-interface capability. It contains a dummy subroutine, called USER, for the purpose of accessing and executing special routines while in the LINDO system. The

---

[7] MPS format is a widely accepted format for describing and storing LP models on the computer. Most commercial LP systems recognize this format; it consists mainly of three components: COLUMNS, ROWS, and RHS (right-hand side).

command USER executes the subroutine, which in the case of this application contains the program required to solve the GP model as a sequence of linear programs. The details of this program are discussed in the following section. Once the GP model has been solved by subroutine USER, the command QUIT returns control to the Executive Control Program.

The component control program defines the input/output files required by subroutine USER, as shown in Figure 3.2. The four input files provide: (1) the basic problem definition (SETUP), (2) the base-year tuition charges (BASEYR), (3) the model in MPS format (FILE FT08F001), and (4) the priority structure (PRIOROPT). The origin of the input files are discussed in Chapter 4. The output files contain reports which are discussed in detail in the next section.

3.5.3 The GP Solution Algorithm

The GP model, as shown in Figure 3.2, is solved using both the LINDO system and a user-written subroutine, USER FORTRAN, that interfaces with LINDO.[8] The program listing for the subroutine is provided in Appendix C.21. This

---

[8] As a point of reference, a GP model consisting of 212 variables, 97 goal equations, a density of 1.9%, and 9 priority levels solves in 2.9 seconds on the IBM 4341 processor. The solution routine includes reading 4 data files and writing 4 reports to data files.

section discusses the subroutine's four primary functions: (1) input the problem to be solved, (2) translate the priority structure provided by the user into the achievement function that is used in the solution routine, (3) solve the GP model, and (4) generate output reports used to assess the solution.

3.5.3.1 Inputs to the Subroutine USER

The data required to solve the GP model are contained in four files, as shown in Figure 3.2. One file - FILE FT08F001, which contains the model formulation in MPS format - is not directly used by the subroutine USER. It is used to define the problem that is to be solved by LINDO.

The other three data files directly support the solution process controlled by the subroutine USER. Specifically, data file SETUP defines the framework of the problem that is to be solved (number of goals of each type included, number of student categories, number of years in the planning horizon, etc.), PRIOROPT contains a translation of the priority structure input by the user, and BASEYR contains the base-year tuition charges (used only as supporting information in an output report).

The notation for identifying the decision variables is simplified for use in the computer solution routine. The hij

subscript, that is used to define each student category, is replaced by a single subscript m, m=1,2,...MM; where MM = HH*II*JJ. Recall that for each particular student category, h refers to the academic level, i to the residency status, and j to the campus location.

The translation between the general model notation and that used in the computer routine is provided in Table 3.1 for the six student categories considered in this research. The number of student categories is a variable and is not limited to the six listed in Table 3.1 (those six best reflect the policies of the institution being modeled).

The deviation variables are still defined as Ng and Pg, g=1,2,...,GG; where Ng in the computer routine is n_g - the negative deviation or level of under-attainment of the g^{th} goal and Pg is the positive deviation, p_g in the model, or over-attainment of the g^{th} goal.

There is also a particular predefined structure to the GP model. The model's goal equations are formulated in the following order (by goal type): aggregate goals, individual goals, relative goals, absolute goals (single, pairwise, four-category) and finally overall goals. Table 3.2 provides an example of one model's goal structure and the corresponding numbering scheme for each of the goals' deviation variables. The model illustrated in Table 3.2

TABLE 3.1

Translation of the decision-variable notation from the general model to the computer solution routine.

Redefine $X_{hijk} = X_{mk}$

GENERAL MODEL	COMPUTER PROGRAM	DEFINITION
X_{111k}	X1k	= Resident Undergraduate (On-Campus)
X_{121k}	X2k	= Non-Resident Undergraduate (On-Campus)
X_{211k}	X3k	= Resident Graduate (On-Campus)
X_{221k}	X4k	= Non-Resident Graduate (On-Campus)
X_{212k}	X5k	= Resident Graduate (Off-Campus)
X_{222k}	X6k	= Non-Resident Graduate (Off-Campus)

* for a 3-year planning horizon, k = 1, 2, 3

TABLE 3.2

Representive goal structure and its corresponding numbering scheme for the model's variables.

PROBLEM STRUCTURE

3-YEAR PLANNING HORIZON — 6 STUDENT CATEGORIES

GOALS INCLUDE:

- STATE REVENUE REQUIREMENT
- SUPPLEMENTAL REVENUE REQUIREMENT - YEARLY
- PERCENTAGE OF AVERAGE TOTAL COST GOALS
- CHANGE IN CHARGES OVER TIME GOALS
- PEER INSTITUTION GOALS
- GENERAL INDIVIDUAL GOALS
  - 2 RELATIVE GOALS
  - 0 SINGLE-CATEGORY ABSOLUTE GOALS
  - 3 PAIRWISE COMPARISON GOALS
  - 1 FOUR-CATEGORY ABSOLUTE GOALS
- SUPPLEMENTAL REVENUE REQUIREMENT - OVERALL

GOAL TYPE	YEAR	ST CAT	DEVIATION VARIABLE NAME	NO.
STATE REVENUE REQMTS GOAL	1		N1	1
			P1	2
	2		N2	3
			P2	4
	3		N3	5
			P3	6
SUPPLM REVENUE YEARLY GOAL	1		N4	7
			P4	8
	2		N5	9
			P5	10
	3		N6	11
			P6	12
PERCENT OF AVG. TOTAL COST GOAL	1	1	N7	13
			P7	14
		2	N8	15
			P8	16
		3	N9	17
			P9	18
		4	N10	19
			P10	20
		5	N11	21
			P11	22
		6	N12	23
			P12	24
	2	1	N13	25
			P13	26
		2	N14	27
			P14	28
		3	N15	29
			P15	30
		4	N16	31
			P16	32
		5	N17	33
			P17	34
		6	N18	35
			P18	36
	3	1	N19	37
			P19	38
		2	N20	39
			P20	40
		3	N21	41
			P21	42
		4	N22	43
			P22	44
		5	N23	45
			P23	46
		6	N24	47
			P24	48
CHANGE IN TUITION OVER TIME GOAL	1	1	N25	49
			P25	50
		2	N26	51
			P26	52
		3	N27	53
			P27	54
		4	N28	55
			P28	56
		5	N29	57
			P29	58
		6	N30	59
			P30	60
	2	1	N31	61
			P31	62
		2	N32	63
			P32	64
		3	N33	65
			P33	66
		4	N34	67
			P34	68
		5	N35	69
			P35	70
		6	N36	71
			P36	72
	3	1	N37	73
			P37	74
		2	N38	75
			P38	76
		3	N39	77
			P39	78
		4	N40	79
			P40	80
		5	N41	81
			P41	82
		6	N42	83
			P42	84
PEER INSTITUTION GOAL	1	1	N43	85
			P43	86
		2	N44	87
			P44	88
		3	N45	89
			N45	90
		4	N46	91
			N46	92
. . .	.	.	.	.
. . .	.	.	.	.
. . .	.	.	.	.
ABSOLUTE GOAL 4	1	1	N94	187
			P94	188
	2		N95	189
			P95	190
	3		N96	191
			P96	192
SUPPLM REVENUE OVERALL GOAL			N97	193
			P97	194

VARIABLE	YEAR	ST CAT	DECISION VARIABLE NAME	NO.
DECISION VARIABLES	1	1	X11	195
		2	X21	196
		3	X31	197
		4	X41	198
		5	X51	199
		6	X61	200
	2	1	X12	201
		2	X22	202
		3	X32	203
		4	X42	204
		5	X52	205
		6	X62	206
	3	1	X13	207
		2	X23	208
		3	X33	209
		4	X43	210
		5	X53	211
		6	X63	212

includes six student categories and spans a three-year planning horizon; the number of the various goal types included in the formulation is detailed at the top of Table 3.2

The solution routine, discussed in detail later, utilizes subroutines included in the LINDO system. The LINDO subroutines that are called from USER require a numeric reference to the variables, i.e. N1 may be referred to as variable 1, P1 as variable 2, X11 as variable 195, etc. LINDO automatically numbers the variables sequentially in the order that they are input from the formulation. Since the problem formulation is input to LINDO through an MPS-formatted file, the number assigned to each variable is controlled and known in advance. The COLUMNS section of the MPS-formatted file lists first, all of the deviation variables in ascending order, followed by the decision variables, sorted first by year then by student category. In general, the numbering scheme illustrated in Table 3.2, is defined as follows:

The variables defined as:

N1,P1,N2,..., NG, PG, X11,..., XM1, X12,..., XMK

translate to the numeric reference:

1, 2, 3,...,2G-1, 2G, 2G+1,...,2G+M,2G+M+1,...,2G+MK

where G is the total number of goals included in the model, M is the number of student categories, and K is the number of years in the planning horizon.

3.5.3.2 Translation of the priority structure to the achievement function

The priority structure entered by the user must be translated to an achievement function that is used by the solution routine to solve the SLGP problem. This translation process is outlined in Figure 3.3. The percentage-of-average-total-cost goal for resident graduate students is used as an example to demonstrate the process. Values that pertain to this goal in Figure 3.3 are "boxed in" or highlighted to facilitate the explanation. Also, for brevity this goal will hereafter be referred to in abbreviated form as, PATC RGR.

Once the user establishes the priority structure in file PRIOR, as shown in Figure 3.3, the DSS automatically performs the remaining steps necessary to generate an achievement function. As shown at the top of Figure 3.3, each goal's ranking and its corresponding subgoal weight are extracted from file PRIOR and stored in the file PRIOROPT. This step is performed in the Component Control Program PRIORITY. Each line in PRIOROPT corresponds to a particular goal. The first two columns for each goal are defined as

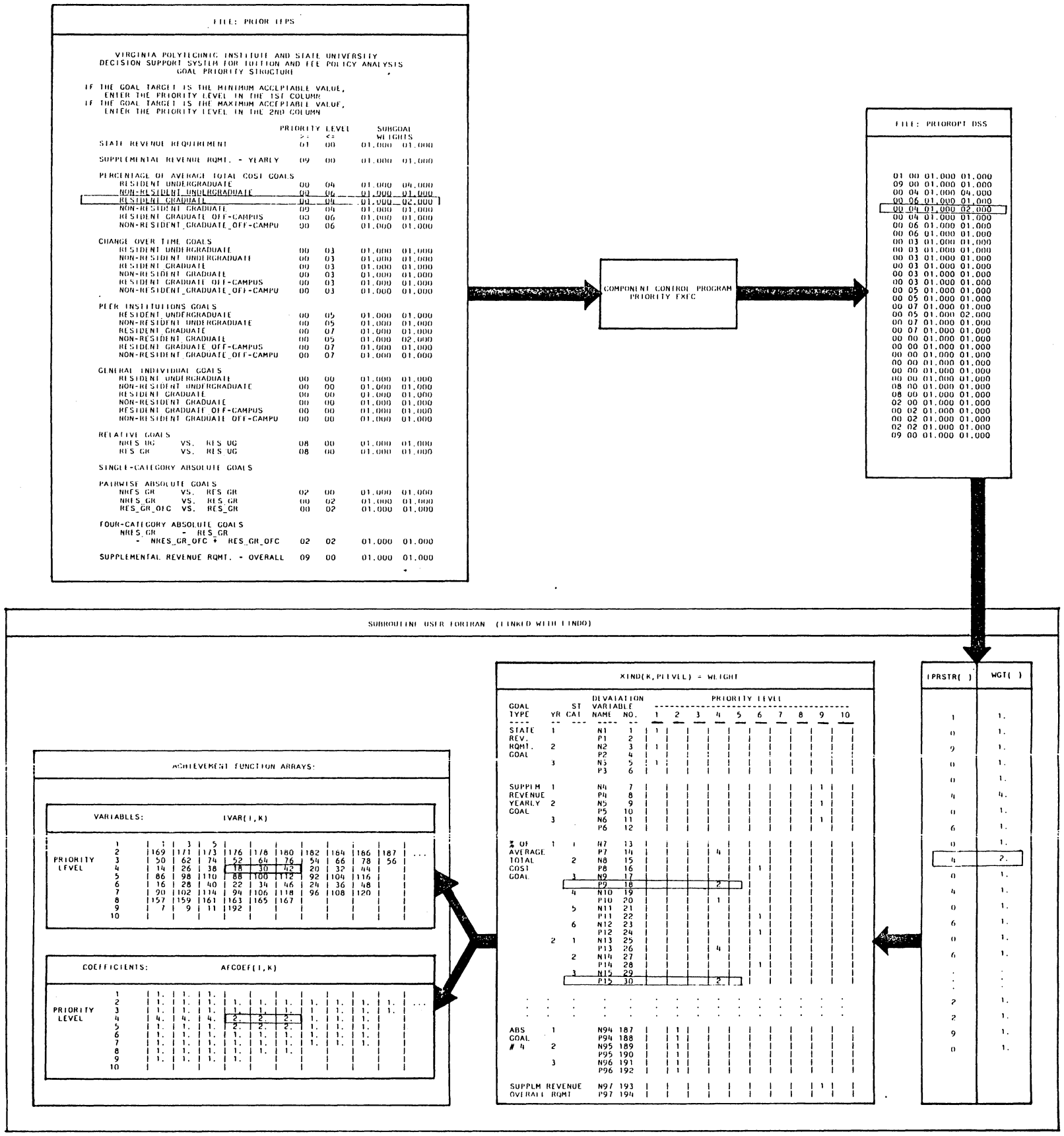


Figure 3.3: Translation of the priority structure to an achievement function.

the priority level for the negative and positive deviation variables, respectively; the last two columns are the subgoal weights for the negative and positive deviation variables, repectively.

The steps shown at the bottom of Figure 3.3 are all included in the subroutine USER, which is linked to LINDO. Once the solution process begins and LINDO is accessed, the priority information on each goal is input to subroutine USER from the file PRIOROPT DSS. The priority information is split into two vectors - IPRSTR containing the goal priority levels and WGT containing the subgoal weights. The information corresponding to the example goal, PATC RGR, was converted from the four columns in row five of PRIOROPT DSS to positions nine and ten in the vectors IPRSTR and WGT.

As previously described, the problem framework is read into USER from the file SETUP DSS; therefore, the total number of goals included in the model and the number of years in the planning horizon are available. Based on the specific problem framework and on the location of the priority level (and subgoal weight) in the vector IPRSTR (and WGT), each goal is uniquely identified. It is assumed that each goal's ranking is the same in each year of the planning horizon. If this assumption is not made, the decision maker would be faced with ranking an exorbitant

number of goals, i.e. the priority structure in Figure 3.3 contains 33 goals - it would contain 97 goals (three-year plannning horizon) if the ranking could vary by year.

The goal priority level and its corresonding subgoal weight are converted from elements in the vectors IPRSTR and WGT to specific deviation variables in the array XIND(K,PLEVEL). The array XIND is used to specify which of the deviation variables are to be included in the achievement function and at what priority level. Each row of the array corresponds to a specific deviation variable, each column represents a priority level, and the value entered in the array for a particular row and column is the subgoal weight. For example, XIND(18,4) = 2 means that LINDO variable 18 (model variable P9) is to be included in the achievement function at priority level four with a coefficient (subgoal weight) of 2. A few further examples should clarify this process.

The first goal in the model formulation, the state revenue requirement, is represented by the first two elements in the vector IPRST, 1 and 0, corresponding to the negative and positive deviation variables, respectively. This means that the decision maker intends to minimize the negative deviation variable at priority level one and not include the positive deviation variable at all in the

achievement function. Following the numbering scheme that was described in the preceeding section and in Table 3.2, the negative deviation variables - N1, N2, and N3 - are represented as variables 1, 3, and 5 in LINDO. These three variables have a value 1. in the first column of array XIND, indicating that they will be included in the achievement function at the first priority level with a coefficient of 1.

Now consider the PATC RGR goal - the negative and positive deviation variables for this goal are represented by elements 9 and 10 in the vectors IPRSTR and WGT (Figure 3.3). The ninth element in IPRSTR is zero, indicating that the negative deviation variable is not to be included in the achievement function; the tenth element has a value of 4, indicating that the positive deviation variable is to be minimized in the achievement function at priority level 4. The tenth element in WGT, 2, means that the coefficient of the positive deviation variable in the achievement function is 2. This is represented in the XIND array in Figure 3.3 as XIND(18,4)=2 for the first year, XIND(30,4)=2 for the second, and XIND(42,4)=2 for the third year.

All of the goals are translated in a similar manner from the vectors IPRSTR and WGT to the array XIND. Once this process is complete, all of the non-zero elements in

the array XIND are ignored and the information in XIND is transferred, as shown in Figure 3.3, to two arrays, IVAR and AFCOEF. These two arrays completely define the achievement function. Array IVAR provides a list of the deviation variables that are to be included in the objective function at each priority level, i.e. LINDO variables 1,3, and 5 (model variables N1, N3, and N5) are included at the first priority level, variables 169, 171, 173, etc. are included at the second priority level. The corresponding achievement function coefficients for the variables are included in the array AFCOEF.

3.5.3.3 The Solution routine

The basic SLGP method involves the solution of P single-objective linear programming problems - one for each priority level, p, p=1,2,...,PP. Only the goals and constraints associated with priority levels 1,...,P are included in each formulation. Once the linear program has been solved at priority level P, all deviation variables associated with that priority level are thereafter (at levels P+1,...,PP) constrained to their primal value. This preserves the preemptive priority concept of not letting a lower level goal degrade a higher priority goal's level of achievement.

As previously indicated, the GP model is solved using a modified SLGP approach. The basic approach is altered so that the model can be solved interactively in a single execution of the LINDO system. To accomplish this, the entire formulation is considered initially, i.e. the entire model is input all at one time at the beginning of the solution process. Hence, the SLGP method developed in this research does not solve a new and separate linear programming formulation at each priority level. When the solution process begins, all of the goals and their corresponding deviation variables are included in the formulation. When the model is solved at the first priority level, the objective function includes only the deviation variables that were specified at the highest priority level. Once an optimal solution is obtained, the upper bound on each deviation variable included in the first priority level is set equal to its primal value. Maintaining the same basis, the deviation variables corresponding to the second priority level are "inserted" into the objective function. The problem is re-solved with the deviation variables from both priority levels in the objective function and the first-priority deviation variables bounded between zero and their primal value from the first step. The process continues for each priority level.

The steps in the GP solution process, that are included in subroutine USER, are outlined in Figure 3.4. The first and last block in Figure 3.4 show that the sequential solution process is repeated at each priority level. Most of the steps in the solution process involve calls to LINDO subroutines that operate on the current basis.

The second block in Figure 3.4 shows that the deviation variables that are included at each priority level are inserted into the objective function by a call to the LINDO subroutine INSERT. The variables that are inserted at each priority level and their corresponding coefficients are controlled by the arrays IVAR and AFCOEF. These arrays were discussed previously and were defined in Figure 3.3.

The problem is solved at each priority level until an optimal solution is found. This is accomplished by a call to the LINDO subroutine GO, as shown in the third block in Figure 3.4. The basis from the previous solution is used as a starting point since subroutine GO operates on the "current" problem. Once optimality is reached at a particular priority level, P, the upper bound for each of the deviation variables associated with that priority level is set equal to its primal value. This is accomplished by two calls to LINDO subroutines: REPVAR to obtain the deviation variables' primal value and SETSUB to set the new

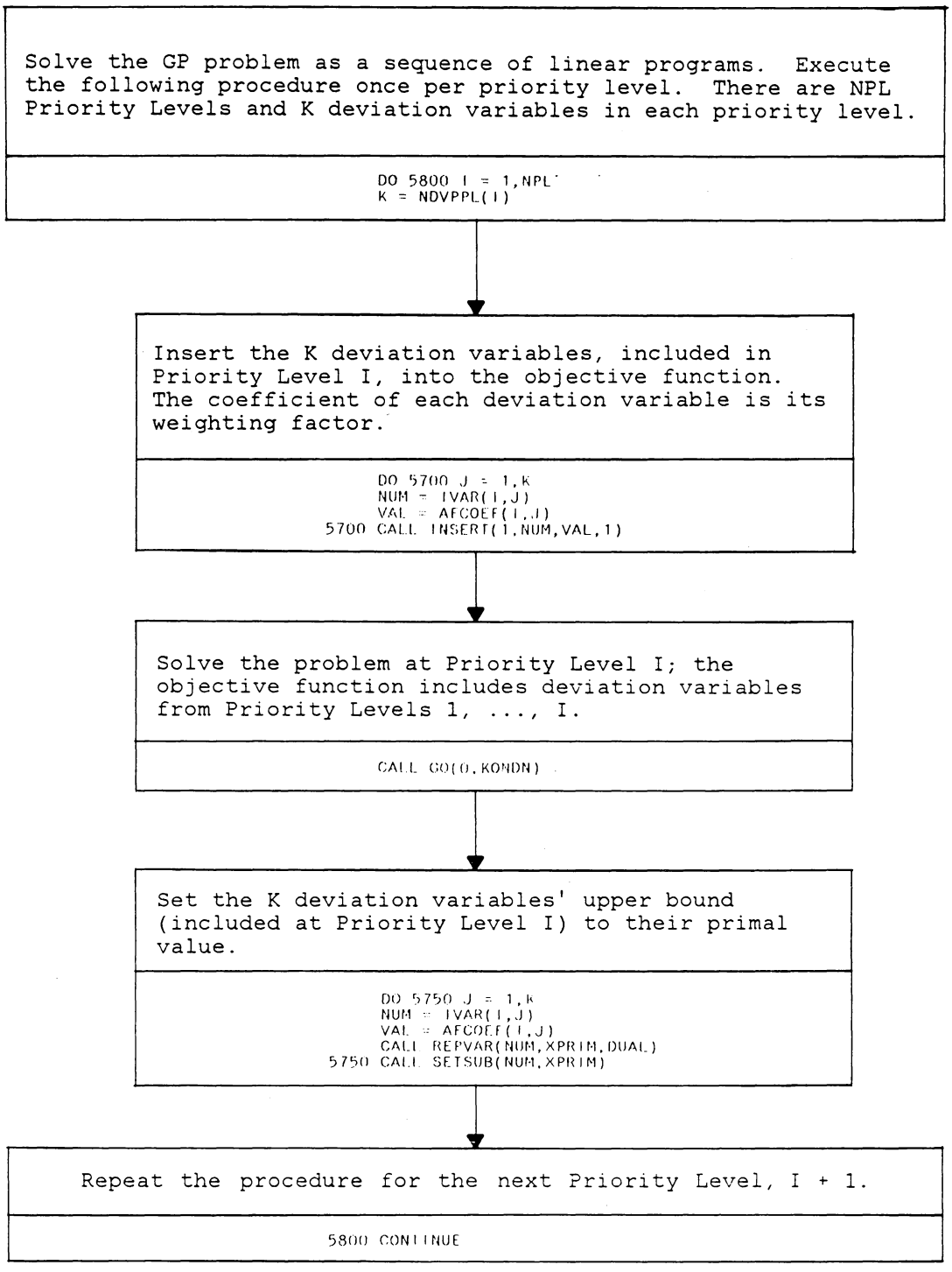


Figure 3.4: Steps in the GP solution process.

upper bound for those variables. The process returns to the second block in Figure 3.4, where the deviation variables associated with priority level P+1 are "inserted" into the objective function. The solution process stops when all of the priority levels have been considered.

3.5.3.4 Output and reports

The final function of subroutine USER is to generate a set of reports for use by the decision maker to assess and analyze the solution. Before leaving the LINDO system and returning to the Executive Control Program, subroutine USER generates four files of output, as shown in Figure 3.2. Each of the output files are discussed below.

Figure 3.5 provides an example of the first output file, TUIT. It contains the value of the decision variables - the tuition charge per credit hour for each student category in each year. The file also includes the base-year tuition charges as a point of reference. This report is intended to give the user the basic information necessary to make a cursory judgement as to whether or not the solution "appears acceptable". Once a solution has been reached, the contents of this file is automatically displayed to the user on the terminal screen; it is also available for hardcopy printing.

TUIT IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
RECOMMENDED TUITION CHARGES, IN DOLLARS PER CREDIT HOUR

	1982	1983	1984	1985
RES_UG	35.70	42.67	51.63	56.79
NRES_UG	77.20	92.64	115.80	119.32
RES_GR	44.89	52.97	64.41	69.65
NRES_GR	49.89	58.87	72.41	79.65
RES_GR_OFC	54.00	62.97	78.09	84.70
NRES_GR_OFC	59.00	68.87	86.09	94.70

Figure 3.5: Display of the GP solution - value of the decision variables

The second output file, TUITION, is similar to the previously described output; the same information is presented but in an IFPS-readable format. This data is subsequently used by an IFPS model and report generator to present the tuition charges on a full-time student basis and in conjunction with other charges - fees, dorm, and dining.

The third file, DEVVAR, contains a report on how well the solution achieved the specified goals. This type of report is essential in assessing a particular solution since it is more difficult to compare alternative model solutions from a goal programming problem than it is to compare solutions from a linear programming problem. There are two reasons for this; the first being, by definition, that no solution to a GP model can be infeasible.[9] The second reason is that the GP solution does not produce a single objective function value, as in linear programming, to use as a measure of solution performance. The achievement function, which is a multidimensional objective function, provides a separate value for each priority level. These values cannot be combined since the different priority levels usually contain variables with different units (incommensurable).

---

[9] unless a constraint (not a goal) in the first priority level is not achieved.

A portion of the achievement report DEVVAR is shown in Figure 3.6. For each goal in each year of the planning horizon, the report indicates whether the goal was achieved or not and provides a comparison of the goal's target value and the actual value that was attained in the solution. It also provides each deviation variable's priority level and value at the final solution. For example, in Figure 3.6, the state revenue requirement - included at the first priority level - was met in all three years; in fact, it was overachieved in 1983 and 1984 by 900. and 1100., respectively. The goals are listed in the same order as the display for entering the priority structure (aggregate goals, individual goals, etc.)

The final file generated by the USER subroutine is TRACE; an example of the report is shown in Figure 3.7. It provides a trace of the decision variable values at each priority level - at each step in the sequential solution to the GP model. This report can be used to assess the degree of conflict that exists in the model between goals and priority levels. It can also be used as a measure of how sensitive the solution is to the priority structure.

DEVVAR IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SUMMARY OF GOAL ACHIEVEMENTS

GOAL ACHVD ?	GOAL DEFINITION	PRIORITY LEVEL OF DEV VAR N	P	VALUE OF THE DEVIATION VARIABLES N	P	GOAL TARGET VALUE	GOAL LEVEL ATTAINED
STATE REVENUE REQUIREMENT							
YES	YEAR: 1983	1.	0.	0.0	0.0	36230.30	36230.30
YES	YEAR: 1984	1.	0.	0.0	900.00	43559.00	44459.00
YES	YEAR: 1985	1.	0.	0.0	1100.00	46762.99	47862.99
SUPPLEMENTAL REVENUE REQUIREMENT-YEARLY							
NO	YEAR: 1983	4.	0.	400.00	0.0	400.00	0.0
YES	YEAR: 1984	4.	0.	0.0	400.00	500.00	900.00
YES	YEAR: 1985	4.	0.	0.0	500.00	600.00	1100.00
PERCENTAGE OF AVERAGE TOTAL COST							
	YEAR: 1983						
NO	RES_UG	0.	5.	0.0	1.24	25.00	26.24
NO	NRES_UG	0.	8.	0.0	6.97	50.00	56.97
YES	RES_GR	0.	5.	0.65	0.0	30.00	29.35
YES	NRES_GR	0.	5.	2.39	0.0	35.00	32.61
YES	RES_GR_OFC	0.	8.	5.11	0.0	40.00	34.89
YES	NRES_GR_OFC	0.	8.	6.85	0.0	45.00	38.15
	YEAR: 1984						
YES	RES_UG	0.	5.	0.12	0.0	28.00	27.88
NO	NRES_UG	0.	8.	0.0	2.53	60.00	62.53
YES	RES_GR	0.	5.	1.70	0.0	33.00	31.30
YES	NRES_GR	0.	5.	12.81	0.0	48.00	35.19
YES	RES_GR_OFC	0.	8.	2.05	0.0	40.00	37.95
YES	NRES_GR_OFC	0.	8.	3.16	0.0	45.00	41.84

Figure 3.6: Goal achievement report of the GP solution.

TRACE IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SOLUTION VALUES FOR THE DECISION VARIABLES AT EACH PRIORITY LEVEL

VALUES AT EACH STEP IN THE SEQUENTIAL LINEAR PROGRAMMING

SOLUTION TO THE GOAL PROGRAMMING PROBLEM

PRIOR LEVEL	RES UG	NRES UG	RES GR	NRES GR	RES GR-OFC	NRES GR-OFC
YEAR 1						
1	42.13	92.86	52.97	58.87	75.33	81.23
2	42.09	92.64	55.47	60.47	65.47	70.47
3	42.67	92.64	52.97	58.87	62.97	68.87
4	42.67	92.64	52.97	58.87	62.97	68.87
5	42.67	92.64	52.97	58.87	62.97	68.87
6	42.67	92.64	52.97	58.87	62.97	68.87
7	42.67	92.64	52.97	58.87	62.97	68.87
8	42.67	92.64	52.97	58.87	62.97	68.87
9	42.67	92.64	52.97	58.87	62.97	68.87
YEAR 2						
1	42.09	92.64	55.47	60.47	65.47	70.47
2	42.67	92.64	52.97	58.87	62.97	68.87
3	42.67	92.64	52.97	58.87	62.97	68.87
4	42.67	92.64	52.97	58.87	62.97	68.87
5	51.24	119.85	65.15	72.41	0.0	101.53
6	51.35	111.71	61.62	72.41	75.30	86.09
7	51.63	115.80	64.41	72.41	78.09	86.09
8	51.63	115.80	64.41	72.41	78.09	86.09
9	51.63	115.80	64.41	72.41	78.09	86.09
YEAR 3						
1	42.67	92.64	52.97	58.87	62.97	68.87
2	42.67	92.64	52.97	58.87	62.97	68.87
3	42.67	92.64	52.97	58.87	62.97	68.87
4	51.35	111.71	61.62	72.41	75.30	86.09
5	51.63	115.80	64.41	72.41	78.09	86.09
6	51.63	115.80	64.41	72.41	78.09	86.09
7	56.49	112.97	67.78	79.65	82.83	94.70
8	56.79	119.32	69.65	79.65	84.70	94.70
9	56.79	119.32	69.65	79.65	84.70	94.70

Figure 3.7: Trace of the decision variable values at each priority level.

3.6 SUMMARY

A goal programming model was developed to allocate an institution's total annual revenue requirements to various student categories. This chapter presented the general GP model formulation and the methodology that was developed in order to solve it.

The GP model is the heart of the DSS. The computerized decision support system is used to formulate and solve GP models of the tuition allocation process, for a specific set of conditions. The general model and solution routine presented in this chapter provide the basis for the DSS. As described in the next chapter, the DSS is a vehicle used to facilitate the process of defining, solving, and analyzing the GP models. Such a process includes: defining the problem's framework, formulating the goals, calculating the model's parameter values, setting the goal target values, establishing the priority structure, and presenting the solution results.

Chapter IV

THE OVERALL DECISION SUPPORT SYSTEM ARCHITECTURE

The objective of this chapter is to describe the overall DSS architecture including the interactive computerized system that was developed to formulate and solve the goal programming model presented in Chapter 3. The chapter describes how the tasks directly associated with the GP model are handled, i.e. formulating the goals, calculating the parameter values, establishing the goal targets, specifying the priority structure, solving the model, and analyzing the solution. It also describes how the following supporting tasks are performed:

1. processing and storing the data,
2. interfacing between the user and the system,
3. providing displays and memory aids,
4. generating reports, and
5. integrating and controlling the numerous programs that comprise the DSS.

The chapter is divided into four sections. The first section provides an overview of the system; the remaining three sections discuss in detail the three subsystems of the DSS: data, dialog, and models.

4.1 SYSTEM DEFINITION

The overview of the system is presented in this section by (1) outlining the steps in a typical application of the DSS and defining the responsibilities of the user and the system, (2) defining the design philosophy and concepts that provided the foundation for the system's development, and (3) describing the software and programming languages that were used to construct the DSS. The section concludes with an outline of how the detailed definition of the system is presented in the remainder of the chapter.

4.1.1 Steps in a typical application of the DSS

As previously described, tuition charges are established by a goal programming model, imbedded in the DSS, that allocates the revenue requirements to student categories. From a predefined set, the user decides which goals to include in the analysis and ranks them in the order of importance to the university, thereby constructing a priority structure. The parameter estimates are either obtained directly from the user or calculated by supporting models. The user controls the data (enrollment levels, cost information, inflation rates, fee charges, etc.) that drive the support models. The DSS automatically formulates the goals, translates them into the form required by the

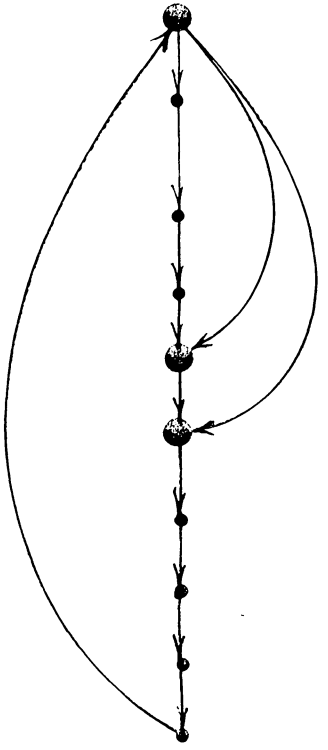
optimization routine, estimates the appropriate parameter values, solves the problem interactively, and generates a set of reports. Once this has been completed and the charges have been established, control is returned to the user for the purpose of exploring and testing alternatives and performing sensitivity analyses.

The specific steps that are involved in a typical session or application of the DSS are shown in Table 4.1. The figure in the left of the table indicates that the user has the choice of starting the session at various points or being guided through each step of the process. The figure also indicates that the user can "loop" through the process and easily explore alternatives and perform sensitivity analyses.

As shown in Table 4.1, the process of establishing tuition charges using the DSS involves dual and shared responsibilities between the DSS and the decision maker. The decision maker is responsible for accessing the system, defining the problem framework, selecting the goals, and setting the priority structure. The DSS formulates and solves the model as well as generates the necessary reports. The responsibility for providing the necessary information for the remainder of the steps is shared between the DSS and the decision maker. While the DSS calculates most of the

TABLE 4.1

Using the DSS involves dual and shared responsibilities between the DSS and the decision maker.



STEPS	RESPONSIBILITY
ACCESS THE SYSTEM	DM
DEFINE THE PROBLEM FRAMEWORK: PLANNING HORIZON, DATASETS	DM
SELECT THE GOALS TO BE INCLUDED	DM
PROVIDE THE PARAMETER VALUES	DSS/DM
ESTABLISH THE GOAL TARGET VALUES	DM/DSS
SET THE PRIORITY STRUCTURE	DM
FORMULATE THE PROBLEM	DSS
SOLVE THE PROBLEM	DSS
GENERATE MANAGEMENT AND BACKUP REPORTS	DSS
ANALYZE THE SOLUTION	DM/DSS

DM = DECISION MAKER

DSS = DECISION SUPPORT SYSTEM

model's parameter values, some must be directly input by the user. Likewise, while the decision maker supplies most of the goal target values, some are automatically calculated by the DSS. The final step in the process - analyzing the solution - is also a joint effort; the DSS supplies the reports and information to the decision maker who ultimately selects the "best" solution. Once the basic problem has been defined and the basic data supplied, the user can bypass many of the steps and quickly test alternative priority structures or measure the effect of changes in the goal target values.

The steps outlined in Table 4.1 and the shared responsibility between the DSS and the decision maker demonstrate two key DSS concepts: (1) the system supports administrative judgements and does not replace them and (2) the system improves the effectiveness of the decision-making process and does not just improve its efficiency.

4.1.2 The design philosophy and concepts

In order for the the DSS to be a valuable and useful part of the decision-making process, it must possess certain characteristics. The DSS was designed to be:

1. interactive - to provide immediate feedback and to provide the decision maker with an active role in the final solution

2. comprehensive - to include and integrate data and reports with modeling and analysis
3. iterative - to facilitate the user's need to explore and test alternatives
4. flexible - to permit modification, extension, and updating
5. easy to use - to facilitate what-if and sensitivity analyses and to minimize the need for the user to be skilled in computers and management science.

These characteristics strongly influenced the system's overall design, including: the development of the software structure, the selection of the programming languages, the utilization of existing software packages, integration of the subsystems, and the layout of the data files and reports.

The basis for the design is the software architecture represented in Figure 4.1. The system is divided into three major subsystems - the dialog subsystem, the model subsystem, and the data subsystem. According to Sprague and Carlson [85], it has been more effective, in building the DSS, to separate the functions of the three subsystems than to intermix them. They also point out that just as the software structure separates the functions it must also provide a mechanism to integrate them through a DSS

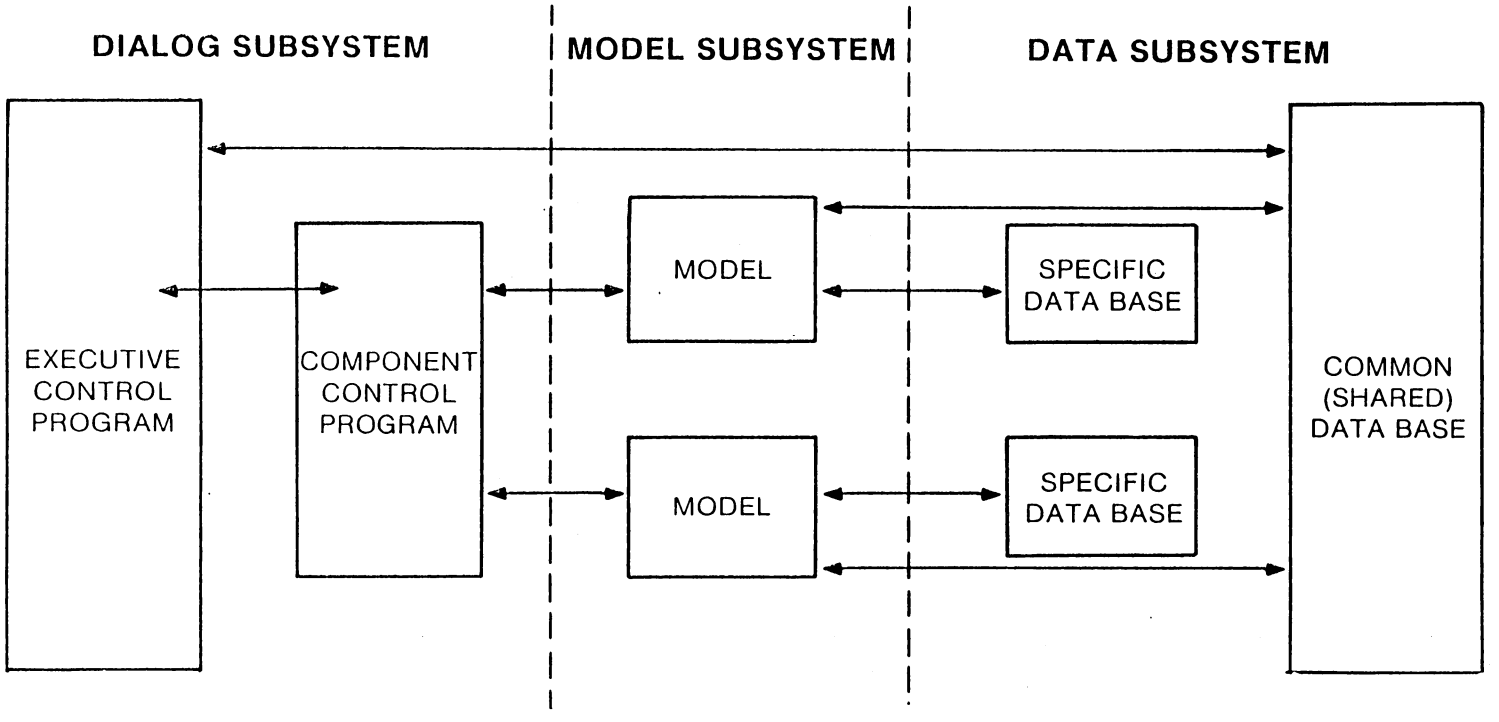


Figure 4.1: The software architecture that is the basis for the DSS design.

architecture. Each of the subsystems are further partitioned into components to facilitate construction and testing, but more importantly to provide flexibility in the system and to facilitate modification, extension, and updating. Figure 4.1 shows how the software interfaces, in this particular DSS, connect the three subsystems and their components, thereby forming an integrated and unified system.

4.1.3 The computer system and programming languages

The DSS was developed and implemented on the Virginia Polytechnic Institute and State University computer system, which consists of two IBM computers - an IBM 3081 and an IBM 4341 Group 2 processor; the operating system is the IBM VM/SP (Virtual Machine/System Product). The sessions are conducted through CMS (Conversational Monitoring System) on an interactive computer terminal, e.g. Memorex 1377, Televideo 970.

All of the programs, models, reports, and data that comprise the DSS are stored in files on a minidisk. Each file is identified by a unique name and type. The file name either indicates the function of the file or provides a description of the information contained within the file. The file type is one the four file types used in the DSS -

EXEC, IFPS, FORTRAN, DSS. The executive control program and the component control programs have a file type of EXEC because they are written using the IBM command language EXEC 2. Similarly, any program written in FORTRAN has a file type of FORTRAN so that it can by run interactively under CMS. Any file that is created or used by the IFPS system has a file type of IFPS; this includes command files, the models and reports file, and data files. Non-IFPS-related data files have a file type of DSS. There is one additional type of file - the file that contains the goal programming model in MPS format. In order for the file to be read by the LINDO system, it has a file name of FILE and a file type of FT08F001.

The DSS utilizes the EXEC 2 interpreter and the FORTRAN H Extended language processors available under CMS. The control programs (executive and component) are written in EXEC 2. As a command programming language, EXEC 2 is used to perform a wide range of functions: (1) provide interactive communication with the user, (2) read, write and insert information in CMS files, (3) execute CMS commands and FORTRAN programs, (4) communicate with other EXECs, (5) create, list, modify, and edit CMS files, (6) store and manipulate variables, and (7) control the flow of the process and perform logical operations. Most of the

programs written in EXEC 2 are discussed in the dialog subsystem section of this chapter.

The full-screen editing capability of XEDIT is used to input data directly to the files. XEDIT is also used to create, modify, and save the files themselves. PF (Program Function) keys are used to execute certain commands by pressing a single key. For example, two of the keys allow the user to examine the contents of a file, one screen at a time; PF key 7 moves one screen toward the beginning of the file, PF8 moves one screen toward the end of the file.

FORTRAN programs are used to create some of the input data files, formulate the goal programming model in MPS format, solve the GP model in conjunction with LINDO, and generate the reports associated with the optimization procedure. All of the FORTRAN programs (H Extended compiler) are run interactively under CMS. They are executed from either the executive control program or from a component control program. All of the input and output data are stored in files; the file name and FORTRAN unit number are defined in the control programs by FILEDEF statements.

As discussed in the previous chapter, the DSS utilizes the commercial software package LINDO to solve the goal programming model. The LINDO package provides the simplex solution algorithm and a user-written FORTRAN subroutine

provides the logic to solve the GP as a sequence of linear programming problems. LINDO is designed for interactive computing, is command oriented, has a FORTRAN interface capability, provides access to LINDO subroutines, handles MPS-formatted files, and translates the problem formulation from MPS to equation form and vice-versa.

IFPS,[10] another commercial software system, is used in the DSS to calculate the parameter values and to generate reports. The interactive financial planning language facilitates the integration of data, models, and reports. It is a non-procedural language that uses English-like terminology. The multipurpose system has the following capabilities [20, p. 1-4]:

1. It is a modeling language used to define relationships and perform financial/quantitative computations and projections.
2. It is a reporting language used to organize information into any desired, presentation-quality format.

---

[10] IFPS Release 9.0 V2 with IFPS/LINK 1.0 V2 normally requires 1024K of memory. Three of the programs used in the DSS require additional memory. The largest requires 1152K of memory in order to solve a problem that is defined by a 505 row (variable) by 16 column array.

3. It is a data-manipulation language used to organize change, and update data and to create and consolidate new data files from existing information.
4. It is a command language used to process a sequence of commands, directives, or options by using a single statement.

The DSS utilizes several of IFPS's built-in financial and statistical functions and subroutines, e.g. amortization, simple linear regression, rounding. The details of most of the IFPS programs are discussed in the models subsystem section of this chapter.

Table 4.2 provides a summary of the tasks that each of the four languages perform in the DSS. Except for LINDO, all of the languages have multiple functions in the DSS. Also many of the tasks are performed by more than one language.

4.1.4 Plan for the detailed presentation of the DSS

The entire system is diagramed and flow charted in Figures 4.2 through 4.5. The left-hand portion of the figures show the flow through the executive control program (ECP); the activities performed by the component control programs (CCPs) are illustrated in the center portion, enclosed in the dashed-line boxes; elements of the data

TABLE 4.2

Summary of the tasks that each of the programming languages perform in the DSS.

	Programming languages			
Task in the DSS	EXEC 2	FORTRAN	LINDO	IFPS
User interface	X			
Process control	X			
Create/modify files:				
Display		X		X
Data	X	X		X
Reports		X		X
Model formulation		X		
Model solution				
Goal programming		X	X	
Parameter values				X

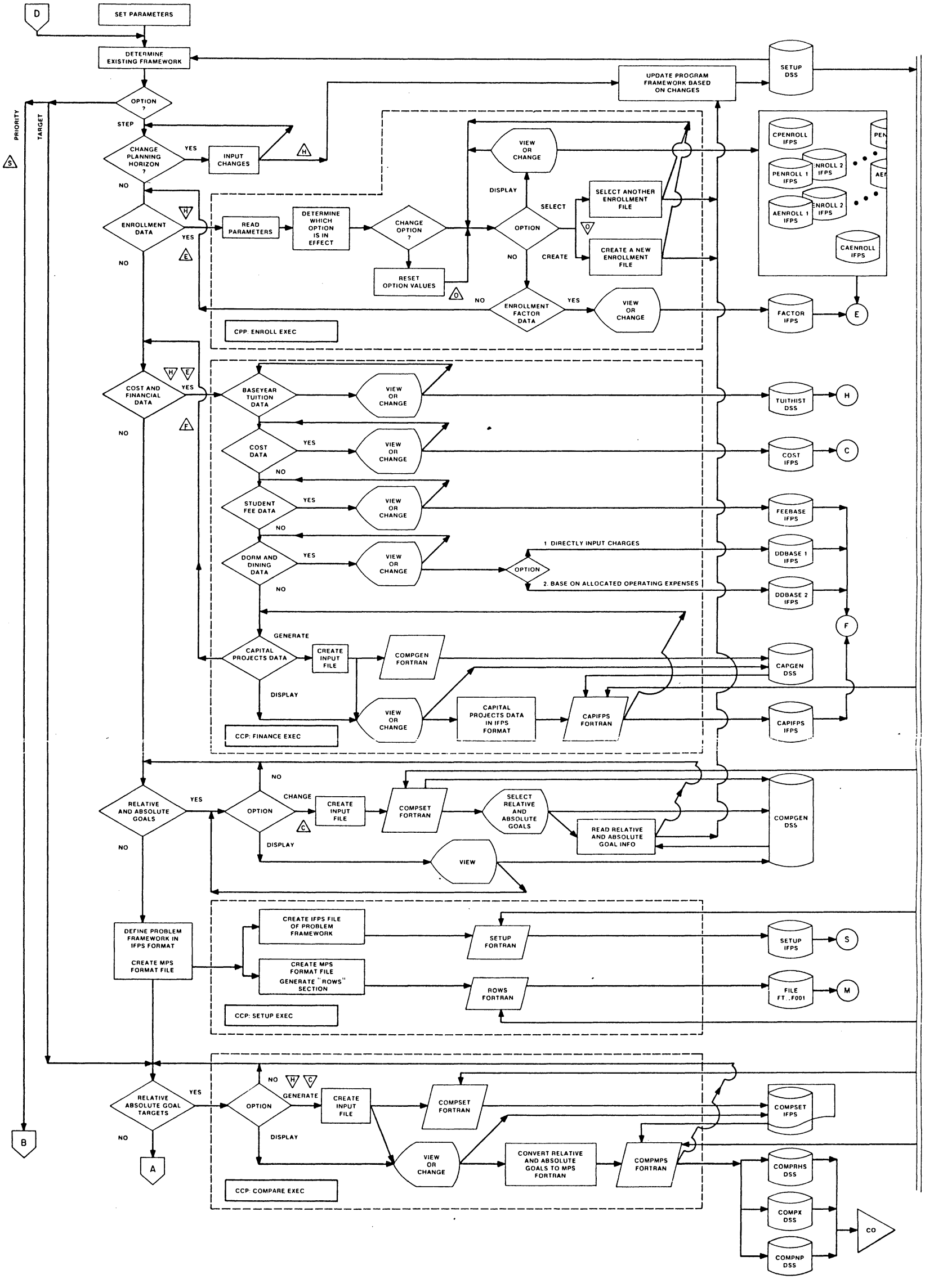


Figure 4.2: Flow-chart diagram of the DSS - Part 1.

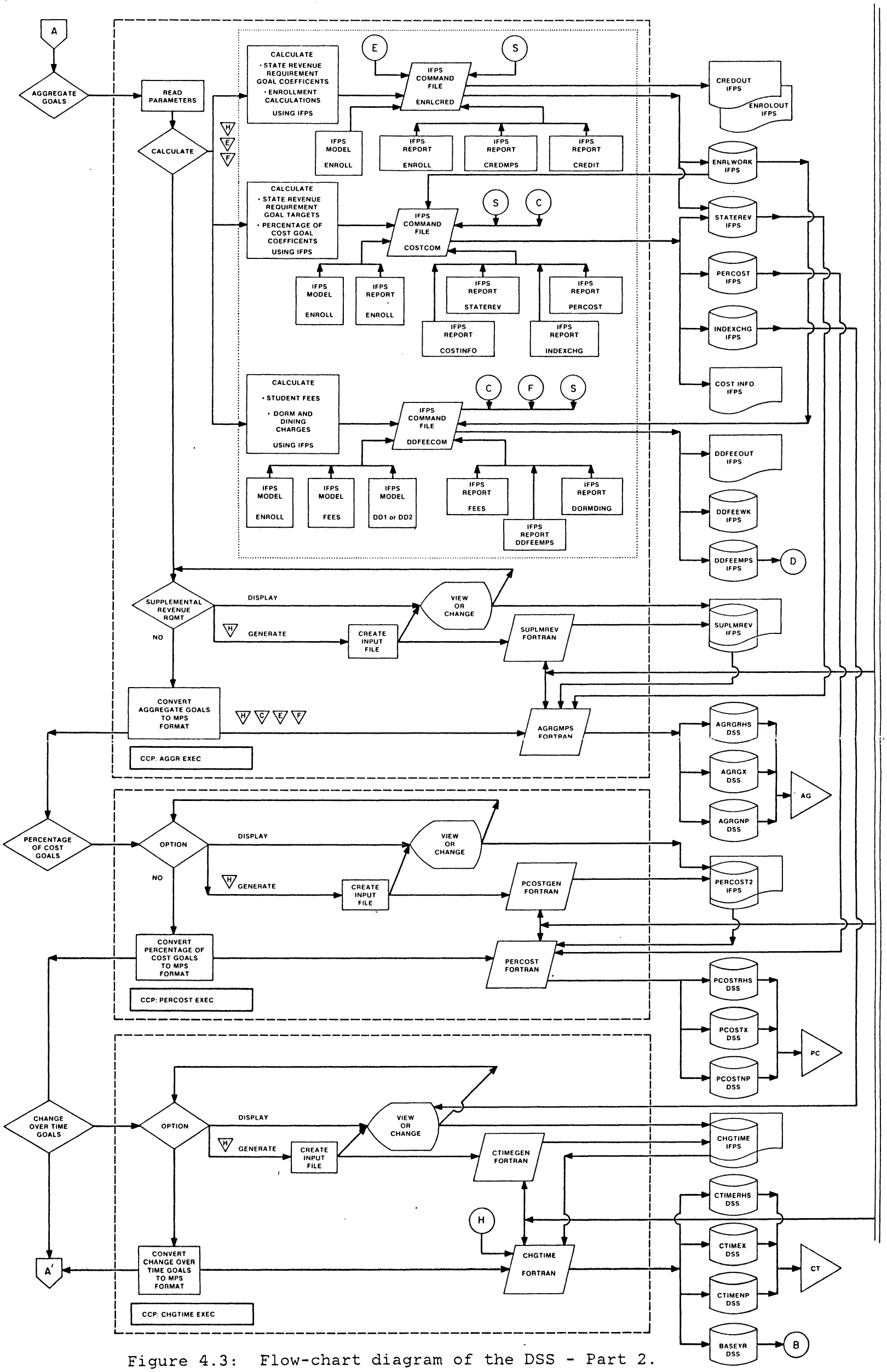


Figure 4.3: Flow-chart diagram of the DSS - Part 2.

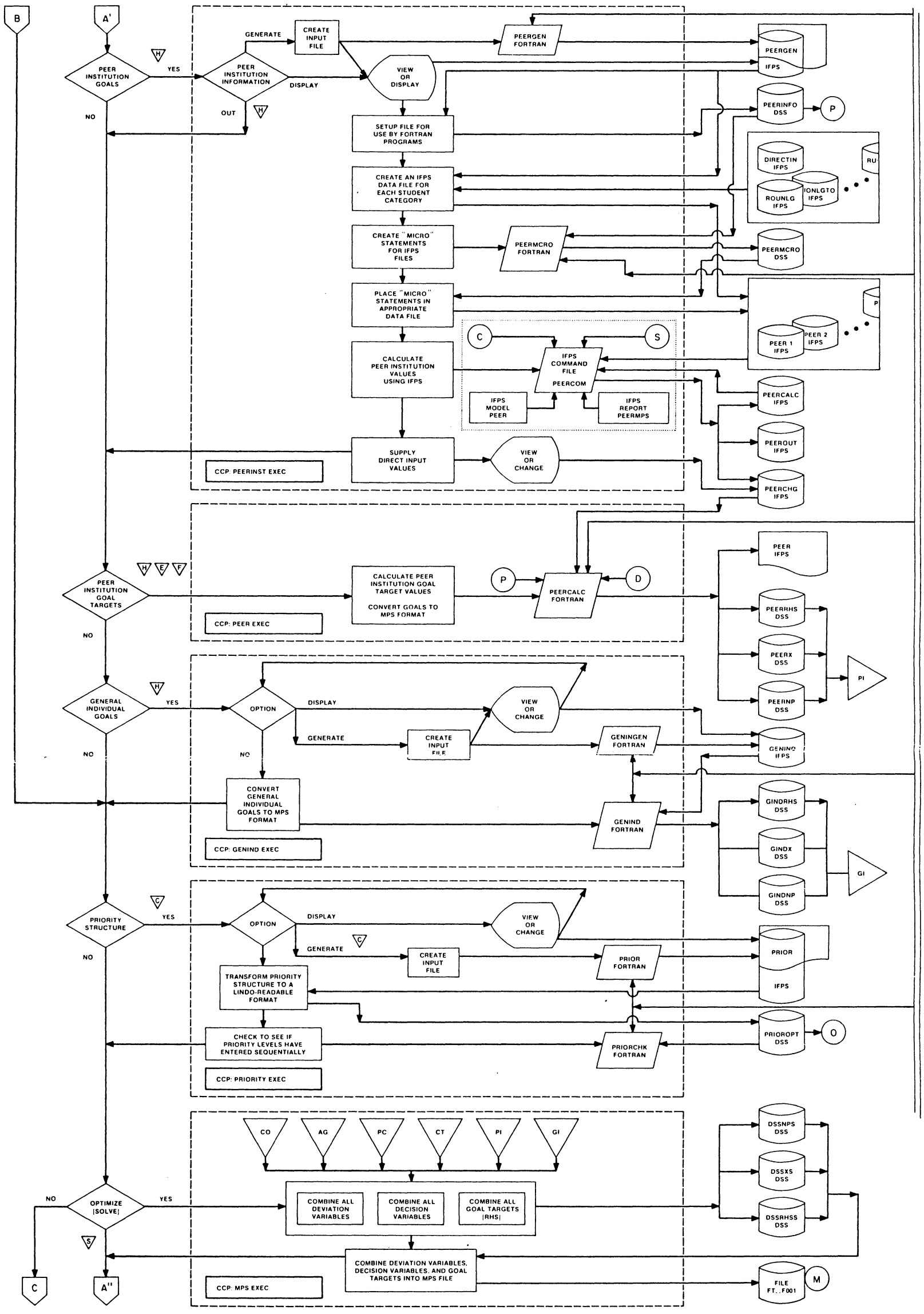
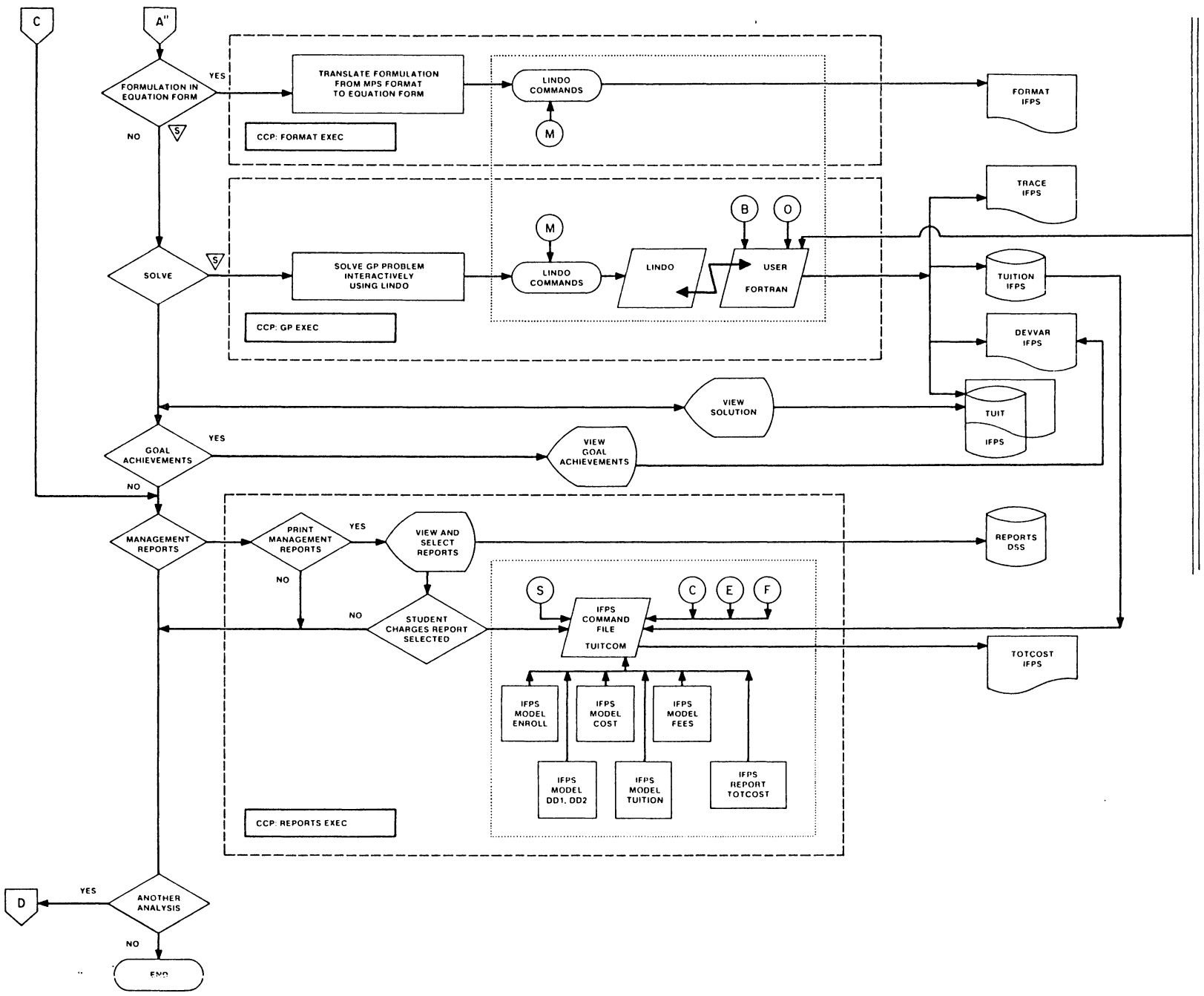


Figure 4.4: Flow-chart diagram of the DSS - Part 3.



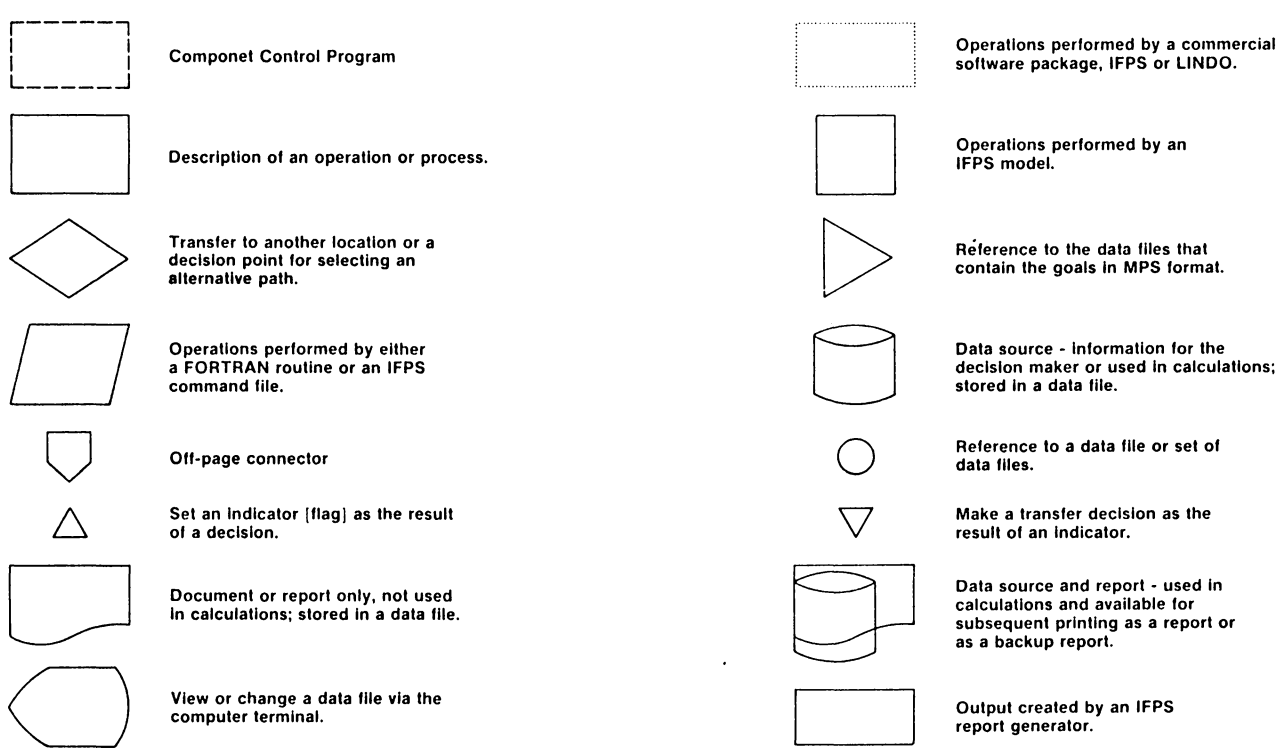


Figure 4.5: Flow-chart diagram of the DSS - Part 4.

subsystem are presented in the right-hand portion of the figures. The components of the models subsystem are included within the CCP that controls the operation. Operations performed by IFPS and LINDO, which consitute most of the models subsystem, are enclosed in the dotted-line boxes in Figures 4.2 through 4.5. These diagrams are used extensively in the detailed description of the system that follows.

The system is defined and described in detail in the following three sections, which correspond to the three subsystems that comprise the DSS. The data, dialog, and models subsystems are so interrelated that it is nearly impossible to discuss one subsystem independently of the other two. Therefore, since the dialog subsystem controls the operations performed in the other subsystems and since it is the primary interface with the user, most of the system will be described in the dialog subsystem.

The program listings for all of the computer programs are provided in the appendixes as follows: executive control program in Appendix A, component control programs in Appendix B, FORTRAN programs in Appendix C, and IFPS programs in Appendix D (arranged alphabetically by the program's name); data files that are not displayed in the text are included in Appendix E.

4.2 THE DATA SUBSYSTEM

The data subsystem, through its memory function, serves to integrate the dialog and model subsystems. It organizes, processes, and stores information as well as manipulates and converts data for use in the different components of the DSS. The data subsystem supplies input data to models, provides workspace for intermediate calculations, stores output reports, and maintains displays and memory aids for the user. As shown in Figure 4.1, the data subsystem contains a set of common or shared data files and a series of data files that are unique to specific models. The common data files, for example, include the enrollment projection files and the problem definition file, both of which are used extensively by many of the models. In contrast, the file that contains the yearly supplemental-revenue goal target values is used by only one model; it is an example of a specific data file.

All of the information used by the DSS is stored in data files, the files are represented symbolically in the right-hand portion of the system diagram (Figures 4.2 through 4.5). The files either contain data - represented by the "disk" symbol, reports - represented by the "documents" symbol, or both data and reports - represented by the "overlaid document and disk" symbols. Whenever

possible the transfer of information is represented by flow lines that run directly to and from the appropriate data file. If a data file is used in more than one figure, it is represented by the circular symbol containing a single-letter identifier. The three data files used to store a particular type of goal in MPS format are represented as a group by the triangular symbol containing a two-letter indentifier. The problem framework file SETUP DSS is used so extensively throughout a session that is is represented by the solid line along the right-hand side of Figures 4.2 through 4.5. It can be thought of as a data "spine" that furnishes the problem framework and definition to most of the computer programs.

All of the files are identified by a file name and a file type. Only the file name is referenced in the text unless two files have the same name but a different type. The data files in the DSS have either one of two file types. As discussed earlier, any file that is used or created by the IFPS system must have a file type of IFPS. Also, in order to simplify the procedure for printing reports (in the CCP REPORTS), all files that are available for printing through the CCP have the same file type, IFPS, whether or not the file is used or created by the IFPS system. All other data files have a file type of DSS.

All files created by the IFPS system have a variable record format; but, all files that are used as a source of input to FORTRAN programs must have a fixed eighty-record format. Therefore, the DSS automatically converts any IFPS-created file that is used as an input to a FORTRAN routine to a fixed eighty-record format.

The individual data files are best discussed either in conjunction with the control program that accesses it or with the model that uses it, rather than in isolation. Since the problem framework file, SETUP DSS, is used so extensively by the DSS, it is defined below, along with the enrollment data files. Examples of many of the data files are displayed in figures throughout the remainder of this chapter. Those that are not displayed in figures in the chapter are included in Appendix E. Table 4.3 provides a summary of where an example of each data file is displayed. The files are arranged alphabetically by file name. Some files are not displayed in the chapters or in the Appendix, either because they are similar to a file that is already displayed or because they are working files. The files that are considered similar are those that contain the goals in MPS format and the enrollment data. They are indicated by an arrow, "--->", in the Figure number column of Table 4.3 that points to the comparable file name in the Comment column.

TABLE 4.3

Summary of where an example of each data file is displayed[1].

Data file name	Data file type	Figure number	Appendix number	Comment
AENROLL_[2]	IFPS	4.7		
AGRGNP	DSS	--->		similar to Fig. 4.12
AGRGRHS	DSS	--->		similar to Fig. 4.12
AGRGX	DSS	--->		similar to Fig. 4.12
BASEYR	DSS	-	-	
CAENROLL	IFPS	4.8		
CAPGEN	DSS	4.11		
CAPIFPS	IFPS	4.11		
CHGTIME	IFPS		E.1	Report
COMPGEN	DSS	4.10		
COMPNP	DSS	4.12		
COMPRHS	DSS	4.12		
COMPSET	IFPS	4.12		Report
COMPX	DSS	4.12		
COST	IFPS		E.2	
COSTINFO	IFPS	4.24		Report
CPENROLL	IFPS	--->		similar to Fig. 4.8
CREDOUT	IFPS	4.20		Report
CTIMENP	DSS	--->		similar to Fig. 4.12
CTIMERHS	DSS	--->		similar to Fig. 4.12
CTIMEX	DSS	--->		similar to Fig. 4.12
DDBASE1	IFPS		E.3	
DDBASE2	IFPS		E.4	
DDFEEMPS	IFPS	-	-	
DDFEEOUT	IFPS	4.22, 4.23		Report
DDFEEWK	IFPS	-	-	
DEVVAR	IFPS	3.6		Report
DSSNP	DSS	--->		----------
DSSNPS	DSS	--->		:
DSSRHS	DSS	--->		similar to Fig. 4.12
DSSRHSS	DSS	--->		(temporary)
DSSX	DSS	--->		:
DSSXS	DSS	4.16,--->		----------

[1]There are a series of peer institution data files (RUONLG, NUONLGTO, etc.) that are not included in the list; they are similar to Figure 4.13.

[2]There are more than one of these files; the "_" represents a number 1,2,..., etc.

Figure 4.3 (Continued)

Data file name	Data file type	Figure number	Appendix number	Comment
ENRLWORK	IFPS	-	-	
ENROLOUT	IFPS		E.5	Report
FACTOR	IFPS		E.6	
FEEBASE	IFPS		E.7	
FILE	FT08F001		E.8	
FORMAT	IFPS	4.17		Report
GENIND	IFPS		E.9	
GINDNP	DSS	--->		similar to Fig. 4.12
GINDRHS	DSS	--->		similar to Fig. 4.12
GINDX	DSS	--->		similar to Fig. 4.12
INDEXCHG	IFPS	4.25		
PCOSTNP	DSS	--->		similar to Fig. 4.12
PCOSTRHS	DSS	--->		similar to Fig. 4.12
PCOSTX	DSS	--->		similar to Fig. 4.12
PEER	IFPS		E.10	Report
PEERCALC	IFPS	-	-	
PEERCHG	IFPS	-	-	
PEERGEN	IFPS	4.14		Report
PEERINFO	DSS	-	-	
PEERMCRO	DSS	-	-	
PEERNP	DSS	--->		similar to Fig. 4.12
PEEROUT	IFPS	-	-	
PEERRHS	DSS	--->		similar to Fig. 4.12
PEERX	DSS	--->		similar to Fig. 4.12
PEER_2	IFPS		E.11	
PENROLL_2	IFPS			similar to Fig. 4.7
PERCOST	IFPS	-	-	
PERCOST2	IFPS		E.12	Report
PRIOR	IFPS	3.3, 4.15		Report
PRIOROPT	DSS	3.3		
REPORTS	DSS	4.18		
SETUP	DSS	4.6		
SETUP	IFPS		E.13	
STATEREV	IFPS	-	-	
SUPLMREV	IFPS		E.14	Report
TOTCOST	IFPS	4.28		Report
TRACE	IFPS	3.7		Report
TUIT	IFPS	3.5		Report
TUITHIST	DSS		E.15	
TUITION	IFPS	-	-	

The files that are not displayed at all, the working files, contain a "-" in both the Figure number and the Appendix number columns of Table 4.3. The word "Report" in the Comment column indicates that the file is available for hardcopy printing through the CCP REPORTS. Notice that the file names with the prefix "DSS" are only temporary files - they are erased once the model has been translated to MPS format.

4.2.1 The problem framework data file, SETUP DSS

As can be seen from the system diagram in Figures 4.2 through 4.5 and as will be apparent in the following discussion of the system, the data file that is accessed the most often by the DSS is SETUP DSS. It contains the definition of the problem's framework; an example of the file is displayed in Figure 4.6.

The first three variables are associated with the planning horizon: BYEAR is the earliest year for which data is available in the DSS, SYEAR is the first year of the planning horizon, and NYEAR is the number of years in the planning horizon. The variable NSCAT is the number of student categories. The next seven variables define the

SETUP DSS

```
BYEAR = 1978
SYEAR = 1983
NYEAR = 3
NSCAT = 6
NAGRG = 2
NINDG = 4
NRCMG = 2
N1ABG = 0
N2ABG = 3
N4ABG = 1
NOVRG = 1
ENROLL = PENROLL1 AENROLL1
ISPAN = 16
NAME1 RESIDENT_UNDERGRADUATE
NAME2 NON-RESIDENT_UNDERGRADUATE
NAME3 RESIDENT_GRADUATE
NAME4 NON-RESIDENT_GRADUATE
NAME5 RESIDENT_GRADUATE_OFF-CAMPUS
NAME6 NON-RESIDENT_GRADUATE_OFF-CAMPUS
ABBR1 RES_UG
ABBR2 NRES_UG
ABBR3 RES_GR
ABBR4 NRES_GR
ABBR5 RES_GR_OFC
ABBR6 NRES_GR_OFC
RELG1 2 1
RELG2 3 1
ABS21 4 3
ABS22 4 3
ABS23 5 3
ABS41 +4 -3 -6 +5
```

Figure 4.6: The problem framework data file, SETUP DSS.

number of goals of each type (aggregate, individual, relative, single-category absolute, pairwise absolute, four-category absolute, and overall) included in the model formulation. The names of the two enrollment data files that are currently in effect are specified by the variable ENROLL. ISPAN is the maximum number of years, both historical and projected, that the DSS can accomodate. The next two sets of variables NAMEm and ABBRm, m=1,2,...MM, are the names (up to 32 characters) and the abbreviations (up to 12 characters) for each of the MM student categories.

The last section of the file SETUP DSS defines the relative and absolute goals that are in the current model formulation. For example, in Figure 4.6 the variable RELG1 is the first relative goal and it includes decision variables two and one. Similarly, variable ABS21 is the first pairwise (two-category) absolute goal and it includes decision variables four and three (non-resident and resident on-campus graduate students).

4.2.2 The enrollment data files

There are two formats used to define the enrollment data values for use in the DSS. The primary format, shown in Figure 4.7, is a typical file used to input the enrollment values; whereas, the secondary format, shown in

AENROLL1 IFPS

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*             BASELINE FILE FOR "ACTUAL" ENROLLMENT VALUES
*    OIR DPB FORM 2-B      (AY 82-83 THRU 88-89)       DATED:  9/29/82
* INSTRUCTIONS:
*      1) DIRECTLY ENTER THE ENROLLMENT VALUES THAT YOU WANT TO USE
*         IN THE APPROPRIATE YEARS,
* OR   2) USE THE 'PREVIOUS MODIFIER' TO INCREASE OR DECREASE THE
*         ENROLLMENT BY A CONSTANT RATE, I.E. TO LET RESIDENT UNDERGRAD
*         ENROLLMENT INCREASE BY 1.5 % PER YEAR AFTER 1984, CHANGE:
*                              1978   1979   1980   1981   1982   1983
*    ACADEMIC YEARS            1984   1985   1986   1987   1988   1989
*ACTUAL REG FT UG RES  =  13637, 13637, 13637, 13637, 13637, 13617,'
*                         13502, 13447, 13453, 13457, 13548
*  TO:
*ACTUAL REG FT UG RES  =  13637, 13637, 13637, 13637, 13637, 13617,'
*                         13502, PREVIOUS * 1.015
***   REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
******    FULL-TIME (FT)
*                         1978   1979   1980   1981   1982   1983
*  ACADEMIC YEARS         1984   1985   1986   1987   1988   1989
*                         1990   1991   1992   1993
ACTUAL REG FT UG RES  =  13637, 13637, 13637, 13637, 13637, 13617,'
                         13502, 13447, 13453, 13457, 13548

ACTUAL REG FT UG NRES =   3975,  3975,  3975,  3975,  3975,  3993,'
                          3953,  3929,  3930,  3928,  3927

ACTUAL REG FT GR RES  =    805,   805,   805,   805,   805,   864,'
                           945,   982,  1027,  1036,  1042

ACTUAL REG FT GR NRES =   1602,  1602,  1602,  1602,  1602,  1713,'
                          1826,  1906,  1968,  1996,  1993

***   REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
******     PART-TIME (PT)
*                         1978   1979   1980   1981   1982   1983
* ACADEMIC YEARS          1984   1985   1986   1987   1988   1989
*                         1990   1991   1992   1993
ACTUAL REG PT UG RES  =    506,   506,   506,   506,   506,   527,'
                           524,   519,   519,   519,   519

ACTUAL REG PT UG NRES =     72,    72,    72,    72,    72,    73,'
                            73,    73,    73,    73,    73

ACTUAL REG PT GR RES  =    432,   432,   432,   432,   432,   462,'
                           489,   511,   532,   540,   538

ACTUAL REG PT GR NRES =    267,   267,   267,   267,   267,   279,'
                           294,   305,   319,   322,   321

***  SUMMER SESSION (SUM) - HEADCOUNT
******    FULL-TIME (FT)
*                         1978   1979   1980   1981   1982   1983
*   ACADEMIC YEARS        1984   1985   1986   1987   1988   1989
```

Figure 4.7: Primary enrollment data file format (file AENROLL1 IFPS).

Figure 4.8, is used internally by the system and is not accessible to the user.

Each file contains the number of students enrolled (headcount) in each year, by the following classifications: academic level, residency status, participation status, session, and campus location. Using the primary format, enrollment values are input directly into the file by the user. Once the appropriate data file has been accessed, each enrollment value is entered or changed by placing the cursor in the appropriate row and column and entering the correct value. The enrollment value (number of students in each classification) may be either directly specified or calculated using an IFPS modifier. The modifier calculates each year's value based on the previous year. For example, using the modifier PREVIOUS*1.05 causes the enrollment in each subsequent year to be 5% higher than the preceeding year, PREVIOUS+100 generates an additional 100 students each year, etc.

The DSS has the capability of using either one or two different enrollment data files at the same time in the model. The use of two different sets of data is necessary because the projected enrollment estimate, used to determine the tuition and fee recovery rate, is usually different from the estimate used to actually set the tuition charges. This

CAENROLL IFPS

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*             BASELINE FILE FOR "ACTUAL" ENROLLMENT VALUES
*         WHEN "PROJECTED" ENROLLMENT IS SPECIFIED AS THE BASE
*
***   REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
******    FULL-TIME (FT)
ACTUAL REG FT UG RES  =  L211
ACTUAL REG FT UG NRES =  L212
ACTUAL REG FT GR RES  =  L213
ACTUAL REG FT GR NRES =  L214
******     PART-TIME (PT)
ACTUAL REG PT UG RES  =  L221
ACTUAL REG PT UG NRES =  L222
ACTUAL REG PT GR RES  =  L223
ACTUAL REG PT GR NRES =  L224
***  SUMMER SESSION (SUM) - HEADCOUNT
******    FULL-TIME (FT)
ACTUAL SUM FT UG RES  =   L231
ACTUAL SUM FT UG NRES =   L232
ACTUAL SUM FT GR RES  =   L233
ACTUAL SUM FT GR NRES =   L234
******    PART-TIME (PT)
ACTUAL SUM PT UG RES  =   L241
ACTUAL SUM PT UG NRES =   L242
ACTUAL SUM PT GR RES  =   L243
ACTUAL SUM PT GR NRES =   L244
***  OFF-CAMPUS (OFC) - FALL QUARTER HEADCOUNT
******    FULL-TIME (FT)
ACTUAL OFC FT UG RES  =   L251
ACTUAL OFC FT UG NRES =   L252
ACTUAL OFC FT GR RES  =   L253
ACTUAL OFC FT GR NRES =   L254
******    PART-TIME (PT)
ACTUAL OFC PT UG RES  =   L261
ACTUAL OFC PT UG NRES =   L262
ACTUAL OFC PT GR RES  =   L263
ACTUAL OFC PT GR NRES =   L264
```

Figure 4.8: Secondary enrollment data file format (file CAENROLL IFPS).

is because the actual tuition charge structure is normally set 1-1/2 to 2-1/2 years after the recovery rate is determined. The first type of enrollment data file maintained by the DSS, referred to as "projected", is used to calculate the tuition and fee recovery rate; the data files are named PENROLL1, PENROLL2, etc. The second type of data file, referred to as "actual", is used to calculate the student fees and to establish the tuition charge structure; the data files are named AENROLL1, AENROLL2, etc.

Since there are two types of enrollment data files, it follows that there are three "enrollment basis" options available for specifying how the data is used in the model's calculations: (1) use the projected data for all calculations, (2) use the actual data for all calculations, or (3) use the projected data for the recovery rate calculations and the actual data to set the tuition charge structure and the student fees.

The IFPS model that performs the calculations contains two sets of variables - one for the projected data and one for the actual data. The values for each set of variables are contained in each of the two types of files. For the first "enrollment basis" option, the actual variables are set equal to the projected values by using the actual enrollment data file CAENROLL. Conversely, the second

option sets the projected variables equal to the actual values by using the enrollment data file CPENROLL. Both CAENROLL and CPENROLL use the secondary data file format. This means that each variable in the file is defined by an IFPS line number instead of a specific enrollment value. For example, in Figure 4.8 all of the values for the actual variables (in the data file CAENROLL) are specified by line numbers; the line numbers correspond to the variables that define the projected enrollment values. Therefore, the IFPS model uses the projected values not only to calculate the recovery rate but to calculate student fees and to set the tuition charge structure.

4.3 THE DIALOG SUBSYSTEM

The dialog subsystem is probably the most important since it provides the interface between the user and the system. Bennett [3] includes the user, the computer terminal, and the software system as parts of the dialog subsystem. He also defines the dialog experience as containing (1) action language - the user's communication with the system, (2) display or presentation language - what the user sees on the display screen of the terminal, and (3) knowledge base - what the user must know in order to use the system effectively. The dialog subsystem is integrated

with the model subsystem in order to control the models' operation and their data sources. It is linked with the data subsystem to provide the user with displays, reports, and memory aids and to store the user's input values in data files.

The DSS's dialog subsystem, as represented in Figure 4.1, contains a single executive control program (ECP) and a series of component control programs (CCPs). The executive control program is the primary link between the user and the DSS. It manages the user's session with the DSS through the display of available options and through control mechanisms that initiate operations when certain conditions exist. Changes made by the user cause "flags" or indicators to be set, which ensure certain operations are performed at later steps in the process. For example, a change in the definition of the planning horizon causes operations to be performed both by the system (change the problem framework data file) and by the user (enter new goal target values). The executive control program also provides the user with direct access to the common and shared data files. Most of the interface between the models and the data is provided by the component control programs. The relationships between the ECP and the CCPs and the functions performed by each are outlined in Figure 4.9. This figure provides an overview of

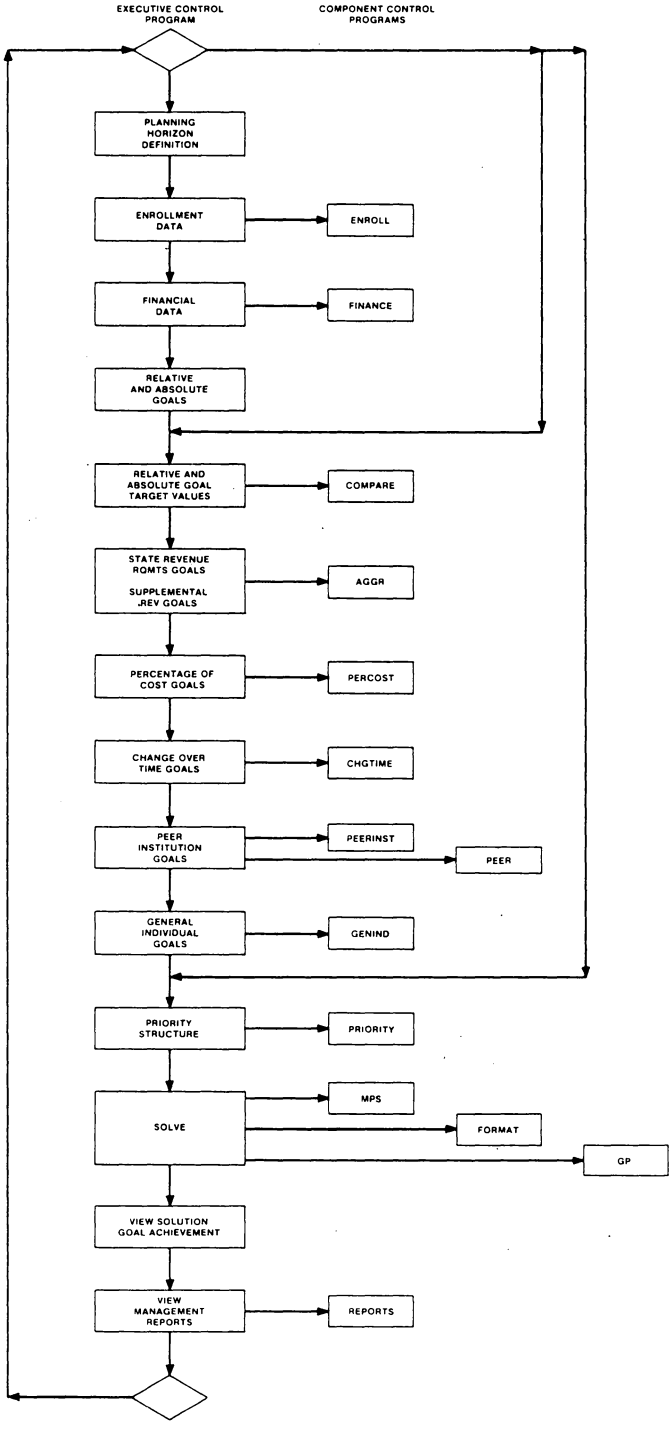


Figure 4.9: The relationship between the executive control program (ECP) and the component control programs (CCPs).

the ECP/CCP interactions; it is a "condensed" version of Figures 4.2 through 4.5.

The DSS contains fifteen component control programs that perform five basic functions: data control, goal formulation, priority structure definition, problem solution, and report generation. Three are used to handle the data files, especially those containing the enrollment and financial data. Seven programs are used to formulate the goals by : (1) prompting the user for information when needed, (2) utilizing the appropriate models and data to estimate the parameter and goal target values, (3) translating the goals to the proper format for the GP solution algorithm, and (4) generating reports to document the results and supply supporting information. Another component program is used to contol the input of the priority structure by the user and to translate it to the proper format for the GP model. Three of the component control programs are used to solve the GP model - one is used to setup the problem in a file that is readable by the solution routine; another provides a listing of the goal programming model in equation form; the third program contains the commands necessary to generate a solution and

store the results. Another CCP is used to provide the user with a menu of reports that are available for printing and a choice of the type of printer to be used.

The executive control program and all of the component control programs are discussed in the following sections. The programs are described in the order that they are accessed in a typical session. The order of the presentation also parallels the flow through the process as illustrated in Figures 4.2 through 4.5.

4.3.1 Executive Control Program DSS

All of the steps in the process of establishing tuition charges using goal programming and the DSS are controlled by a single program - the Executive Control Program (ECP) - named DSS. The ECP's logic is illustrated in Figures 4.2 through 4.5; all of the operations not enclosed by the dashed-line boxes are performed by the ECP. As can be seen from the figures, the ECP serves (1) as an integrator of the three subsystems - dialog, data, and models, (2) as the primary interface between the DSS and the user, and (3) as a controller to assign tasks and operations to the appropriate component programs.

The first section of the ECP defines a set of parameters that are used throughout the session. The first

set of parameters include: the default value for the dorm-and-dining calculation option (2), the number of reports that are available for printing (16), the earliest year for which data are available (1978), the maximum length of the planning horizon, in years (6), and the unit number used to identify the file containing the goal programming model in MPS format (08). The second set consists of six "flags" or indicators that are used to control the flow through the system; they are initialized to zero. The final set of parameters define the framework of the problem that is currently in effect. These values are read into the ECP from the file SETUP DSS.

Before proceeding with the session, the DSS checks to see if there is enough disk space available on the user's A-disk to execute all of the routines. If there is enough space, the session continues; if there is not enough space, a message is generated that informs the user that there may not be enough space to run the entire session. The user has the option to either continue the session or to quit and acquire additional disk space.

Subsequently, the user is provided with three options for starting the session - either start at the beginning so that the data can be reviewed or changed ("step"), at the point where the goal target values are set ("target"), or

where the priority structure is established ("priority"). These options facilitate the consideration of alternatives and the performance of sensitivity analyses by bypassing, when desired, steps in the process that remain unchanged. The discussion of the ECP that follows is based on the use of the "step" option, since it is the most comprehensive.

It provides the user with the opportunity to view or change the basic data, set the goal targets, establish the priority structure, solve the model, review the results, and print any of the reports.

The DSS displays the planning horizon variables on the terminal screen and the user is given the opportunity to change them. The planning horizon is defined by two variables - the starting year and the number of years that are to be included in the analysis. If either or both variables are changed, an indicator is set that ensures certain other changes are made at later points in the session. A change in the definition of the planning horizon requires the user to re-enter the goal target values for the following: relative and absolute goals, supplemental revenue requirements, percentage-of-average-total-cost goals, general individual goals, and change-in-tuition-charges-over-time goals. Also the following must be re-calculated: state-revenue-requirement decision-variable coefficients and

goal target values, percentage-of-average-total-cost goal coefficients, student fees and dorm-and-dining charges, and the peer institution goal target values.

Prior to changing the planning horizon definition in the file SETUP DSS, the ECP checks to see that the planning horizon is not defined for more years than the maximum allowed and that it is not defined beyond the latest year allowed. If the planning horizon is incorrectly defined, an error message is displayed informing the user of this fact. The definition is then re-entered and if it is within the prescribed bounds, the new values for the plannning horizon variables are stored in the file SETUP DSS.

The user is provided the option to review the enrollment or financial data. If the enrollment data is to be displayed or changed, control is passed to the component control program ENROLL. The CCP FINANCE is accessed if the user wants to view or change the financially-oriented data.

The process of selecting which relative and absolute goals to include in the goal programming model is managed by the ECP. As shown in Figure 4.3, there are three options available for specifying the goals. The "no" option is used to bypass this step. The "display" option allows the user to view only the goals that are in effect. Any changes made through the "display" option are ignored by the system; changes can only be made through the "change" option.

The "change" option is used to add or delete goals. If this option is used, all of the relative and absolute goal target values and the priority structure must be re-entered. The user selects which goals to include from a list stored in the file COMPGEN. The file is generated by the FORTRAN program COMPGEN.

As shown by an example of the file COMPGEN in Figure 4.10, four type of goals are included: relative goals, single-category absolute goals, pairwise comparison absolute goals, and four-category absolute goals. Recall that these goals are defined and discussed in detail in Chapter 3, sections 3.3.3 and 3.3.4. Since the relative goals and the pairwise absolute goals both contain two student categories in each goal equation, the list of possible goals in COMPGEN for these goal types include all of the possible pairwise combinations of the student categories. For the single-category goals, the file lists a possible goal for each student category. The user can specify up to ten four-category goals in COMPGEN.

In order to include a particular goal in the formulation, the user changes the "N" in the first column of the line describing each goal to "Y". There is a provision for including a particular goal in the formulation more than once. This is necessary if a goal target value is specified

```
                                COMPGEN  DSS

          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
        DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
                ALL OF THE POSSIBLE COMPARISON GOALS

RELATIVE-COMPARISON GOALS
Y       2  1        NRES_UG      /  RES_UG       = VALUE
Y       3  1        RES_GR       /  RES_UG       = VALUE
N       4  1        NRES_GR      /  RES_UG       = VALUE
N       5  1        RES_GR_OFC   /  RES_UG       = VALUE
N       6  1        NRES_GR_OFC  /  RES_UG       = VALUE
N       3  2        RES_GR       /  NRES_UG      = VALUE
N       4  2        NRES_GR      /  NRES_UG      = VALUE
N       5  2        RES_GR_OFC   /  NRES_UG      = VALUE
N       6  2        NRES_GR_OFC  /  NRES_UG      = VALUE
N       4  3        NRES_GR      /  RES_GR       = VALUE
N       5  3        RES_GR_OFC   /  RES_GR       = VALUE
N       6  3        NRES_GR_OFC  /  RES_GR       = VALUE
N       5  4        RES_GR_OFC   /  NRES_GR      = VALUE
N       6  4        NRES_GR_OFC  /  NRES_GR      = VALUE
N       6  5        NRES_GR_OFC  /  RES_GR_OFC   = VALUE

SINGLE-CATEGORY ABSOLUTE GOALS
N 1     1           RES_UG                       = VALUE
N 1     2           NRES_UG                      = VALUE
N 1     3           RES_GR                       = VALUE
N 1     4           NRES_GR                      = VALUE
N 1     5           RES_GR_OFC                   = VALUE
N 1     6           NRES_GR_OFC                  = VALUE

PAIRWISE-COMPARISON ABSOLUTE GOALS
N 2     2  1        NRES_UG      -  RES_UG       = VALUE
N 2     3  1        RES_GR       -  RES_UG       = VALUE
N 2     4  1        NRES_GR      -  RES_UG       = VALUE
N 2     5  1        RES_GR_OFC   -  RES_UG       = VALUE
N 2     6  1        NRES_GR_OFC  -  RES_UG       = VALUE
N 2     3  2        RES_GR       -  NRES_UG      = VALUE
N 2     4  2        NRES_GR      -  NRES_UG      = VALUE
N 2     5  2        RES_GR_OFC   -  NRES_UG      = VALUE
Y 2     4  3        NRES_GR_OFC  -  NRES_UG      = VALUE
Y 2     4  3        NRES_GR      -  RES_GR       = VALUE
Y 2     5  3        RES_GR_OFC   -  RES_GR       = VALUE
N 2     6  3        NRES_GR_OFC  -  RES_GR       = VALUE
N 2     5  4        RES_GR_OFC   -  NRES_GR      = VALUE
N 2     6  4        NRES_GR_OFC  -  NRES_GR      = VALUE
N 2     6  5        NRES_GR_OFC  -  RES_GR_OFC   = VALUE

FOUR-CATEGORY ABSOLUTE COMPARISON GOALS:
USING THE NUMBERS THAT CORRESPOND TO THE STUDENT CATEGORIES LISTED BELOW
CONSTRUCT FOUR-CATEGORY ABSOLUTE GOALS.  FOR EXAMPLE, TO  MAINTAIN THE
SAME DIFFERENTIAL CHARGE BETWEEN RESIDENT  AND NON-RESIDENT GRADUATE
STUDENTS, BOTH ON-CAMPUS AND OFF-CAMPUS, ENTER +4 -3 -6 +5.
THIS CORRESPONDS TO (NRES_GR - RES_GR) - (NRES_GR_OFC - RES_GR_OFC)
Y 4     +4 -3 -6 +5
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0
N 4     +0 -0 -0 +0

NUMBER  CATEGORY
   1     RES_UG
   2     NRES_UG
   3     RES_GR
   4     NRES_GR
   5     RES_GR_OFC
   6     NRES_GR_OFC
```

Figure 4.10: Display for specifying which relative and absolute goals to include in the model (file COMPGEN).

as a range rather than as a single value; e.g., the non-resident graduate charge should be between $10 and $20 per credit hour more than the resident graduate charge. Multiple goals are included by changing columns eight and eleven in the file COMPGEN. These numbers identify the student categories that are included in the goal. Each student category and its corresponding identification number are defined at the bottom of Figure 4.10. Notice that two pairwise comparison absolute goals, that include student categories three and four have been included in the model displayed in Figure 4.10.

To include four-category goals in the formulation, the user must specify the identification numbers that correspond to each of the student categories and the appropriate sign. An example is included in Figure 4.10. The goal selections are stored in the the file COMPGEN by pressing the PF2 key.

Once the relative and absolute goals have been selected, the ECP determines (from the file COMPGEN) which goals are included in the formulation and stores that updated information in the problem framework file SETUP DSS, as shown in Figure 4.2. The goal target values are established at a later point in the session through the CCP

COMPARE. Only the goals selected in this step of the process are included in the goal-target-value file, not all of the possible goals that are listed in the file COMPGEN.

Most of the other activities performed by the ECP involve calls to the appropriate CCP, based on either a request by the user or on the status of a certain indicator. As shown in Figure 4.9, all of the goals are formulated and the priority structure is established in CCPs, which are discussed in detail in subsequent sections. Before establishing the goals, the ECP calls the CCP SETUP to create two files: the first defines the problem framework in an IFPS-readable format; the second creates the MPS-format file used by the optimization routine.

When the model is ready to be solved, the ECP calls CCP MPS to establish the input file for the optimization routine and CCP GP to solve the problem as a sequence of linear programs using the LINDO software system. Once the problem has been solved, control is returned to the ECP and the solution values, stored in the file TUIT, are displayed on the computer terminal's screen. When the user has finished viewing the solution values, pressing the PF2 key stores the solution values and provides the user with the option of viewing a report that summarizes the level of goal achievement attained by the present solution. The report is

stored in the file DEVVAR. Through a call to the CCP REPORTS, the ECP provides the user with a menu of sixteen reports that are available for hardcopy printing. The final activity performed by the ECP is to see if the user wants to perform another analysis. If the response is yes, the process starts over at the beginning, at the point where the indicators are initialized; if the response is no, control is returned to the operating system VM/CMS.

4.3.2 Component Control Program ENROLL

Once the planning horizon has been established in the ECP, the user has the option of accessing the enrollment data. If the definition of the planning horizon has changed this session, a message is generated reminding the user of this fact and suggests that the enrollment data be reviewed.

If the user chooses to access the enrollment data, control is passed to the CCP ENROLL, along with the names of the enrollment files that are currently in effect. As described in the data subsystem section, there are three "enrollment basis" options for specifying which type of enrollment data to use in the calculations; either the actual data is set the same as the projected, or the projected data is set the same as the actual, or both are different. The ECP passes the names of the enrollment data

files that are currently in effect to CCP ENROLL. The CCP determines which "enrollment basis" option is in effect by checking the first character of each data file name. If the first character of the actual data file name is "C", then Option 1 is in effect; if the first character of the projected data file is "C", then Option 2 is in effect; and, if neither data file name starts with a "C", then Option 3 is in effect. The naming convention for enrollment data files was explained in the data subsystem section (section 4.2.2). At this point, the option that is currently in effect is displayed on the screen and the user has the opportunity to change it, if desired.

Subsequently, as shown in Figure 4.2, the user has the choice of either displaying or changing the enrollment data file that is in effect, selecting another data file, or creating a new data file. The user chooses one of these alternatives from an option menu displayed on the terminal screen. If the "enrollment basis" option is changed then either the select or the create option must be used.

The "display" option is used to view or change the enrollment data file that is currently in effect. It is automatically displayed when this option is selected and changes are made directly to the file on the display screen. When all of the changes have been made, pressing the PF2 key

stores the updated file and returns the option menu to the screen.

The "select" option provides a list of all of the available actual or projected data files. This list is generated when the CCP executes the CMS FILELIST command. The contents of each file is examined by placing the cursor in front of the desired file name and pressing the PF11 key (XEDIT). After viewing the desired file the PF3 key is used to return to the list of files. If PF3 is used while the list of files is on the display screen, the process is returned to CCP ENROLL and a prompt requests the user for the file number that is to be used. An error message is displayed if either the specified file does not exist or if anything but a number is entered; the desired file number must then be re-entered. Once this process is completed, the option menu is returned to the screen.

The "create" option allows the user to generate a new data file. The new file is a copy of some existing file. Once the new file is established it can be modified by using the "display" option. In order to create a new enrollment data file the user supplies a number for the new file and the number of the existing file that is to be used as the basis for the new file. An error message is generated if the number specified for the new file already exists or if

the file specified as the basis for the new file does not exist. Once the process is completed, the option menu is returned to the screen.

In both the "select" and "create" options, the new enrollment data file name is recorded in the problem framework file, SETUP DSS.

The "no" option completes the process of specifying the enrollment data file(s). It also provides the user with option of viewing or changing such enrollment factors as the number of credit hours to be considered full-time for fee purposes, average student credit hour information, student retention factors between sessions, the fraction of students receiving tuition waivers, and graduate student employment rates. This information is stored in the file FACTOR; an example of this file is provided in Appendix E.6 .

Once the enrollment factor data has been displayed, control is returned back to the ECP. Subsequently the new enrollment data file names are read into the ECP from the problem framework file SETUP DSS. Also the ECP sets a "flag" indicating that the enrollment data has changed this session. The following variables are a function of enrollment and therefore must be re-calculated whenever the enrollment data changes: state revenue requirement goal coefficients and target values, student fees, average cost per student, and the recovery rate.

4.3.3 Component Control Program FINANCE

After the enrollment values are established, the user has the option of accessing the financial data - including base-year tuition charges, basic cost information, student fees, dorm-and-dining charges, and capital projects. If either the planning horizon or the enrollment data changed this session, a message is generated reminding the user that this has occurred and suggests that the financial data be reviewed.

If the user chooses to access the financial data, control is passed to the CCP FINANCE. As shown in Figure 4.2, the CCP provides the option of viewing or changing five sets of financially-related data files. An example of each data file is provided in Appendix E. In all cases the data are entered directly on the terminal screen and once all of the changes have been made in a particular data file, pressing PF2 (file) stores the changes and returns the option menu to the screen.

Historical tuition charges, expressed in dollars per credit-hour for each student category, are stored in the file TUITHIST (an example is provided in Appendix E.15). They are used in the change-over-time goal target calculations and as a point of reference in many of the reports. A message is displayed to remind the user that the base-year tuition values should be reviewed.

The cost file COST, as shown in Appendix E.2, includes the consumer price index (CPI), the institution's operating expenses (adjusted instructional E&G), and the tuition and fee recovery rate. It also contains many of the factors used in the recovery rate calculations: undergraduate and graduate cost per student, policy rates by student category, and the maximum allowable (by state policy) year-to-year percentage change in the recovery rate. If the recovery rate is entered as zero, then the calculated value is used to generate the state revenue requirement; on the other hand, if the rate is directly specified then the specified value overrides the calculated value.

The CPI is used to project tuition charges at peer institutions and can be used used to project fees and other charges. The reason that the CPI was chosen over other economic indexes, e.g. the Higher Education Price Index (HEPI), was that the CPI is readily accessible and more importantly, published projections are available [75]. The historical CPI values stored in the file COST relate to September of the specified year (when most changes in tuition charges are effective) and are based on 1967 equal to 100.

All of the data pertaining to student fees are stored in the file FEEBASE; an example of which is provided in

Appendix E.7. Four categories of student fees are included in this model: athletic, bus, student center, and health services. Four inputs for each fee category are required in each year; they include the quarterly fees itself, the annual operating expenses on which the charge is based, an economic escalator, and the proportion of part-time students that are assessed the charge. It is assumed that all full-time on-campus students and only a portion of part-time on-campus students pay the student fees. It is also assumed that off-campus students will not pay any of the fees.

There is a hierarchy of options available for calculating student fees. First, the quarterly fee may be explicitly specified. If a value of zero is specified then the fee is calculated based on annual operating expenses and enrollment. If the operating expenses also have a value of zero, then the fee is projected using the appropriate escalator. Finally, if the escalator is entered as a negative value, the CPI is used as the index to project future fees.

There are two options for calculating the dorm-and-dining charges - either directly input the charges for each dorm group or input the overall operating expenses and have the charges for each group allocated based on a set of user-specified ratios. Since many of the dormitories vary in

terms of quality, there is a different charge for each dorm group.

The first option requires the direct specification of the quarterly dorm charge and the number of students residing in each dorm group. The quarterly charge for each type of meal plan is also directly specified. The data file that is used is DDBASE1, an example of which is provided in Appendix E.3.

The overall operating expenses for the dorm and dining systems are input in Option two. The number of students and a charge index are also specified for each dorm group. The data are specified through the file DDBASE2, an example of which is provided in Appendix E.4. The charge index is used to establish the quarterly charge for each dorm group based on an allocation of the overall operating expenses. The same concept is used in establishing the quarterly charge for meal plans.

In both options, if either the quarterly charge for each dorm group or the annual operating expenses are entered as zero, then future charges are projected using the appropriate escalator. If the escalator has a negative value, then the CPI is used as index to project future values. The calculations are defined in detail later in this chapter, in section 4.4.5.

The last financially-oriented data file, CAPGEN, contains information on the capital projects that affect either student fees or dorm-and-dining charges. The file contains the following information for up to three projects in each of the fee categories: a project identifier, the amortized loan amount, the unamortized loan amount, the interest rate, the project life, the number of payments per year, and the year that the project begins.

An IFPS model is used to determine the quarterly charge per student for the capital projects and add that amount to the student fees described earlier. Details of the fee calculations are provided later in this chapter, in section 4.4.4. The file containing the basic capital projects data (CAPGEN) is designed for user input; it is not in a format that is conducive to IFPS. Therefore, as shown in Figure 4.2, a FORTRAN program (CAPIFPS) is used to create an IFPS-readable file, CAPIFPS, from the user-input file CAPGEN. An example of both files is shown in Figure 4.11. The CAPGEN file can accomodate up to three projects for each fee category. File CAPIFPS contains only the projects for which data is supplied; it also stores the information in terms of IFPS variables.

Once the capital projects data are established, control is returned to the ECP. At this point the ECP sets a "flag"

CAPGEN DSS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
CAPITAL PROJECTS DATA RELATED TO STUDENT FEES

ATHLETIC ASSOCIATION			
PROJECT IDENTIFIER	XXXXXX	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	00000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0000	0.0000	0.0000
PROJECT LIFE - YEARS	00.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1900	1900	1900
BUS SYSTEM			
PROJECT IDENTIFIER	XXXXXX	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	00000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0000	0.0000	0.0000
PROJECT LIFE - YEARS	00.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1900	1900	1900
STUDENT CENTER			
PROJECT IDENTIFIER	SQ ADD	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	12500.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0950	0.0000	0.0000
PROJECT LIFE - YEARS	20.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1984	1900	1900
HEALTH SERVICE			
PROJECT IDENTIFIER	INFIRM	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	01000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0925	0.0000	0.0000
PROJECT LIFE - YEARS	20.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1985	1900	1900
DORM AND DINING			
PROJECT IDENTIFIER	NEW UG	WAREHS	XXXXXX
AMORTIZED AMOUNT (000)	02500.	04000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0800	0.1000	0.0000
PROJECT LIFE - YEARS	20.000	15.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1983	1984	1900

CAPIFPS IFPS

```
*       VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*    DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*          CAPITAL PROJECTS DATA RELATED TO STUDENT FEES
SC1 AMORT    = 12500.0000
SC1 UNAMORT  =     0.0
SC1 INT RATE =     0.0950
SC1 PR LIFE  =    20.0000
SC1 PPY      =     1.0000
SC1 START    =     7.0000
HS1 AMORT    =  1000.0000
HS1 UNAMORT  =     0.0
HS1 INT RATE =     0.0925
HS1 PR LIFE  =    20.0000
HS1 PPY      =     1.0000
HS1 START    =     8.0000
DD1 AMORT    =  2500.0000
DD1 UNAMORT  =     0.0
DD1 INT RATE =     0.0800
DD1 PR LIFE  =    20.0000
DD1 PPY      =     1.0000
DD1 START    =     6.0000
DD2 AMORT    =  4000.0000
DD2 UNAMORT  =     0.0
DD2 INT RATE =     0.1000
DD2 PR LIFE  =    15.0000
DD2 PPY      =     1.0000
DD2 START    =     7.0000
```

Figure 4.11: Capital projects data files: CAPGEN for user input and CAPIFPS in IFPS-readable format.

to indicate that the financial data has changed and that the following must be re-calculated: state revenue requirement goal coefficients and target values, student fees, dorm-and-dining charges, average cost per student, and the recovery rate.

4.3.4 Component Control Program SETUP

This CCP has two functions: define the problem framework in IFPS format and create an MPS format file for the optimization routine. Both functions are accomplished by executing FORTRAN programs. Each program requires only one input source - the data file SETUP DSS, containing the definition of the problem framework.

The FORTRAN program, SETUP, creates the data file SETUP IFPS, as shown in Appendix E.13. It contains a set of IFPS MICRO definitions that establish the column statements and student category names used by all of the IFPS models and reports. The column statements depend on the number of years in the planning horizon and the starting year.

The FORTRAN program, ROWS, establishes the data file FILE FT08F001. It contains the goal programming model formulation in MPS format. Recall that the MPS format is a widely accepted method for storing mathematical programming problems. At this point in the session, the file is created

and only the "ROWS" section is generated. The ROWS section gives the name and the equality or inequality type of each row in the model formulation. The other two MPS sections, COLUMNS and RHS, are added to the file at a later point (section 4.3.13).

4.3.5 Component Control Program COMPARE

As shown in Figure 4.2, the relative and absolute goal target values are established by a call to the CCP COMPARE. Since the values are directly specified and not calculated, they must be re-entered each time the planning horizon changes. A "flag" is set in the ECP when the planning horizon is changed indicating that the goal target values must be regenerated.

The relative and absolute goal targets are also regenerated if the user added or deleted any of the goals from the problem formulation. Recall that relative and absolute goals are added or deleted from the model formulation by using the "change" option in the previously discussed process in the ECP for specifying which goals to include.

The file that stores the relative and absolute goal targets, COMPSET, is generated by the FORTRAN program COMPSET. The program creates a file that contains a

description of each goal that is included in the model and a default value (zero) for each goal in each year of the planning horizon. The goals that are included in the model formulation are obtained from the problem framework file, SETUP DSS. The file COMPSET is displayed on the terminal screen and the goal target values are directly specified by the user. After the values have been entered, pressing the PF2 key automatically stores the values and converts the relative and absolute goals to MPS format. The file COMPSET is not only a data file, it is available for printing as a source of backup information.

If the file is not regenerated, the user views or changes the relative and absolute goal targets by using the "display" option. Changes are made directly to the file COMPSET displayed on the terminal screen.

Once the goal target values are set, the relative and absolute goals are transferred to MPS format by the FORTRAN program COMPMPS. Using a pairwise comparison goal as an example, Figure 4.12 shows a goal in written form, in equation form, as it is represented in the input file COMPSET, and finally in MPS format.

As illustrated in Figure 4.2, COMPMPS obtains input data from two files, SETUP DSS and COMPSET. SETUP DSS specifies the number of goals that are included in the

Goal specified verbally

The non-resident graduate charge should be $5.00, $8.00, and $10.00 per credit hour greater than the resident graduate charge in years 1, 2, and 3, respectively.

Goal specified in equation form

$$
\begin{aligned}
-X_{31} + X_{41} + n_{85} - p_{85} &= 5.00 \\
-X_{32} + X_{42} + n_{86} - p_{86} &= 8.00 \\
-X_{33} + X_{43} + n_{87} - p_{87} &= 10.00
\end{aligned}
$$

Goal as specified by the user in the DSS input file COMPSET

```
VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
COMPARISON GOAL TARGET VALUES

RELATIVE COMPARISON GOALS

  2  1        NRES_UG     /  RES_UG      = 2.00000    FOR YEAR 1983
                                         = 2.00000    FOR YEAR 1984
                                         = 2.00000    FOR YEAR 1985

  3  1        RES_GR      /  RES_UG      = 1.20000    FOR YEAR 1983
                                         = 1.20000    FOR YEAR 1984
                                         = 1.20000    FOR YEAR 1985

SINGLE-CATEGORY ABSOLUTE GOALS

PAIRWISE-COMPARISON ABSOLUTE GOALS

  4  3        NRES_GR     -  RES_GR      = 5.00000    FOR YEAR 1983
                                         = 8.00000    FOR YEAR 1984
                                         = 10.0000    FOR YEAR 1985

  4  3        NRES_GR     -  RES_GR      = 10.0000    FOR YEAR 1983
                                         = 16.0000    FOR YEAR 1984
                                         = 20.0000    FOR YEAR 1985

  5  3        RES_GR_OFC  -  RES_GR      = 10.0000    FOR YEAR 1983
                                         = 15.0000    FOR YEAR 1984
                                         = 20.0000    FOR YEAR 1985

FOUR-CATEGORY ABSOLUTE COMPARISON GOALS

  4 -3 -6  5    NRES_GR     -  RES_GR      - NRES_GR_OFC +  RES_GR_OFC
                                         = 0000000    FOR YEAR 1983
                                         = 0000000    FOR YEAR 1984
                                         = 0000000    FOR YEAR 1985
```

Goal specified in MPS format

COMPX DSS

X21	ROW10080	1.000
X11	ROW10080	-2.000
X22	ROW10081	1.000
X12	ROW10081	-2.000
X23	ROW10082	1.000
X13	ROW10082	-2.000
X31	ROW10083	1.000
X11	ROW10083	-1.200
X32	ROW10084	1.000
X12	ROW10084	-1.200
X33	ROW10085	1.000
X13	ROW10085	-1.200
X41	ROW10086	1.000
X31	ROW10086	-1.000
X42	ROW10087	1.000
X32	ROW10087	-1.000
X43	ROW10088	1.000
X33	ROW10088	-1.000
X41	ROW10089	1.000
X31	ROW10089	-1.000
X42	ROW10090	1.000
X32	ROW10090	-1.000
X43	ROW10091	1.000
X33	ROW10091	-1.000
X51	ROW10092	1.000
X31	ROW10092	-1.000
X52	ROW10093	1.000
X32	ROW10093	-1.000
X53	ROW10094	1.000
X33	ROW10094	-1.000
X41	ROW10095	1.000
X31	ROW10095	-1.000
X61	ROW10095	-1.000
X51	ROW10095	1.000
X42	ROW10096	1.000
X32	ROW10096	-1.000
X62	ROW10096	-1.000
X52	ROW10096	1.000
X43	ROW10097	1.000
X33	ROW10097	-1.000
X63	ROW10097	-1.000
X53	ROW10097	1.000

COMPNP DSS

N079	ROW10080	1.000
P079	ROW10080	-1.000
N080	ROW10081	1.000
P080	ROW10081	-1.000
N081	ROW10082	1.000
P081	ROW10082	-1.000
N082	ROW10083	1.000
P082	ROW10083	-1.000
N083	ROW10084	1.000
P083	ROW10084	-1.000
N084	ROW10085	1.000
P084	ROW10085	-1.000
N085	ROW10086	1.000
P085	ROW10086	-1.000
N086	ROW10087	1.000
P086	ROW10087	-1.000
N087	ROW10088	1.000
P087	ROW10088	-1.000
N088	ROW10089	1.000
P088	ROW10089	-1.000
N089	ROW10090	1.000
P089	ROW10090	-1.000
N090	ROW10091	1.000
P090	ROW10091	-1.000
N091	ROW10092	1.000
P091	ROW10092	-1.000
N092	ROW10093	1.000
P092	ROW10093	-1.000
N093	ROW10094	1.000
P093	ROW10094	-1.000
N094	ROW10095	1.000
P094	ROW10095	-1.000
N095	ROW10096	1.000
P095	ROW10096	-1.000
N096	ROW10097	1.000
P096	ROW10097	-1.000

COMPRHS DSS

RHS	ROW10080	0.0
RHS	ROW10081	0.0
RHS	ROW10082	0.0
RHS	ROW10083	0.0
RHS	ROW10084	0.0
RHS	ROW10085	0.0
RHS	ROW10086	5.000
RHS	ROW10087	8.000
RHS	ROW10088	10.000
RHS	ROW10089	10.000
RHS	ROW10090	16.000
RHS	ROW10091	20.000
RHS	ROW10092	10.000
RHS	ROW10093	15.000
RHS	ROW10094	20.000
RHS	ROW10095	0.0
RHS	ROW10096	0.0
RHS	ROW10097	0.0

Figure 4.12: Example of a goal represented in its four forms.

formulation by type (number of relative goals, number of single-category absolute goals, etc.), and the number of years in the planning horizon. COMPSET specifies which decision variables are included in which goal, its coefficient, and the goal target value. Again using the pairwise comparison goal in Figure 4.12 as an example, the FORTRAN program COMPMPS interprets the " 4 3 " in columns three and six of COMPSET as decision variables four and three (non-resident graduate and resident graduate, respectively). Since the goal is formulated as a difference, order is important - decision variable four has a positive coefficient, decision variable three a negative coefficient. The goal target value in each of the three years is obtained from columns 44-50 in the file COMPSET.

The relative and absolute goals are stored in MPS format in three files, as illustrated in Figure 4.12. The decision variables are stored in the file COMPX, the deviation variables in COMPNP, and the goal targets or right-hand-side values are stored in COMPRHS.

4.3.6 Component Control Program AGGR

Recall that there are two types of aggregate goals - the state revenue requirement and the supplemental revenue requirement. The coefficient of the decision variables and

the goal target values for the state revenue requirement goal are calculated and do not require any input from the user. The only information directly provided by the user in establishing the aggregate goals are the target values for the supplemental revenue requirements.

Six parameters are passed from the ECP to CCP AGGR, three of which are "flags" or indicators used to control certain operations. One indicator ensures that the state revenue requirement decision-variable coefficients and goal target values are re-calculated when either the planning horizon, enrollment data, or cost data are changed. A second indicator regenerates the supplemental revenue input file whenever the planning horizon is changed. The third indicator is used to translate the aggregate goals to MPS format whenever any relative of absolute comparison goals are either added or deleted from the formulation. This is necessary because of the way the general goal programming model is structured. The relative and absolute goals are the next-to-last set of goals in the formulation and the overall supplemental revenue requirement goal is the last. Therefore, if any relative or absolute goals are added or deleted, the overall supplemental revenue goal's row number (for MPS purposes) and deviation-variable number will change. The final three parameters that are passed from the

ECP to CCP AGGR include: the projected enrollment data file name, the actual enrollment data file name, and the dorm-and-dining calculation option number.

As shown in Figure 4.3, all of the necessary calculations are made using IFPS. Three sets of calculations are performed, each requiring a call to the "IFPS System". In each case an IFPS command file is used to coordinate the process by defining which IFPS models, data files, and report generators are to be used.

The first set of IFPS calculations process the enrollment data and generate the decision-variable coefficients for the state revenue requirement goal. As shown at the top of Figure 4.3, four files are created - two are management reports and two are data files. One of the management reports summarizes the enrollment data used (file ENROLOUT); the other summarizes the credit-hour data (file CREDOUT). The data file ENRLWORK contains the solution of the enrollment program for use by other IFPS models . The data file STATEREV stores the decision-variable coefficients for the state revenue goal.

As illustrated in Figure 4.3, two IFPS models are used to calculate the state revenue requirement goal target values and the decision-variable coefficients for the percentage-of-average-total-cost goals. A management report

is generated to summarize the cost data used in the analysis; it is stored in the file COSTINFO. The data file INDEXCHG is created to aid in establishing the change-in-tuition-charges-over-time goal targets. It contains, for each year, the percentage change in revenue required and the percentage change in the CPI. The state revenue goal target values are appended to the file that contains the decision-variable coefficients, STATEREV. The file PERCOST is created to store the coefficient of each decision variable in the percentage-of-average-total-cost goals.

The third IFPS routine calculates the student fees and the dorm-and-dining charges. A management report, stored in the file DDFEEOUT, summarizes these charges. The charges are also stored in abbreviated form in the file DDFEEMPS for use in the peer institution goal calculations.

The goal target values for the supplemental revenue requirements are stored in the file SUPLMREV (see Appendix E.14 for an example). Recall that a supplemental revenue goal may be specified for each year in the planning horizon and over the entire span of the planning period. Both types of goal values are directly specified by the user in file SUPLMREV. As shown by the indicator in Figure 4.3, the targets must be re-entered if the planning horizon is changed. When this condition occurs the file is regenerated

with default values of zero by executing the FORTRAN program SUPLMREV. Note also in Figure 4.3 that the file SUPLMREV is a data file and is available for printing as a backup report.

The aggregate goals are converted to MPS format by the FORTRAN program AGRGMPS. As illustrated in Figure 4.3, the program's input is obtained from three files. SETUP DSS defines the planning horizon and the number of goals included in the model formulation. The file STATREV provides the coefficients of the decision variables and the target values for the state revenue goals. Targets for the supplemental revenue goals are obtained from the file SUPLMREV. The decision-variable data is stored in MPS format in the file AGRGX; likewise, the deviation variables are stored in the file AGRGNP and the goal targets or right-hand-side values are stored in AGRGRHS.

4.3.7 Component Control Program PERCOST

The percentage-of-average-total-cost goals are established in the CCP PERCOST, as shown in Figure 4.3. The goal target values, stored in the file PERCOST2, are supplied directly by the user through the "display" option. An example of the input file PERCOST2 is provided in Appendix E.12. Whenever the planning horizon is changed an

indicator, passed from the ECP, causes the input file PERCOST2 to be regenerated. The FORTRAN program PCOSTGEN is called to regenerate the input file. The user replaces the default values (zero) with the desired goal target values. PERCOST2 is not only a data file; it also is available for printing through the CCP REPORTS as a backup report.

The percentage-of-cost goals are converted to MPS format by the FORTRAN program PERCOST. As illustrated in Figure 4.3, the program's input is obtained from three files. SETUP DSS defines the planning horizon and the number of goals included is the model formulation. The coefficients of the decision variables are obtained from the file PERCOST (created by IFPS in the CCP AGGR), and the goal target values are obtained from the file PERCOST2. The percentage-of-cost goals are stored in MPS format in three data files: PCOSTX contains the decision variables, PCOSTNP contains the deviation variables, and PCOSTRHS contains the target values.

4.3.8 Component Control Program CHGTIME

The process of establishing the change-in-tuition-charges-over-time goals is very similar to the previously defined process of establishing the percentage-of-average-total-cost goals. The process is illustrated in Figure 4.3.

The "display" option is used to review or change the goal target values stored in the file CHGTIME. An example of this file is provided in Appendix E.1. Prior to accessing CHGTIME and in order to assist the user in establishing the goal target values, the DSS automatically provides a display of three economic indicators - the percentage change in the institution's operating expenses and revenue requirements and the percentage change in the CPI - for each year in the planning horizon. The indicators are stored in the file INDEXCHG and are created by IFPS in the CCP AGGR.

The economic index information display is replaced by the file used to input the goal target values, CHGTIME, by pressing the PF2 key. The desired change in tuition charges for each student category in each year of the planning horizon is entered directly into the file CHGTIME. These values are then saved by pressing the PF2 key.

Whenever the planning horizon is changed an indicator is passed from the ECP that causes the input file CHGTIME to be regenerated. The FORTRAN program CTIMEGEN is used to generate a new file. The user replaces the default values (zero) with the desired goal target values. CHGTIME is not only a data file; it is also available for printing as a backup report.

The change-in-tuition-over-time goals are converted to MPS format by the FORTRAN program CHGTIME. As illustrated in Figure 4.3, the program's input is obtained from three files. SETUP DSS defines the planning horizon and the number of goals included in the model's formulation. The goal target values are obtained from the file CHGTIME. The base-year tuition charges for each student category are obtained from the file TUITHIST. The conversion to MPS format is stopped and an error message is generated if any of the tuition values are found to be zero. The user is warned that at least one base-year tuition charge is zero and the file TUITHIST is displayed on the screen so that the value(s) can be changed. Once all of the base-year values are non-zero, the conversion to MPS format continues.

Four data files are generated by the FORTRAN program CHGTIME. The decision variables, deviation variables, and goal target values are stored in MPS format in three of the files - CTIMEX, CTIMENP, and CTIMERHS, respectively. The fourth file, BASEYEAR, contains the base-year tuition charges which are used in several of the management reports.

4.3.9 Component Control Program PEERINST

The expected tuition and fee charges at peer institutions are established by the CCP PEERINST. If the planning horizon has changed this session, control is automatically passed from the ECP to PEERINST; otherwise, the user has the option of viewing or changing the values that are in effect.

There are two approaches to setting the peer institution goal targets, either by directly specifying the value or by estimating the expected charges. In this case estimating means projecting historical charges based on some index, such as the CPI.

The data sources available in the DSS for projecting charges at other institutions are: Land-Grant colleges and universities, NCHEMS 1A institutions[11] and Virginia four-year colleges and universities. These data sources include for each institution, the charges by student category for tuition and fees and for tuition, fees, room and board. In establishing the goal target values, the DSS converts the projected total charges to a dollar-per-credit-hour charge

---

[11] those major publicly-supported research universities, as indicated by the National Center for Higher Education Management Systems (NCHEMS), that "awarded over fifty doctor's degrees and that receives over ten million dollars in federal government support of the academic sciences in the specified year." (Virginia Tech IR Series, vol. 82, no. 35.)

for tuition only.

The tuition charges for each student category and for each data source are stored in a separate peer institution data file. There is a specific naming convention for the data files based on their contents. Table 4.4 defines the naming convention for the eight-character file name and provides an example. The first four characters of the file name identify the student category, the next two identify the campus location, the fifth and sixth characters identify the data source, and the last two characters identify whether the data include room and board charges.

The data in each peer institution file are arranged by percentile and year, as shown by the example in Figure 4.13. The first data value in the file, 5621, means that in 1978 99% of the Land-Grant institutions charged non-resident undergraduate students less than $5621 per year for tuition, fees, room and board.

The file PEERGEN, an example of which is presented in Figure 4.14, contains the information required to set the peer institution goals. As seen in Figure 4.14, the user must supply four data items for each student category. The first item is the number of credit hours on which the goal is based. For the example in Figure 4.14, the first four student categories are based on annual full-time (for fee

TABLE 4.4

Naming convention for the peer institution data files.

Position[1]	Identification	Possible values
1	Residency status	R = resident N = non-resident
2	Academic level	U = undergraduate G = graduate
3-4	Campus location	ON = on-campus OF = off-campus
5-6	Data source	LG = land grant NC = NCHEMS 1A VA = Virginia
7-8	Type of charge	= tuition and fees TO = tuition, fees, room & board

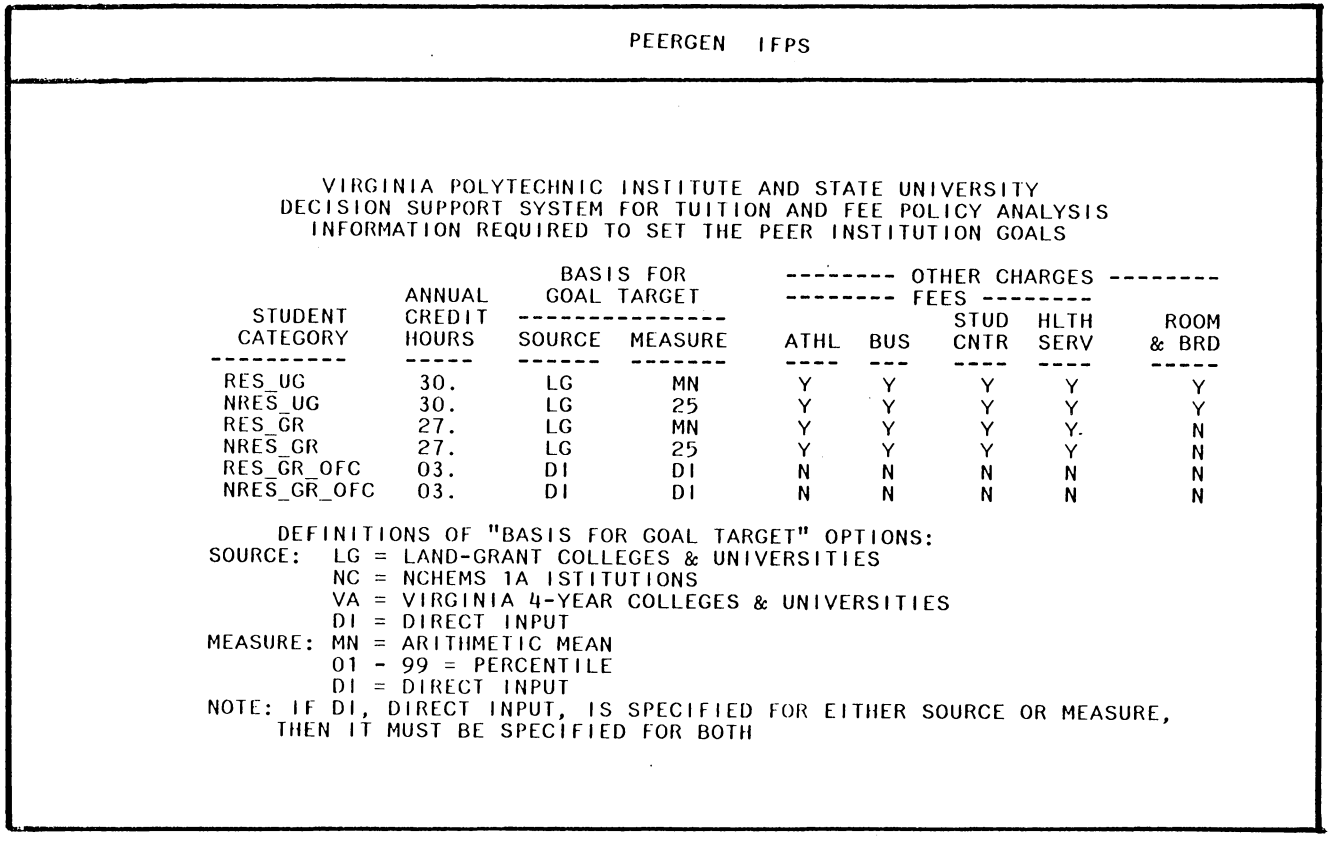
Example: The file name NUONLGTO means that the data are total charges - tuition, fees, room and board (TO) - for on-campus (ON) non-resident (N) undergraduate (U) students at Land-Grant colleges and universities (LG).

[1] character position in the eight-character file name.

NUONLGTO IFPS

```
*        VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*     DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                PEER INSTITUTION DATA FOR:
* NON-RES UNDERGRAD TUITION, FEES, RM & BRD AT LAND-GRANT INSTITUTIONS
*
*             1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988
CHARGE 99   = 5621,6199,6590,7540,8628,9300
CHARGE 95   = 5253,5625,6202,7160,7859,8730
CHARGE 90   = 4480,4893,5126,6248,7063,7806
CHARGE 75   = 4053,4385,4722,5494,6089,6526
CHARGE 50   = 3405,3657,3897,4400,4827,5477
CHARGE 25   = 2984,3095,3184,3806,4061,4494
CHARGE 10   = 2538,2756,2942,3384,3712,4102
CHARGE 05   = 2310,2619,2822,3305,3364,3700
CHARGE 01   = 1557,2535,2654,2990,3116,2727
MEAN        = 3482,3779,4011,4687,5198,5643
```

Figure 4.13: Example of a peer institution data file (file NUONLGTO).



PEERGEN IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
INFORMATION REQUIRED TO SET THE PEER INSTITUTION GOALS

STUDENT CATEGORY	ANNUAL CREDIT HOURS	BASIS FOR GOAL TARGET		OTHER CHARGES / FEES				
		SOURCE	MEASURE	ATHL	BUS	STUD CNTR	HLTH SERV	ROOM & BRD
RES_UG	30.	LG	MN	Y	Y	Y	Y	Y
NRES_UG	30.	LG	25	Y	Y	Y	Y	Y
RES_GR	27.	LG	MN	Y	Y	Y	Y	N
NRES_GR	27.	LG	25	Y	Y	Y	Y	N
RES_GR_OFC	03.	DI	DI	N	N	N	N	N
NRES_GR_OFC	03.	DI	DI	N	N	N	N	N

DEFINITIONS OF "BASIS FOR GOAL TARGET" OPTIONS:
SOURCE: LG = LAND-GRANT COLLEGES & UNIVERSITIES
NC = NCHEMS 1A ISTITUTIONS
VA = VIRGINIA 4-YEAR COLLEGES & UNIVERSITIES
DI = DIRECT INPUT
MEASURE: MN = ARITHMETIC MEAN
01 - 99 = PERCENTILE
DI = DIRECT INPUT
NOTE: IF DI, DIRECT INPUT, IS SPECIFIED FOR EITHER SOURCE OR MEASURE,
THEN IT MUST BE SPECIFIED FOR BOTH

Figure 4.14: The display for providing the peer institution information (file PEERGEN).

purposes) credit hours per student, e.g. 30 for undergraduates, 27 for graduate students, and the last two categories (off-campus) are based on the charges for a single three-credit-hour course.

The second and third data items required to estimate the peer institution goals are source and measure. Source refers to the data source on which the estimate is based, e.g., Land-Grant institutions, Virginia colleges and universities. Measure refers to the specific statistical measure of the data source used to estimate the goal value - arithmetic mean or percentile. The use of "DI", direct input, for the source and measure means that the user will directly specify the goal value. For example, in Figure 4.14, the first student category's peer institution goal is based on the mean (MN) charge at Land-Grant institutions (LG); the second is based on the 25th percentile (25) at Land-Grant institutions (LG); and, the last category's value will be directly supplied by the user (DI).

The final data item involves "other charges." The user specifies which of the "other charges", listed in Figure 4.14, are to be included in the peer institution goal target value calculations. As described earlier, information on charges at peer institutions is usually available in terms of the full-time charges for a particular student category,

including either tuition and fees or tuition, fees, room and board. A separate data file is maintained for each of these two types of charges for each student category. The DSS selects the appropriate file depending upon whether a Y or N is entered in the Room-and-Board columns of file the PEERGEN (Figure 4.14).

When all of the information is entered into PEERGEN, pressing the PF2 key stores the data and returns the process to the CCP. File PEERGEN is not only a data and information file, it can be printed through the CCP REPORTS and used as backup information. It is created by executing the FORTRAN program, PEERGEN.

CCP PEERINST uses the information contained in the file PEERGEN to create another data file, PEERINFO, for use by the FORTRAN programs PEERMCRO and PEERCALC. Only the essential data from the file PEERGEN is used in PEERINFO; none of the tutorial information is retained. The Y/N indicators for specifying which "other charges" to include in the analysis are converted to 1/0 notation. Also the "measure" column values are changed either from MN (mean) to -1 or from DI (direct input) to 0; the percentile values remain the same.

The file PEERGEN is also used to create a separate peer-institution working file for each student category.

The files are named PEERm where m=1,2,...,MM and MM is the number of student categories. An example of a PEERm file is given in Appendix E.11. CCP PEERINST copies the appropriate peer institution data file to the working file for each student category. For example, the first student category in Figure 4.14 is resident undergraduate and the peer institution goal is based on total charges at Land-Grant institutions. Therefore, the working file PEER1 is created as a copy of the data file RUONLGTO. If DI, direct input, is specified as the source and measure in the information file PEERGEN, then a copy of DIRECTIN is placed in the file PEERm. DIRECTIN contains a zero value for each percentile in each year.

CCP PEERINST interprets the name of the peer institution data file to be used for each student category. It builds the file name based on the information contained in PEERGEN and according to the naming convention described in Table 4.4. An error message is displayed if the CCP cannot locate a data file that corresponds to the information specified in PEERGEN. For example, assume that the user specified in PEERGEN that the peer institution goal for off-campus resident graduate students is to be based on total charges at Land-Grant institutions. The CCP searches for the file RGOFLGTO, but it cannot be found because the

Land-Grant data is not segregated by on-campus and off-campus locations, (Most institutions do not provide dorm-and-dining services at off-campus locations). The error message displays which file(s) cannot be found and indicates that the data can be handled as direct input or that it can be respecified. If it is to be respecified, PEERGEN is displayed on the terminal screen again and the user changes the data source designation.

Before the peer institution goals can be calculated, each of the working files (PEERm) must define the statistical measure to be used (mean, 90th percentile, etc.) and the name of the student category. This information is contained in IFPS MICRO statements. The MICRO statements are created by the FORTRAN program PEERMCRO. The number of student categories in the problem formulation is obtained from the file SETUP DSS and the statistical measure for each student category is obtained from the file PEERINFO. The MICRO statements are generated by the FORTRAN program and stored in the file PEERMCRO. Subsequently, CCP PEERINST reads the MICRO statements from PEERMCRO and places two statements (measure and name) in the appropriate working file.

The peer institution goals are calculated using IFPS. As shown in Figure 4.4, the IFPS models and report

generators obtain the basic peer institution data for each student category from the working files PEERm. The column statements are obtained from the file SETUP IFPS and the CPI, used to project the future charges, is obtained from the file COST. The IFPS model PEER is used to estimate the expected charge at peer institutions for each student category. It uses a simple linear regression model based on historical charges and the CPI from the six-year period 1978-1983. The model is described in detail in section 4.4.7 of this chapter. The historical and projected values are stored in IFPS format in the file PEEROUT. The projected values, for the years in the planning horizon, are stored in the file PEERCHG. Once the calculations have been completed, the file PEERCHG is displayed on the terminal screen and the user supplies the values that were designated DI, direct input. These have a value of zero in PEERCHG. The user directly enters the direct-input values on the screen and presses the PF2 key to store the data and return control to the ECP.

4.3.10 Component Control Program PEER

The peer institution goal target values are re-calculated if any of the following variables were changed during the present session: planning horizon, enrollment

data, financial data. The goal target calculation process is controlled by the CCP PEER.

There is no user interaction in setting the goal targets - CCP PEER executes the FORTRAN program PEERCALC which calculates the targets and converts the peer institution goals to MPS format. As illustrated in Figure 4.4, PEERCALC obtains input data from four files. The problem framework is defined by the file SETUP DSS. The file PEERINFO provides for each student category, the number of credit hours on which the goal is based and the indicators as to which "other charges" are included. The "other charges" values, for each student category in each year of the planning horizon, is obtained from the file DDFEEMPS. The expected charges at peer institution for each student category in each year is read from the file PEERCHG.

The goal target value, for each student category and for each year in the planning horizon, is the expected peer-institution tuition charge, net of any fees or room and board charges. It is calculated by subtracting the appropriate "other charges" (student fees, dorm and dining) from the peer institutions' expected total charge, estimated in the previously discussed CCP, PEERINST. A report summarizing the values used in the calculations is generated and stored in the file PEER (see Appendix E.10 for an example).

The peer institution goals are translated to MPS format by the FORTRAN program PEERCALC; the goals are stored in three files: PEERX contains the decision variables, PEERNP contains the deviation variables, and PEERRHS contains the target or right-hand-side values.

4.3.11 Component Control Program GENIND

The general individual goals are established in the CCP GENIND, as shown in Figure 4.4. The goal target values, stored in the file GENIND, are supplied directly by the user through the "display" option. An example of the input file is shown in Appendix E.9. Whenever the planning horizon is changed, an indicator is passed from the ECP that causes the input file GENIND to be regenerated. The FORTRAN program GENINGEN is called to regenerate the input file. The user then replaces the default values (zero) with the desired goal target values. GENIND is not only a data file; it can also be printed through the CCP REPORTS and used as a backup report.

The goals are converted to MPS format by the FORTRAN program GENIND. As illustrated in Figure 4.4, the program's input is obtained from two files. SETUP DSS defines the planning horizon and the number of goals included in the model formulation and the goal target values are obtained

from the file GENIND. The general individual goals are stored in MPS format in three data files: GINDX contains the decision variables, GINDNP contains the deviation variable, and GINDRHS contains the goal target values.

4.3.12 Component Control Program PRIORITY

Once all of the goals are established, they must be ranked in the order of importance to the university, thereby establishing the priority structure. Each goal included in the model formulation is assigned to a priority level. The priority structure is stored in the file PRIOR, an example of which is presented in Figure 4.15. As shown in Figure 4.4, control is passed from the ECP to CCP PRIORITY if the user chooses to review the priority structure or if the relative or absolute goals have changed this session.

The priority structure is re-generated whenever any of the relative or absolute goals are changed because the priority structure contains only the relative and absolute goals that are included in the model formulation. It follows that if any of these goals are added or deleted during the current session, the priority structure display file must be updated to reflect this change. The priority structure is re-generated by the CCP PRIORITY executing the FORTRAN program PRIOR. The program obtains the number of

PRIOR IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
GOAL PRIORITY STRUCTURE

IF THE GOAL TARGET IS THE MINIMUM ACCEPTABLE VALUE,
ENTER THE PRIORITY LEVEL IN THE 1ST COLUMN
IF THE GOAL TARGET IS THE MAXIMUM ACCEPTABLE VALUE,
ENTER THE PRIORITY LEVEL IN THE 2ND COLUMN

	PRIORITY LEVEL >=	PRIORITY LEVEL <=	SUBGOAL WEIGHTS	
STATE REVENUE REQUIREMENT	01	00	01.000	01.000
SUPPLEMENTAL REVENUE RQMT. - YEARLY	09	00	01.000	01.000
PERCENTAGE OF AVERAGE TOTAL COST GOALS				
RESIDENT_UNDERGRADUATE	00	04	01.000	04.000
NON-RESIDENT_UNDERGRADUATE	00	06	01.000	01.000
RESIDENT_GRADUATE	00	04	01.000	02.000
NON-RESIDENT_GRADUATE	00	04	01.000	01.000
RESIDENT_GRADUATE_OFF-CAMPUS	00	06	01.000	01.000
NON-RESIDENT_GRADUATE_OFF-CAMPU	00	06	01.000	01.000
CHANGE OVER TIME GOALS				
RESIDENT_UNDERGRADUATE	00	03	01.000	01.000
NON-RESIDENT_UNDERGRADUATE	00	03	01.000	01.000
RESIDENT_GRADUATE	00	03	01.000	01.000
NON-RESIDENT_GRADUATE	00	03	01.000	01.000
RESIDENT_GRADUATE_OFF-CAMPUS	00	03	01.000	01.000
NON-RESIDENT_GRADUATE_OFF-CAMPU	00	03	01.000	01.000
PEER INSTITUTIONS GOALS				
RESIDENT_UNDERGRADUATE	00	05	01.000	01.000
NON-RESIDENT_UNDERGRADUATE	00	05	01.000	01.000
RESIDENT_GRADUATE	00	07	01.000	01.000
NON-RESIDENT_GRADUATE	00	05	01.000	02.000
RESIDENT_GRADUATE_OFF-CAMPUS	00	07	01.000	01.000
NON-RESIDENT_GRADUATE_OFF-CAMPU	00	07	01.000	01.000
GENERAL INDIVIDUAL GOALS				
RESIDENT_UNDERGRADUATE	00	00	01.000	01.000
NON-RESIDENT_UNDERGRADUATE	00	00	01.000	01.000
RESIDENT_GRADUATE	00	00	01.000	01.000
NON-RESIDENT_GRADUATE	00	00	01.000	01.000
RESIDENT_GRADUATE_OFF-CAMPUS	00	00	01.000	01.000
NON-RESIDENT_GRADUATE_OFF-CAMPU	00	00	01.000	01.000
RELATIVE GOALS				
NRES_UG VS. RES_UG	08	00	01.000	01.000
RES_GR VS. RES_UG	08	00	01.000	01.000
SINGLE-CATEGORY ABSOLUTE GOALS				
PAIRWISE ABSOLUTE GOALS				
NRES_GR VS. RES_GR	02	00	01.000	01.000
NRES_GR VS. RES_GR	00	02	01.000	01.000
RES_GR_OFC VS. RES_GR	00	02	01.000	01.000
FOUR-CATEGORY ABSOLUTE GOALS				
NRES_GR - RES_GR - NRES_GR_OFC + RES_GR_OFC	02	02	01.000	01.000
SUPPLEMENTAL REVENUE RQMT. - OVERALL	09	00	01.000	01.000

Figure 4.15: The display for setting the priority structure (file PRIOR).

goals included in the model formulation and the student category names from the file SETUP DSS. The updated file, PRIOR, is displayed on the terminal screen. The user changes the default ranking for each goal (priority level zero) to the desired value.

As shown in Figure 4.15, the file PRIOR contains the priority level and subgoal weights for each goal included in the model formulation. Actually there are two columns in Figure 4.15 for specifying each goal's priority level. Specifying a non-zero value in the first column causes the negative deviation variables for that goal to be minimized in the achievement function. Another way of stating this is that the goal target value is assumed to be the minumum acceptable value. For example, the negative deviation variables for the state revenue goal in Figure 4.15 will be minimized in the first (or highest) priority level, meaning that the user wants to ensure that the tuition charge structure is established in such a way that total revenue is at least as large as the state requirement. Conversely if a non-zero value is entered in the second priority level column, the goal target value is considered as the maximum acceptable value. This causes the positive deviation variables to be minimized in the achievement function at the appropriate priority level. If both columns are entered as

non-zero values, the goal is treated as an equality and the solution procedure attempts to attain the target value exactly.

The subgoal weights, as defined in the previous chapter (section 3.4.1), are used to give goals within the same priority level different emphasis. Since the value of the subgoal weight is the coefficient of the corresponding deviation variable in the achievement function, the default value provided in the file PRIOR is 1.

Before continuing the description of the activities performed by CCP PRIORITY, it is necessary to present three general comments associated with establishing the priority structure. First, each goal is assumed to have the same rank in each year of the planning horizon. Therefore, there is no reference to time in the file PRIOR and the priority structure does not have to be re-entered if the planning horizon is changed. Second, a goal is excluded from consideration in the solution routine if either both priority levels or both subgoal weights are entered as zero. If the user wants to exclude a goal from consideration, it is recommended that both priority level values be set to zero. Third, the priority levels must be entered sequentially; e.g., if the highest priority-level value specified in PRIOR is 8, then priority levels 1,2,...,7 must also be specified.

Once the priority structure is established, CCP PRIORITY creates the file PRIOROPT for use by the optimization routine. The priority level and subgoal weights are "stripped" from the file PRIOR and are stored in a more convienent format in the file PRIOROPT.

Since the optimization routine will fail if the priority levels are not entered sequentially, a check is made to see if this has occurred before the optimization routine is called. As shown in Figure 4.4, this check is made by the FORTRAN program PRIORCHK. If the priority level values are not sequential, an error message is generated informing the user of this fact and the priority structure is re-displayed for correction. If the priority levels are sequential, control is returned to the ECP.

4.3.13 Component Control Program MPS

Three CCPs are involved in the solution of the goal programming model; the first one accessed by the ECP is MPS. MPS establishes the MPS format file that is used by LINDO to solve the goal programming model. There is one condition when the ECP does not access CCP MPS; this occurs when the user bypasses the data and goal-setting sections and proceeds directly to the priority structure. In this case the goals in the model are not changed, only the ranking of

the goals are changed; therefore, the ECP accesses only one of the solution CCPs, GP.

All of the files that contain the goals in MPS format are combined into a single file, as shown in Figure 4.4. The files containing the deviation variables (AGRGNP, COMPNP, CTIMENP, GINDNP, PCOSTNP, and PEERNP) are processed first. They are combined into one file DSSNP and then sorted first by deviation variable number and second by whether the variable is a negative deviation or a positive deviation. The deviation-variable statements are stored in the file DSSNPS after sorting; the unsorted file DSSNP is erased.

The decision variables are processed in a similar manner. They are sorted first by year, second by student category, and third by row number (goal), thereby grouping like decision variables together. This is necessary because a requirement of LINDO (and most commercial LP systems) is that all elements for a given variable appear together.

A portion of the file that contains the decision-variable statements in the proper order, DSSXS, is presented in Figure 4.16. The first column contains the variable's name, the second contains the row name; and the third column is the variable's coeffecient in the just specified row. Notice that there are eight statements for decision variable

DSSXS DSS

X11	ROW10002	438.663
X11	ROW10008	0.895
X11	ROW10026	1.000
X11	ROW10032	-1.210
X11	ROW10044	30.000
X11	ROW10062	1.000
X11	ROW10080	-2.000
X11	ROW10083	-1.200
X21	ROW10002	124.875
X21	ROW10009	0.895
X21	ROW10027	1.000
X21	ROW10033	-1.250
X21	ROW10045	30.000
X21	ROW10063	1.000
X21	ROW10080	1.000
X31	ROW10002	76.453
X31	ROW10010	0.895
X31	ROW10028	1.000
X31	ROW10034	-1.230
X31	ROW10046	27.000
X31	ROW10064	1.000
X31	ROW10083	1.000
X31	ROW10086	-1.000
X31	ROW10089	-1.000
X31	ROW10092	-1.000
X31	ROW10095	-1.000
X41	ROW10002	13.930
X41	ROW10011	0.895
X41	ROW10029	1.000
X41	ROW10035	-1.230
X41	ROW10047	27.000
X41	ROW10065	1.000
X41	ROW10086	1.000
X41	ROW10089	1.000
X41	ROW10095	1.000
X51	ROW10002	16.484
X51	ROW10012	0.895
X51	ROW10030	1.000
X51	ROW10036	-1.250
X51	ROW10048	3.000
X51	ROW10066	1.000
X51	ROW10092	1.000
X51	ROW10095	1.000
X61	ROW10002	0.538
X61	ROW10013	0.895
X61	ROW10031	1.000
X61	ROW10037	-1.250
X61	ROW10049	3.000
X61	ROW10067	1.000
X61	ROW10095	-1.000
X12	ROW10003	434.943
X12	ROW10014	0.785
X12	ROW10032	1.000
X12	ROW10038	-1.100
X12	ROW10050	30.000
X12	ROW10068	1.000
X12	ROW10081	-2.000
X12	ROW10084	-1.200
X22	ROW10003	123.627
X22	ROW10015	0.785
X22	ROW10033	1.000
X22	ROW10039	-1.100
X22	ROW10051	30.000
X22	ROW10069	1.000
X22	ROW10081	1.000

Figure 4.16: The decision variables in MPS format (file DSSXS).

X11, the tuition charge for resident undergraduate students in the first year of the planning horizon. The first statement corresponds to the state revenue requirement goal, the second to the percentage-of-average-total-cost goal, and the third and fourth to the change-in-tuition-over-time goal, etc.

The right-hand-side or goal target values are combined in a similar manner. They are sorted by row number and stored in the file DSSRHSS.

The MPS file used in the optimization, FILE FT08F001, consists of three components. An example of the MPS file is provided in Appendix E.8. The ROWS section, created in the CCP SETUP, gives the name and the equality or inequality type of each row. The second component, COLUMNS, lists each variable (decision and deviation), the corresponding row number where it appears in the model, and its coefficient. The COLUMNS section is created by adding the files DSSNPS and DSSXS to FILE FT08F001. The third component, RHS, lists the right-hand-side value (goal target value) for each row in the model. The RHS section is added to FILE FT08F001 by appending the file DSSRHSS. An ENDATA statement is added as the last line in the MPS file.

4.3.14 Component Control Program FORMAT

Prior to solving the model, the user has the option of obtaining a listing of the entire model formulation in equation form. In order to obtain the formulation, control is passed to the CCP FORMAT, as shown in Figure 4.5. The CCP accesses the LINDO system and provides the LINDO commands necessary to list the model. The model formulation is displayed on the terminal screen and is stored in the file FORMAT for hardcopy printing, if desired. A portion of the GP model in equation form, as it is stored in the file FORMAT, is displayed in Figure 4.17.

4.3.15 Component Control Program GP

The model is solved using the optimization procedure defined in the CCP GP. Once control is passed from the ECP, CCP GP accesses the LINDO system and executes a series of LINDO commands. The solution process is controlled by a special FORTRAN subroutine that is interfaced with the LINDO system. The process, as illustrated in Figure 4.5, was discussed in detail in the previous chapter (sections 3.5.2 and 3.5.3). Control is returned to the ECP when the model is solved and the file TUIT is automatically displayed on the terminal screen. It contains the value of the decision variables - the tuition charge per credit hour for each student category in each year, including the base year.

FORMAT IFPS

```
MIN
SUBJECT TO
   2)      N001 -  P001 + 438.66309 X11 + 124.875 X21 + 76.453 X31
     + 13.93 X41 + 16.48399 X51 + 0.538 X61 =     36230.30078
   3)      N002 -  P002 + 434.94312 X12 + 123.627 X22 + 81.799 X32
     + 14.798 X42 + 16.65401 X52 + 0.543 X62 =     43559.00391
   4)      N003 -  P003 + 432.7959 X13 + 122.638 X23 + 85.216 X33
     + 15.441 X43 + 16.929 X53 + 0.55 X63 =     46762.99219
   5)      P001 +  N004 -  P004 =     400
   6)      P002 +  N005 -  P005 =     500
   7)      P003 +  N006 -  P006 =     600
   8)      N007 -  P007 + 0.615 X11 =     25
   9)      N008 -  P008 + 0.615 X21 =     50
  10)      N009 -  P009 + 0.554 X31 =     30
  11)      N010 -  P010 + 0.554 X41 =     35
  12)      N011 -  P011 + 0.554 X51 =     40
  13)      N012 -  P012 + 0.554 X61 =     45
  14)      N013 -  P013 + 0.54 X12 =     28
  15)      N014 -  P014 + 0.54 X22 =     60
  16)      N015 -  P015 + 0.486 X32 =     33
  17)      N016 -  P016 + 0.486 X42 =     48
  18)      N017 -  P017 + 0.486 X52 =     40
  19)      N018 -  P018 + 0.486 X62 =     45
  20)      N019 -  P019 + 0.505 X13 =     30
  21)      N020 -  P020 + 0.505 X23 =     65
  22)      N021 -  P021 + 0.455 X33 =     35
  23)      N022 -  P022 + 0.455 X43 =     40
  24)      N023 -  P023 + 0.455 X53 =     40
  25)      N024 -  P024 + 0.455 X63 =     45
  26)      N025 -  P025 +  X11 =     42.12601
  27)      N026 -  P026 +  X21 =     92.64
  28)      N027 -  P027 +  X31 =     52.97
  29)      N028 -  P028 +  X41 =     58.87
  30)      N029 -  P029 +  X51 =     65.88
  31)      N030 -  P030 +  X61 =     71.98
  32)      N031 -  P031 - 1.21 X11 +  X12 =     0
  33)      N032 -  P032 - 1.25 X21 +  X22 =     0
  34)      N033 -  P033 - 1.23 X31 +  X32 =     0
  35)      N034 -  P034 - 1.23 X41 +  X42 =     0
                    .
                    .
                    .
  91)      N090 -  P090 -  X33 +  X43 =     20
  92)      N091 -  P091 -  X31 +  X51 =     10
  93)      N092 -  P092 -  X32 +  X52 =     15
  94)      N093 -  P093 -  X33 +  X53 =     20
  95)      N094 -  P094 -  X31 +  X41 +  X51 -  X61 =     0
  96)      N095 -  P095 -  X32 +  X42 +  X52 -  X62 =     0
  97)      N096 -  P096 -  X33 +  X43 +  X53 -  X63 =     0
  98)      P001 +  P002 +  P003 +  N097 -  P097 =     2000
```

Figure 4.17: A portion of the GP model in equation form (file FORMAT).

Another one of the four reports generated through CCP GP (stored in the file DEVVAR) contains information on how well the solution achieved the specified goals. As shown in Figure 4.5, the option to view this report is controlled by the ECP.

4.3.16 Component Control Program REPORTS

The final CCP is used to select which reports to print. The user is first required to provide the location (box number) where the output is to be sent. Subsequently the menu of available reports is displayed on the terminal screen; the list of reports is stored in the file REPORTS. Figure 4.18 provides an example of this display.

Within the file REPORTS there are three pieces of information stored on each available report. The first is a single-character indicator for how the report is to be printed. The second field contains the file name where the report is stored; the third contains a brief description of the report. Three options are available for printing the reports, controlled by the single-character indicator - R, L, or N - in the first column of each report statement. R causes the report to be generated on standard computer paper, L generates the report on letter-size and letter-quality paper (from the IBM 6670 Laser printer), and N is used if the report is not to be printed at all.

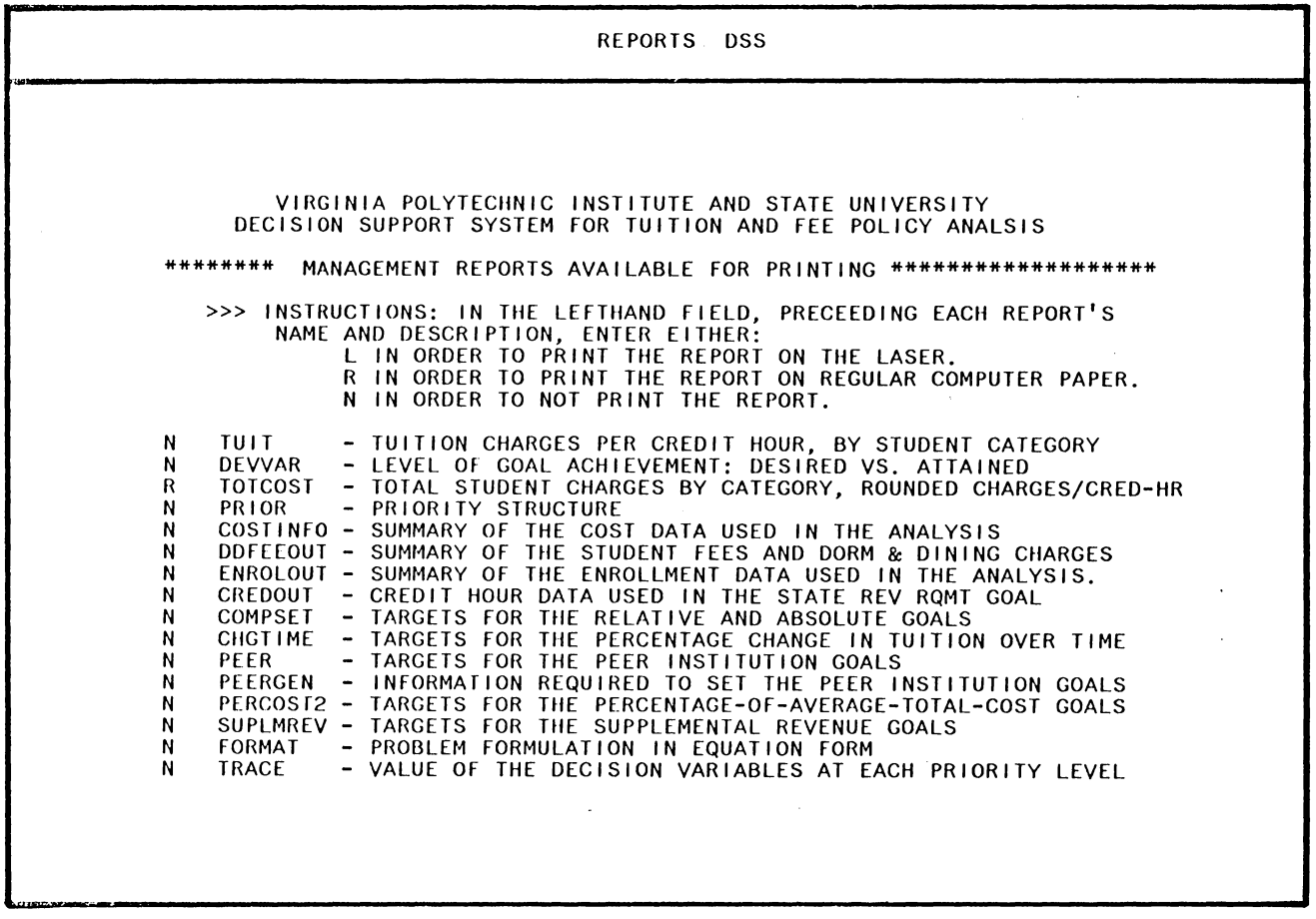


Figure 4.18: The display for selecting which reports to print (file REPORTS).

All of the reports listed in the file REPORTS have already been generated except for the third report, TOTCOST - the total charges by student category. It is an IFPS generated report, that is based on the tuition charges that were just calculated in the CCP GP. As shown in Figure 4.5, if this report is selected for printing, it is generated by a call to the IFPS system. Five IFPS models are used to perform the calculations and the resulting reports are stored in the file TOTCOST.

After the reports have been submitted for printing, control is returned to the ECP where the user can either perform another analysis or end the current session.

4.4 THE MODELS SUBSYSTEM

The analytic portion of the DSS is provided in the models subsystem. Most of the models are used to calculate parameter and goal target values. One interactive optimization routine is used to solve the goal programming model. By imbedding the models within the DSS, input values are automatically supplied and reports that document the solution are automatically generated.

The models subsystem is a collection of separate models, each dealing with a distinct problem. This separation not only facilitates model development, testing,

modification, and extension but also creates a demand for an interface between the models themselves as well as with the dialog and data subsystems. The models subsystem is integrated with the dialog subsystem to provide control over the models' operation and use; it is integrated with the data subsystem to receive input data and to generate output information.

There are two functions of the model subsystem in the DSS. The first function, and the focus of this section, is the calculation of the GP model's parameter values - the decision variable coefficients and the goal target values. The second function, the solution of the GP model once it is formulated, was discussed in detail in Chapter 3 (section 3.5).

All of the model's parameter values that are not directly specified are calculated using the IFPS system. When certain calculations are required, the DSS accesses IFPS and passes control to an IFPS command file. Each command file contains a sequence of statements that define which IFPS models and data files to use and which reports to generate. The models are the core of IFPS and are therefore given the most attention in this section. Program listings for the seven models used in the DSS are provided in Appendix D.3. The program listings for the sixteen report

generators that are used to format the output are provided in Appendix D.4.

The notation used in this section is similar to the notation used in Chapter 3 to define the decision variables. The subscripts have the same meaning as they did in Chapter 3; but, two superscripts have been added to some variables and the the fee categories and dorm groups are indexed. For example, $CHG_k(f)$ is the charge for fee category f, in year k. The notation used in this section is defined below:

subscripts

h = 1,2 (academic level: undergraduate, graduate)

i = 1,2 (residency status: resident, non-resident)

j = 1,2 (location: on-campus, off-campus)

k = 1,...,KK (academic year)

superscripts

s = 1,2 (session: regular, summer)

t = 1,2 (participation status: full-time and part-time)

index

f = 1,2, ... ,5 (other student charges: athletic fee, bus fee, student center fee, health service fee, dorm-and-dining charge)

r = 1,...,RR (dorm group: for undergraduate (h=1), RR=5 - upper quad, lower quad, prairie quad, new undergraduate, and special purpose; for graduate (h=2), RR=2 - single or double occupancy.

4.4.1 The linkage between the DSS and IFPS

When the IFPS system is used to perform the calculations in the DSS it is accessed directly from a component control program (CCP), as shown in Figure 4.19. Two programs, written in EXEC 2 command language, provide the linkage between the DSS and IFPS. The program IFPSDSS passes control from the CCP to IFPS and the program QUIT returns control to the CCP once the calculations have been made. The calculation process within the IFPS system is controlled by an IFPS command file.

Normally IFPS is accessed through CMS by entering "IFPS", which executes Execucom's EXEC, IFPS. The user is then prompted for the model-and-reports file name; once the file has been specified, the user provides an executive-level command, e.g. build/execute a model, execute a command file. These steps are automatically performed by the DSS.

When a CCP is ready for IFPS to perform certain calculations, it executes the EXEC file, IFPSDSS, using the command below:

```
EXEC IFPSDSS A B C D E F G H
```

IFPSDSS is a modified version of Execucom's file IFPS EXEC that accesses the IFPS system and passes the necessary commands and parameters to the system. As shown in the command above, up to eight parameters are passed from the

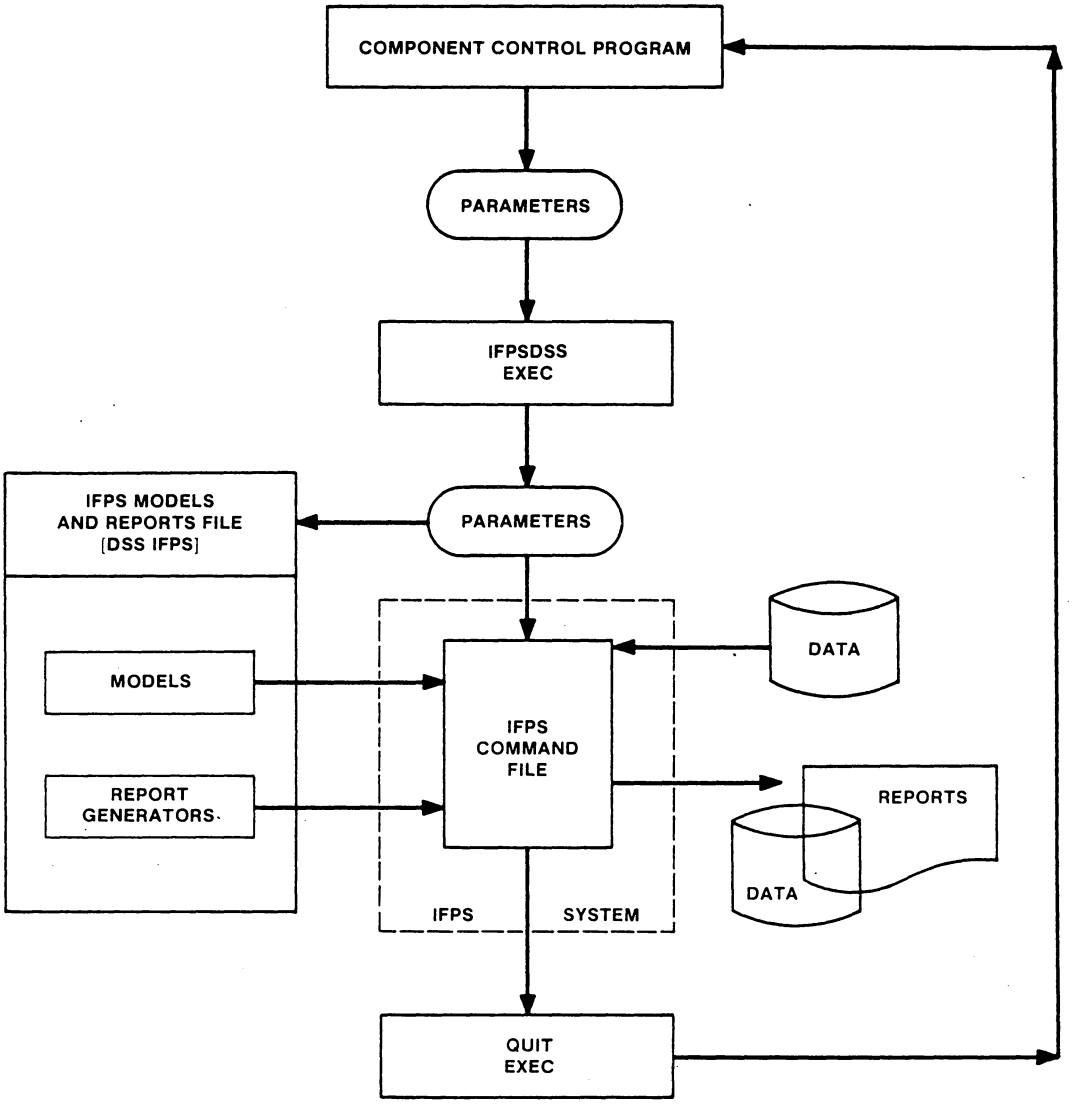


Figure 4.19: Calculations are performed in the DSS by accessing IFPS directly from a CCP.

CCP to IFPSDSS. The first three parameters, A-C, are dummy values, D is the name of the model-and-reports file name (DSS), E is the name of the command file that controls the IFPS operations, and F-H are parameters used by the IFPS command file, e.g. name of the enrollment files.

4.4.2 IFPS Command files

The calculations performed by IFPS are controlled through five command files. Each command file contains a set of statements that cause certain IFPS operations to be performed in a prescribed sequence - they designate which data files to use, which models to solve, and where to store the results.

Figures 4.2 through 4.5 show where each of the command files are used in the DSS. The figures also identify which of the IFPS models and report generators are used, the source of the input data, and what output data files and reports are generated. CCP AGGR in Figure 4.3 accesses three of the command files - ENRLCRED, COSTCOM, and DDFEECOM; CCP PEERINST calls the command file PEERCOM and CCP REPORTS calls TUITCOM. The program listings for the command files are provided in Appendix D.2.

4.4.3 IFPS model ENROLL

All of the enrollment-related variables are calculated by the IFPS model ENROLL. Data are supplied from the appropriate "actual" and "projected" data files and from the file FACTOR. The model has two functions: first, calculate the coefficients of the decision variables for the state revenue requirement goals and second, calculate the value of enrollment variables that are used by other IFPS models and that provide the basis for several management reports.

Recall that the enrollment values are specified in each year by: (1) session (regular, summer), (2) participation status (full-time, part-time), (3) academic level (undergraduate, graduate), (4) residency status (resident, non-resident), and (5) location (on-campus, off-campus). Many of the variables that are used in the equations described below can be expressed at the same level of detail as enrollment; but, many have been simplified, mainly to facilitate the data input process. For example, the retention-factor variable, used to account for the change in enrollment between academic terms, can be specified at the same level of detail as enrollment; but, in this model, it is assumed to vary only by session and by location and not by participation status, academic level, or residency status.

4.4.3.1 Calculation of the decision-variable coefficients for the state revenue requirement goals.

The first function of the model is to calculate the decision-variable coefficients for the state revenue requirement goal that was formulated in Equation (3.1). Each coefficient CH_{hijk}, as expressed in Equation (4.1) below, is the expected number of annual credit hours per student-category. The coefficients are based on the "actual" enrollment data and on the number of credit hours required to be considered full-time for tuition-charging purposes. Recall that the model is based on a "per-credit-to-plateau" charge function. This means that beyond some credit-hour value, students pay the same total charge, regardless of the actual number of hours in which they are enrolled.

$$CH_{hijk} = \sum_{s=1}^{2} R_j^s \sum_{t=1}^{2} [EA_{hijk}^{s,t} * CH_{hj}^{s,t} + TWCH_{hijk}^{s,t}] \quad (4.1)$$

where:

R_j^s = retention factor to account for different enrollment levels between academic terms. For example, if it is expected that 96.6% of Fall quarter's on-campus students return in the Winter term, and 93.8% of Fall's students return in the Spring, then $R_1^1 = 1 + .966 + .938 = 2.904$. For summer session, $R_j^2 = 1$.

$EA^{s,t}_{hijk}$ = "actual" headcount enrollment, as input from the file AENROLL_ or CAENROLL.

$CH^{s,t}_{hj}$ = for t=1: number of credit hours per term that a student must carry in order to be considered full-time for tuition-charge purposes.

for t=2: average number of credit hours per term for students not paying full-time charges. The same value is used for resident and non-resident students and from year-to-year.

$TWCH^{s,t}_{hijk}$ = adjustment in credit hours to account for students that receive tuition waivers. Usually some percentage of the non-resident graduate students have their out-of-state tuition charge waived. Therefore, they are considered in-state students for revenue purposes.

Because of the tuition waivers, the number of resident graduate students (h=2,i=1) must be increased for revenue purposes, therefore:

$$TWCH^{s,t}_{21jk} = TW^{s,t}_{22jk} * EA^{s,t}_{22jk} * CH^{s,t}_{22jk}$$

Conversly, the number of non-resident graduate students (h=2,i=2) must be decreased for revenue purposes, therefore:

$$TWCH^{s,t}_{22jk} = -TWCH^{s,t}_{21jk}$$

Since it is assumed that undergraduates do not receive tuition waivers:

$TWCH^{s,t}_{lijk}$ = 0, otherwise

$TW^{s,t}_{hijk}$ = fraction of the students receiving tuition waivers. This term is non-zero only when h=2 and i=2, i.e. it only pertains to non-resident graduate students.

The IFPS report generator CREDMPS transfers the decision-variable coefficients for the state revenue requirement goals to a FORTRAN-readable format and stores them in the file STATEREV. A management report that summarizes these values is created by the IFPS report generator CREDIT and is stored in the file CREDOUT. An example of the report is presented in Figure 4.20.

4.4.3.2 Calculation of enrollment values

The number of "full-time equivalent" (FTE) students is calculated in the IFPS enrollment model. It is used in the tuition and fee recovery rate calculations performed by the IFPS model COST. As shown in Equation (4.2) below, it is based on the "projected" data and considers only Fall-quarter enrollment.

$$FTE^{s}_{hijk} = ACH^{s}_{hj}\left[\sum_{t=1}^{2} EP^{s,t}_{hijk}\right]/FTECH_{h} \qquad (4.2)$$

where:

CREDOUT IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
EXPECTED CREDIT HOURS BY STUDENT CATEGORY 05/22/84

	1983	1984	1985
REGULAR AND SUMMER SESSIONS			
RESIDENT UNDERGRADUATE	438,663	434,943	432,796
NON-RESIDENT UNDERGRADUATE	124,875	123,627	122,638
RESIDENT GRADUATE	76,453	81,799	85,216
NON-RESIDENT GRADUATE	13,930	14,798	15,441
OFF-CAMPUS			
RESIDENT GRADUATE	16,484	16,654	16,929
NON-RESIDENT GRADUATE	538	543	550
TOTAL EXPECTED CREDIT HOURS	670,941	672,364	673,570

Figure 4.20: Summary of the credit hour data used in the state revenue requirement goal (file CREDOUT).

ACH^{s}_{hj} = overall average number of credit hours carried per student per academic term. In this model, it is assumed to vary only by session, academic level, and campus location.

$EP^{s,t}_{hijk}$ = "projected" headcount enrollment, as input from the file PENROLL_ or CPENROLL.

$FTECH_h$= number of credit hours carried per student per academic term to be considered a "full-time equivalent". $FTECH_1$= 15, $FTECH_2$= 12.

The number of annual FTE students, defined below in Equation (4.3), is used in the IFPS model COST to calculate the average cost per student. It is the variable defined by Equation (4.2) converted to an annual basis.

$$AFTE_{hijk} = \sum_{s=1}^{2} R^{s}_{j} * FTE^{s}_{hijk} \qquad (4.3)$$

Model ENROLL calculates a series of summary values for each set of enrollment data, Fall-quarter headcount, FTE, etc. The summary values, for example total on-campus students or total graduate students, are used in other IFPS models and reports. The management report that is generated directly from the model by the IFPS report generator ENROLL

summarizes the enrollment data used by the DSS. It is stored in the file ENROLOUT, an example of which is included as Appendix E.5. The report displays by student category and by year the "actual" and "projected" enrollment data used in the DSS both in terms of headcount and FTE.

4.4.4 IFPS model FEES

The quarterly charges per student for each of the fee categories - athletic, bus, student center, and health service - are calculated by the IFPS model FEES. The charge for each fee category includes two components; $BF_k(f)$, a "basic" charge to cover operating expenses and existing debt service costs and $DS_k(f)$, a "new debt service" charge for new capital projects. The calculation process is flow charted in Figure 4.21.

As shown in the top portion of Figure 4.21, the file FEEBASE provides the data for calculating the basic-charge component. For each fee category in each year, these inputs include: the quarterly charge, annual operating expenses (in thousands of dollars), an economic escalator, and the proportion of part-time students that are expected to pay the particular fee. The user supplies the fee data through the CCP FINANCE.

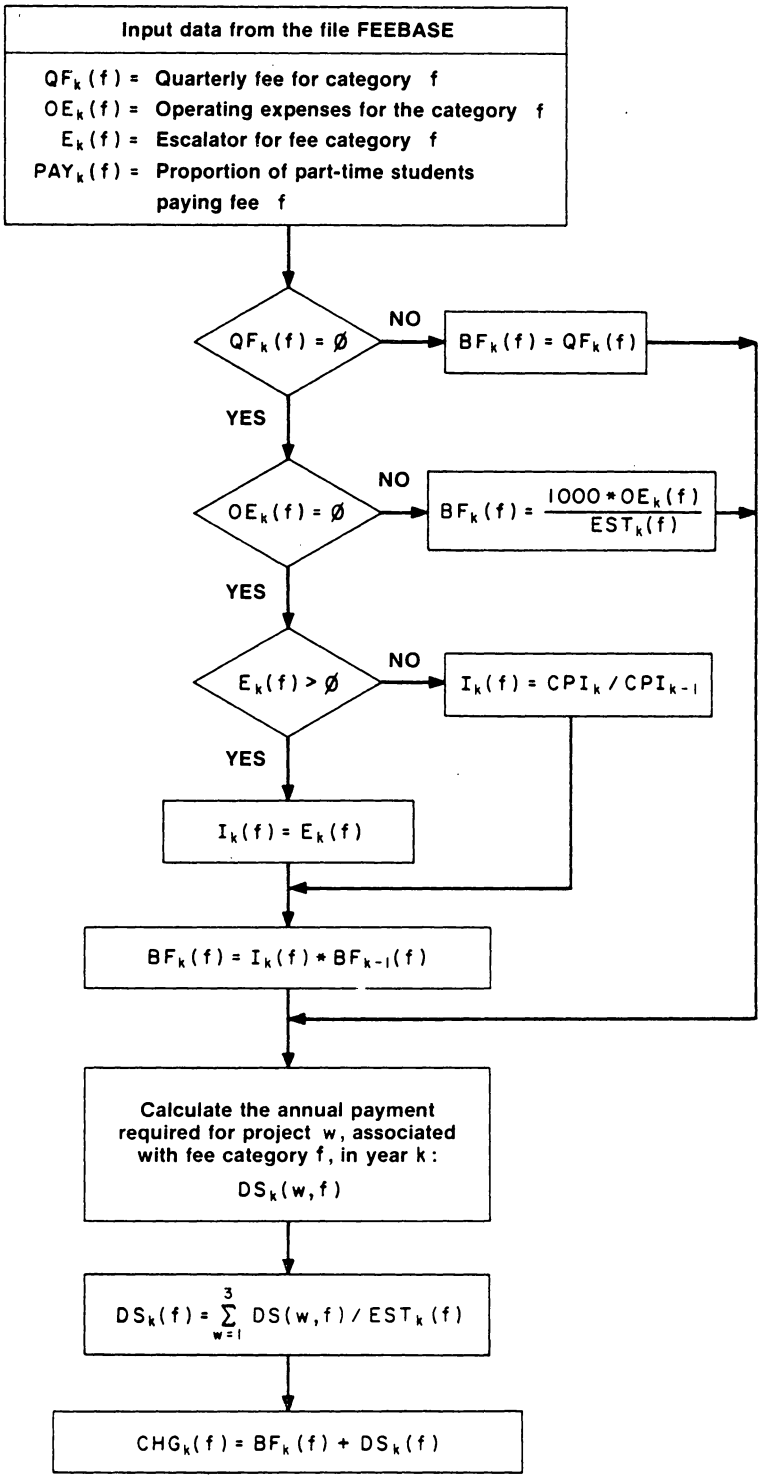


Figure 4.21: Process for calculating the quarterly charge for the various fee categories.

There is a hierarchy of options available for calculating student fees. Referring to the first decision block in Figure 4.21, if the quarterly charge is directly specified, then it is used as the basic fee value. If it is not specified (entered as zero), a test is made to see if the operating expenses have been specified. If they have been, then the basic fee is calculated by dividing the annual operating expenses by the number of students expected to pay the particular fee. The number of students, equivalent annual full-time students for fee purposes, is defined as:

$$EST_k(f) = \sum_{h=1}^{2}\sum_{i=1}^{2}\sum_{j=1}^{2} R_j^s \sum_{s=1}^{2} [EA_{hijk}^{s,1} + PAY_{jk}(f)*EA_{hijk}^{s,2}] \quad (4.4)$$

$PAY_{jk}(f)$ = fraction of the part-time students that pay fee f. It is assumed that all students are assessed the same fee charge, except off-campus; off-campus students do not pay fees. Therefore, $PAY_{2k}(f) = 0$, for all f and k.

If the both the quarterly charge and the operating expenses have not been directly specified, the model calculates the basic-charge component based on the previous year's value.

As shown in Figure 4.21, either the escalator is used to project the basic charge or the change in the CPI is used, if the escalator is entered as a negative value.

The second component of the student fee charge is for debt service on new capital projects. The information for each capital project that is associated with each fee category is entered in the file CAPGEN and translated to an IFPS-readable file, CAPIFPS, by the CCP FINANCE. Up to three projects can be specified for each fee category. Each of the projects' loan amount is amortized using the IFPS subroutine AMORT. The subroutine determines the annual payment required to repay the loan in equal installments, including payment of accrued interest and payment against the principal. The amount required in year k for the w^{th} project associated with fee category f, $DS_k(w,f)$, is calculated using:

$$DS_k(w,f) = \frac{A*r*(1+r)^n}{(1+r)^n - 1} \qquad (4.5)$$

where, for each project w: A is the loan amount, r is the effective annual interest rate, and n is the life of the loan, in years. The quarterly debt service charge that is assessed to each student is determined by dividing the toal annual amortized loan amount for all of the projects

associated with a certain fee category by the number of equivalent annual students defined by Equation (4.4).

As shown in the last block of Figure 4.21, the total quarterly charge for each fee category is the sum of the basic charge and the new-debt-service charge.

The IFPS report generator FEES creates a management report, as shown in Figure 4.22, that summarizes the student fee charges. It displays, by year, the basic charge and the new-debt-service charge for each fee category. The report is stored in the file DDFEEOUT.

4.4.5 IFPS models DD1 and DD2

Recall that there are two options available for calculating the dorm-and-dining charges - either directly input the charges for each dorm group (Option 1) or input the overall operating expenses and have the charges for each group allocated based on a set of user-specified ratios (Option 2). The user specifies the option and enters the data through CCP FINANCE. IFPS model DD1 and data file DDBASE1 are used when Option 1 is selected and DD2 and DDBASE2 are used when Option 2 is selected. Since the two models are similar, they are both discussed in the same section.

DDFEEOUT IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

STUDENT FEE DATA

	1983	1984	1985
ANNUAL EQUIVALENT STUDENTS			
ATHLETIC FEE	62,877	63,024	63,132
BUS FEE	62,093	62,228	62,328
STUDENT CENTER	66,011	66,209	66,351
HEALTH SERVICE FEE	62,093	62,228	62,328
ANNUAL STUDENT FEES			
ATHLETIC ASSOCIATION			
BASE FEE	72	78	84
DEBT SERVICE	-	-	-
TOTAL FEE	72	78	84
BUS SYSTEM			
BASE FEE	12	15	15
DEBT SERVICE	-	-	-
TOTAL FEE	12	15	15
STUDENT CENTER			
BASE FEE	63	69	72
DEBT SERVICE	-	63	63
TOTAL FEE	63	132	138
HEALTH SERVICES			
BASE FEE	69	72	78
DEBT SERVICE	-	-	6
TOTAL FEE	69	72	84
TOTAL ANNUAL FEES	$216	$297	$321

Figure 4.22: Summary of the student fee charges generated by the DSS (file DDFEEOUT).

The dorm-and-dining charge, like the fee charges, includes two components: a basic charge to cover operating expenses and existing debt service costs and a new-debt-service charge for proposed capital projects. The information for each capital project is entered in the file CAPGEN and translated to an IFPS-readable file, CAPIFPS, by the CCP FINANCE. Each of the projects' loan amount is amortized using the IFPS subroutine AMORT. The annual repayment amount for each project is calculated by using Equation (4.5), the same as in the previous section. The quarterly new-debt-service charge is the sum of the annual amortized loan amount for all of the proposed dorm-and-dining capital projects divided by the number of students residing in the dormitories.

The basic dorm-and-dining charge component, BDD_{hk}, is the average charge for an undergraduate (h=1) or a graduate student (h=2) in year k. It is the sum of the average basic dorm charge, BR_{hk}, and the basic dining charge, BB_{hk}.

As discussed earlier, the dorm charges vary by dorm group. BR_{hk} is the weighted average of those charges, expressed as:

$$BR_{hk} = \left[\sum_{r=1}^{RR} RCHG_{hk}(r) * RST_{hk}(r) \right] / \sum_{r=1}^{RR} RST_{hk}(r) \qquad (4.6)$$

$RST_{hk}(r)$ is the number of students residing in dorm r group; it is supplied by the user in either data file DDBASE1 or DDBASE2. $RCHG_{hk}$ is the quarterly charge for dorm group r.

The charge for each dorm type, $RCHG_{hk}(r)$, is either directly supplied by the user (Option 1) or calculated based on operating expenses (Option 2). If Option 2 is used, then the annual operating expenses for the dormitory system, OE_k, and a charge index $I_{hk}(r)$ for each dorm group are supplied by the user in the file DDBASE2.

The index $I_{hk}(r)$ specifies the desired relative charge difference between dorm group r and some baseline group. For example, assume that the prairie-quad dorm group is considered to be the baseline; then $I_{1k}(3)=1.00$. If the administration wants to charge the students residing in the new undergraduate dorm group 40% more than those living in the prairie-quad group, then $I_{1k}(4)=1.40$; likewise, if the administration wants the upper-quad group to be priced 15% below prairie quad, then $I_{1k}(1)=0.85$.

Equation (4.7) is used in the IFPS model DD2 to establish the quarterly charge for each dorm group, based on its charge index and on the system's expected operating expenses.

$$RCHG_{hk}(r) = \frac{I_{hk}(r)*OE_k}{3*\sum_{h=1}^{2}\sum_{r=1}^{RR} RST_{hk}(r)*I_{hk}(r)} \qquad (4.7)$$

The basic dining charges, BB_{hk}, are handled the same as the basic dorm charges, except there are not as many different charge types. The model only considers two meal plans, differentiated by academic level. This distinction is made because the graduate meal plan is based on a calendar quarter; whereas the undergraduate plan is based on an academic quarter. The quarterly charge for both plans is either directly specified in the file DDBASE1, under Option 1, or calculated based on expected operating expenses, under Option 2.

If the quarterly charges have a value of zero in the data file DDBASE1 or if the operating expenses have a value of zero in the data file DDBASE2, then the model calculates the quarterly charge based on the previous year's value. Either the escalator that is specified in the data file is used to project the charge or the change in the CPI is used, if the escalator is entered as a negative value.

The IFPS report generator DORMDING creates a management report that summarizes the dorm-and-dining charges. As shown in Figure 4.23, the report displays for each year in the planning horizon, the number of students and the basic charge by dorm group, the new-debt-service charge by project, and the total annual charges. It is stored with the student fees report in the file DDFEEOUT.

DDFEEOUT IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
DORM AND DINING DATA 04/12/84

	1983	1984	1985
UNDERGRADUATE STUDENTS			
UPPER QUAD	1,950	1,950	1,950
LOWER QUAD	4,380	4,380	4,380
PRAIRIE QUAD	1,969	1,969	1,969
NEW UNDERGRADUATE DORM	342	342	342
SPECIAL PURPOSE HOUSING	108	108	108
TOTAL UNDERGRADUATE STUDENTS	8,749	8,749	8,749
GRADUATE STUDENTS			
GRADUATE SINGLE OCC	36	36	36
GRADUATE DOUBLE OCC	120	120	120
TOTAL GRADUATE STUDENTS	156	156	156
TOTAL ON CAMPUS	8,905	8,905	8,905
BASE DORMITORY CHARGES (QUARTERLY)			
UPPER QUAD	$141	$151	$158
LOWER QUAD	159	170	179
PRAIRIE QUAD	159	170	179
NEW UNDERGRADUATE DORM	220	236	248
SPECIAL PURPOSE HOUSING	223	238	250
UNDERGRADUATE AVERAGE	$158	$169	$178
GRADUATE SINGLE OCC	346	370	389
GRADUATE DOUBLE OCC	219	234	246
GRADUATE AVERAGE	$248	$266	$279
BASE DINING CHARGES (QUARTERLY)			
UNDERGRADUATE	$345	$369	$388
GRADUATE	397	424	446
NEW DEBT SERVICE (QUARTERLY)			
PROJECT 1	$10	$10	$10
PROJECT 2	-	20	20
PROJECT 3	-	-	-
TOTAL DEBT SERVICE	$10	$29	$29

Figure 4.23: Summary of the dorm-and-dining charges generated by the DSS (file DDFEEOUT).

The report generator DDFEEMPS is used to create a data file (DDFEEMPS) that contains the student fee charges and the dorm-and-dining charges for each year in the planning horizon. It is used by a FORTRAN program in the CCP PEER to calculate the peer institution goal target values.

4.4.6 IFPS Model COST

The model COST performs four functions; it: (1) calculates the tuition and fee recovery rate, (2) determines the state revenue requirement goal target values, (3) calculates the decision-variable coefficeints for the percentage-of-average-total-cost goals, and (4) creates a display to assist the user in establishing the change-over-time goals. The IFPS report generator COSTINFO summarizes the results of the calculations performed by the model. An example of the report is presented in Figure 4.24.

4.4.6.1 Tuition and fee recovery rate

The tuition and fee recovery rate establishes the proportion of an institution's operating expenses that must be derived from tuition revenue. The recovery rate may be either directly specified in the data file COST or

COSTINFO IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF COST INFORMATION AND DATA 05/22/84

	1983	1984	1985
ADJUSTED INSTRUCTIONAL E&G	104,515,143	119,142,901	127,722,440
TUITION AND FEE RECOVERY RATE (1)	.3500	.3685	.3689
UNADJUSTED STATE REVENUE RQMT	$36,580,300	$43,909,003	$47,112,991
MISCELLANEOUS FEES	-350,000	-350,000	-350,000
STATE REVENUE REQUIREMENT	$36,230,300	$43,559,003	$46,762,991
CHANGE IN RQMT. OVER PRIOR YEAR	17.52%	20.23%	7.36%
COMPONENTS OF THE TUITION AND FEE RECOVERY RATE			
1. STUDENT ENROLLMENT MIX (2)			
RESIDENT UNDERGRADUATE	.6629	.6565	.6525
NON-RESIDENT UNDERGRADUATE	.1906	.1884	.1870
RESIDENT GRADUATE	.1093	.1156	.1195
NON-RESIDENT GRADUATE	.0372	.0395	.0410
(2)BASED ON THE FOLLOWING WEIGHTED AVG. OF NON-RESIDENT GRADUATE STUDENT EMPLOYMENT RATES			
WEIGHTED AVG EMPLOYMENT RATE	.5000	.5000	.5000
2. COST RATIOS			
UNDERGRADUATE	.6677	.6550	.6471
GRADUATE	2.9358	2.8803	2.8456
3. POLICY RATES			
RESIDENT UNDERGRADUATE	-	25.0000	25.0000
NON-RESIDENT UNDERGRADUATE	-	75.0000	75.0000
RESIDENT GRADUATE	-	25.0000	25.0000
NON-RESIDENT GRADUATE	-	75.0000	75.0000
4. PARTIAL RECOVERY RATES (BY STUDENT CATEGORY)			
RESIDENT UNDERGRADUATE	-	.1075	.1056
NON-RESIDENT UNDERGRADUATE	-	.0926	.0908
RESIDENT GRADUATE	-	.0832	.0850
NON-RESIDENT GRADUATE	-	.0852	.0876
RECOVERY RATE	-	.3685	.3689
BASIS FOR PERCENTAGE OF AVERAGE COST GOAL:			
AVG. ANNUAL COST PER FULL-TIME STUDENT	4876	5551	5940
CONSUMER PRICE INDEX	301.80	316.89	332.73
CHANGE IN CPI	2.90%	5.00%	5.00%

(1) MAXIMUM INCREASE .025/YEAR; MAXIMUM DECREASE .0125/YEAR

Figure 4.24: Summary of the cost data used in the DSS (file COSTINFO).

calculated by the procedure outlined below. The formula used in Virginia to calculate each institution's rate is composed of three components - policy rate, student mix, and cost ratio - as shown in Equation (4.8). The rate is automatically calculated if the value specified in the data file is zero; otherwise the specified value is used.

$$TFRR_k = \sum_{h=1}^{2} \sum_{i=1}^{2} PR_{ik} * SMIX_{hik} * CR_{hk} \quad (4.8)$$

The product of the three components yields a partial recovery rate for each student category. The sum of the partial rates is the institution's recovery rate for year k. The recovery rate calculations are summarized in a management report, as shown in Figure 4.24. The recovery rate formula is based on four student categories: resident and non-resident undergraduate and resident and non-resident graduate students.

The policy rate component, PR_{ik}, is the percentage of total instructional cost that the state believes the student should pay. The variable is only a function of residency status. Currently the policy suggests that Virginia residents (undergraduate and graduate) should pay 25% of the cost of instruction, while non-residents should pay three

times that amount or 75%. The policy rates are stored in the data file COST.

The enrollment mix component is based on projected Fall-term FTE students, previously defined in the model ENROLL section as Equation (4.2). The enrollment values are adjusted for the non-resident graduate students that are considered "significantly employed by the institution for carrying out its educational activities." For the purpose of determining the recovery rate, up to 50% of the non-resident FTE graduate students that are involved in such activities are considered as in-state students. The graduate student employment rate is based on an equally-weighted three-year moving average. The adjusted number of FTE students in each category is expressed as:

$$AFTE_{hik} = \sum_{j=1}^{2} FTE^{1}_{hijk} + ADJ_{hijk} \qquad (4.9)$$

ADJ_{hik} is the allowance for non-resident graduate students employed in educational activities.

In order to increase the number of in-state FTE graduate students:

$$ADJ_{21k} = \sum_{j=1}^{2} FTE^{1}_{22jk} * GER_{k}$$

In order to decrease the number of non-resident graduate students by the same amount:

$$ADJ_{22k} = -ADJ_{21k}$$

The undergraduate enrollment values are unaffected, therefore:

$ADJ_{lik} = 0$

GER_k = non-resident graduate student employment rate - the fraction of non-resident graduate students that are employed by the institution for educational purposes. The data is stored in the file COST; the IFPS model uses a three-year moving average of the historical data.

The enrollment mix component, $SMIX_{hik}$, is expressed in Equation (4.10) as the projected fraction of the total FTE that is included in each category.

$$SMIX_{hik} = \frac{AFTE_{hik}}{\sum_{h=1}^{2}\sum_{i=1}^{2} AFTE_{hik}} \qquad (4.10)$$

The third component, cost ratio, is used in the formula to account for the difference in instructional costs at the two academic levels - undergraduate and graduate. The cost ratio, expressed in Equation (4.11) is defined as either the undergraduate or graduate cost per FTE, divided by the institution's average cost per FTE.

$$CR_{hk} = \frac{COST_{hk}}{\sum_{i=1}^{2} AFTE_{hik} * COST_{hk}} \qquad (4.11)$$

$COST_{hk}$ is the average Educational and General (E&G) direct cost per FTE, by academic level, as reported by the Virginia Information Exchange Procedure (IEP). The values are stored in the file COST.

The model includes the state policy provision that the single-year change in the recovery rate is limited to a maximum increase of 2.5% and a maximum decrease of 1.25%. The percentage-change limits are variables, accessible to the user in the file COST.

4.4.6.2 State revenue requirement goal target value

The right-hand-side value or target value for the state revenue goal, $REVENUE_k$, as defined by Equation (4.12), is the amount of revenue that the institution must generate each year from tuition charges. As shown by Equation (4.12), the goal is the product of two variables. The first, $AIEG_k$, is the adjusted instructional E&G or the annual operating expenses for the university. The value is input in the data file COST. The second variable, $TFRR_k$, is the tuition and fee recovery rate that was discussed in the previous section and defined by Equation (4.8).

$$REVENUE_k = AIEG_k * TFRR_k \qquad (4.12)$$

The calculation is summarized in the top portion of the management report displayed in Figure 4.24.

The IFPS report generator STATEREV transfers the goal target values to a FORTRAN-readable format and stores them in the file STATEREV with the decision-variable coefficients for the state revenue goals.

4.4.6.3 Percentage-of-average-total-cost goals

The IFPS model COST generates the decision-variable coefficients for the percentage-of-cost goals. From the goal formulation in Equation (4.4), the coefficients are defined as $PSCH_h/AVGCOST_k$. $PSCH_h$ is the minimum number of credit hours that a student must carry per academic term to be classified as full-time for tuition purposes. The values are stored in the enrollment file FACTOR by academic level. The average cost per FTE, $AVGCOST_k$, is defined as:

$$AVGCOST_k = AIEG_k \Big/ \left[\sum_{h=1}^{2} \sum_{i=1}^{2} \sum_{j=1}^{2} AFTE_{hijk} \right] \qquad (4.13)$$

where $AIEG_k$ is the institution's annual operating expenses specified in the file COST and $AFTE_{hijk}$ is the number of annual FTE students that was calculated by the IFPS model ENROLL in Equation (4.3). The value of $AVGCOST_k$ is included in the management report displayed in Figure 4.24.

The IFPS report generator PERCOST transfers the decision-variable coefficients for each year in the planning horizon to the data file PERCOST in a FORTRAN-readable format. The values are used by the CCP PERCOST to establish the goals in MPS format.

4.4.6.4 Change-in-tuition-charges-over-time goals

The IFPS report generator INDEXCHG creates a summary of economic indicators that are calculated in the model COST. The indicators are guidelines that help the user establish the change-over-time goal target values in the CCP PERCOST. As shown in Figure 4.25, the display includes the percentage change in the institution's operating expenses, the state revenue requirement, and the CPI for each year in the planning horizon. The display is stored in the file INDEXCHG.

4.4.7 IFPS Model PEER

The purpose of the model is to project tuition charges at peer institutions for each student category. The model is executed from the CCP PEERINST through the IFPS command file PEERCOM.

Recall that in CCP PEERINST the user specified for each student category the data source - e.g., Land-Grant

INDEXCHG IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SUMMARY OF KEY ECONOMIC INDICATORES THAT ARE HELPFUL IN ESTABLISHING THE PERCENTAGE-CHANGE-OVER-TIME GOAL TARGET VALUES

	1983	1984	1985
CHANGE IN THE INSTITUTION'S OPERATING EXPENSES OVER THE PREVIOUS YEAR	8.95%	14.00%	7.20%
CHANGE IN THE STATE REVENUE REQUIREMENT OVER THE PREVIOUS YEAR	17.52%	20.23%	7.36%
CHANGE IN THE CONSUMER PRICE INDEX OVER THE PREVIOUS YEAR	2.90%	5.00%	5.00%

Figure 4.25: Display of key indicators used to establish the change-over-time goals (file INDEXCHG).

institutions, public universities in Virginia - and the statistical measure used to estimate the goals target value - e.g., mean, 25^{th} percentile. Once the information was specified, a separate peer institution working file was created for each student category. The files were named PEERm where m=1,2,...,MM and MM is the number of student categories. Each file contains the historical source data (1978-1983) by percentile (similar to that displayed in Figure 4.13) and the specified measure.

The IFPS command file PEERCOM executes the model PEER once for each student category, i.e, for each file PEERm. The expected peer institution charge is projected for each percentile by simple linear regression, using the CPI as the independent variable. The projection is based on historical peer institution and CPI data from 1978 through 1983. An example of the regression analysis is shown in Figure 4.26 for a single student category and five percentile levels. The solid lines are regression lines "fit" to the historical data; the dashed lines are the projections. Each regression equation, as shown in Equation (4.14), provides an estimate of the p^{th} percentile tuition charge at peer institutions for a particular student category in a given year, $PEER_{hijk}(p)$.

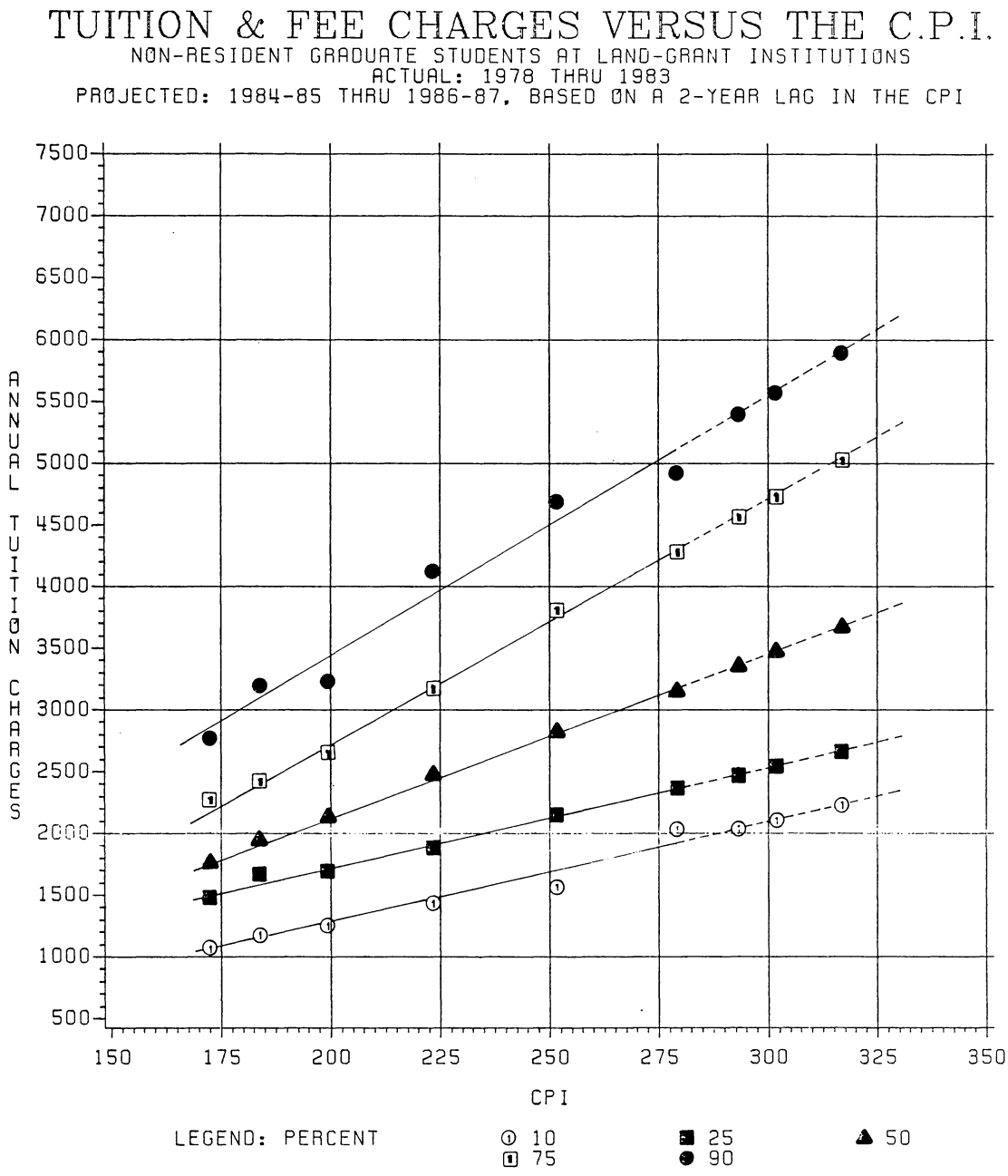


Figure 4.26: Regression analysis of tuition charges versus the CPI.

$$PEER_{hijk}(p) = a_{hijk}(p) + b_{hijk}(p)*CPI_{k-2} \qquad (4.14)$$

$$\text{for } p = 1,5,10,25,50,75,90,95,99$$

Where $a_{hijk}(p)$ and $b_{hijk}(p)$ are the regression coefficients, the intercept and slope, respectively. CPI_{k-2} is the consumer price index for the second year prior. Recall that $PEER_{hijk}$ is the right-hand-side value of Equation (4.6).

If a percentile level, other than one of the nine used in the regression analysis is specified, the model uses linear interpolation to estimate the expected charge. This is represented graphically in Figure 4.27, where for a particular student category all of the percentile values in each year have been connected with linear line segments, thereby forming a distribution of tuition and fee charges. The solid lines are historical data for 1978-1983 and the dashed lines are projections for 1984-1986. For example, the circled point on the middle dashed-line indicates that according to the model, in 1985 60% of the Land-Grant institutions will charge non-resident graduate students $4000 or less per year for tuition and fees.

A two-year lag in the CPI was used as the independent variable in the model because it provided the best "fit" to the data, in terms of R^2 (coefficient of determination). An

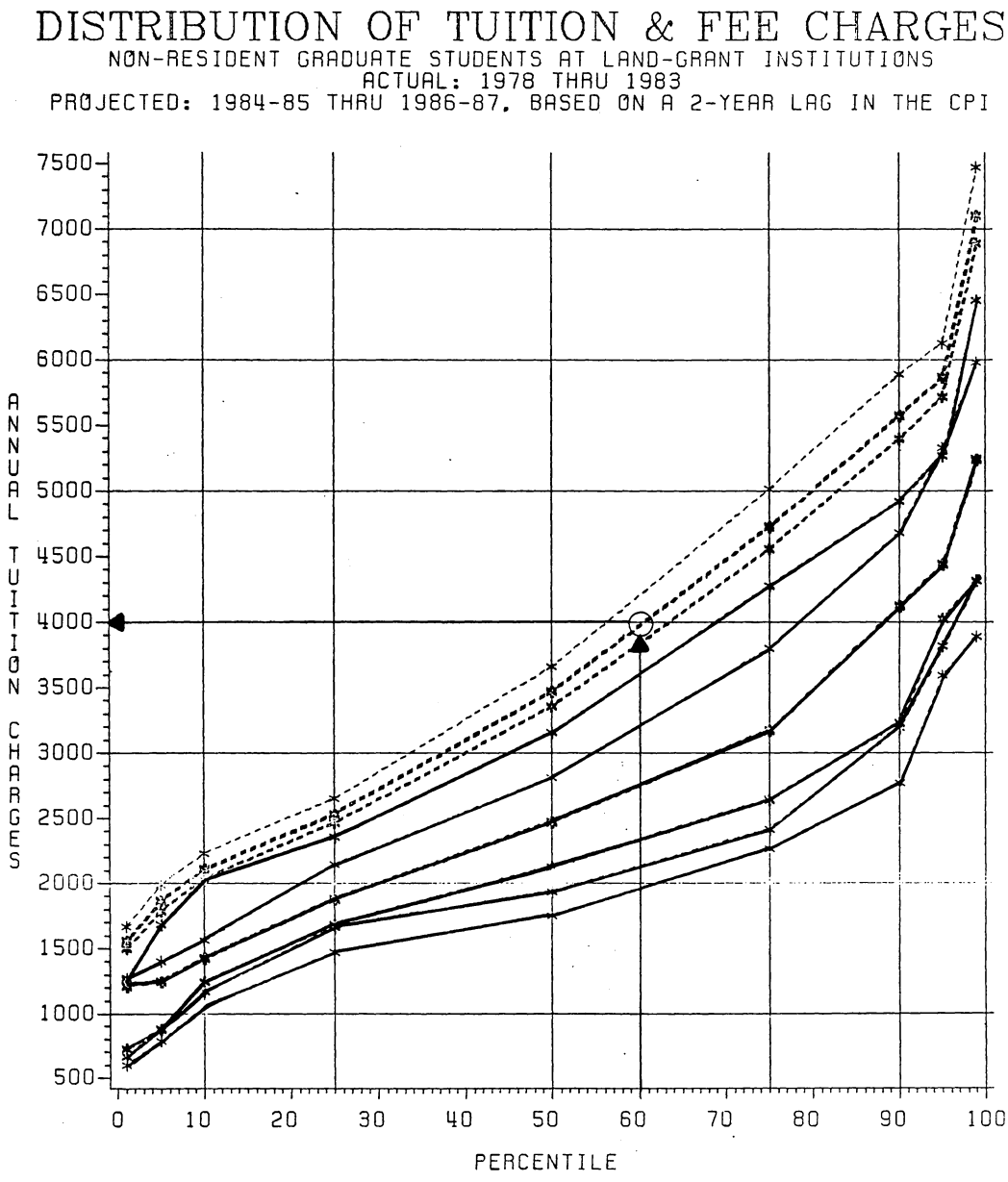


Figure 4.27: Distribution of tuition and fee charges.

analysis was performed on six data sets to see whether a zero, one, or two-year lag provided a better fit. The R^2 value was determined for 180 regression equations - for each data set, an equation was fit for each of the nine percentiles (1,5,10,25,50,75,90,95,99) and the mean, each with a lag of zero, one, and two years. Table 4.5 provides the R^2 values for the mean: the R^2 values for each percentile are listed in Appendix F. As can be seen from Table 4.5, the two-year lag clearly provided the best fit. The pattern was not quite as dominant for the percentiles. For the most part, whenever the R^2 values for the percentiles did not rank as they do in Table 4.5, the data was in the extreme part of the distribution (percentiles 1,5,95,99).

The peer institution values generated by the model PEER for each year in the planning horizon are stored in the file PEERCHG. The IFPS report generator PEERMPS is used to write the values to the file in a FORTRAN-readable format.

4.4.8 IFPS Model TUITION

Unlike all of the other IFPS models, TUITION is used after the goal programming model has been solved and the tuition values have been determined. The model is not executed unless the user selects the TOTCOST report (total

TABLE 4.5

R^2 values for tuition charges versus 0, 1, or 2-year lag in the CPI.

For Land-Grant Colleges and Universities
from 1978 through 1983

Student category	Lag time in the CPI		
	0 year	1 year	2 year
Resident undergraduate Tuition and fees	.8893	.9711	.9955
Non-resident undergraduate Tuition and fees	.8982	.9736	.9965
Resident undergraduate Tuition, fees, room & board	.9345	.9893	.9868
Non-resident undergraduate Tuition, fees, room & board	.9233	.9849	.9923
Resident graduate Tuition and fees	.8859	.9676	.9973
Non-resident graduate Tuition and fees	.9053	.9785	.9941

student charges by category) from the menu displayed by CCP REPORTS. The model's function is to perform the calculations necessary to generate a set of management reports, similar to the one in Figure 4.28. The report summarizes for each student category, the annual charges for a full-time student for each year in the planning horizon and the base year. It also provides the absolute and percentage change between years and relative to the base year. Each student category report is created by a separate IFPS report generator, TOTCOST1, TOTCOST2, etc. All of the reports are stored together in the file TOTCOST.

Before the tuition charges generated by the GP model can be implemented, they must be rounded to the nearest $0.50 per credit hour for undergraduates and to the nearest $1.00 for graduate students. Model TUITION rounds the decision variables and generates a report that compares the expected revenue from the rounded charge structure with the institution's revenue requirements.

4.5 SUMMARY

This chapter has presented the DSS that was developed to allocate a university's multi-year revenue requirements to various student categories. The system was presented in terms of its three subsystems. The highly integrated

TOTCOST IFPS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
FULL-TIME ON-CAMPUS NON-RESIDENT UNDERGRADUATE 04/12/84

	1982	1983	1984	1985
TUITION AND FEES:				
TUITION	$2,322	$2,781	$3,477	$3,576
PERCENT INCREASE OVER PRIOR YEAR	-	19.8%	25.0%	2.8%
PERCENT OF AVERAGE COST	51.4%	57.0%	62.6%	60.2%
FEES:				
ATHLETIC ASSOCIATION FEE	66	72	78	84
BUS SYSTEM FEE	12	12	15	15
STUDENT CENTER FEE	60	63	132	138
HEALTH SERVICE FEE	66	69	72	84
A. TOTAL TUITION AND FEES	$2,526	$2,997	$3,774	$3,897
DORM & DINING:				
BASE ROOM	$444	$474	$507	$534
BASE BOARD	$1,023	$1,035	$1,107	$1,164
NEW FACILITIES	-	29	88	88
B. TOTAL ROOM AND BOARD	$1,467	$1,538	$1,702	$1,786
NON-RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS				
TOTAL CHARGES (A)	$2,526	$2,997	$3,774	$3,897
DOLLAR INCREASE OVER PRIOR YEAR	-	$471	$777	$123
PERCENT INCREASE OVER PRIOR YEAR	-	18.6%	25.9%	3.3%
PERCENT INCREASE OVER BASE YEAR	-	18.6%	49.4%	54.3%
NON-RESIDENT UNDERGRADUATE LIVING ON-CAMPUS				
TOTAL CHARGES (A + B)	$3,993	$4,535	$5,476	$5,683
DOLLAR INCREASE OVER PRIOR YEAR	-	$542	$941	$207
PERCENT INCREASE OVER PRIOR YEAR	-	13.6%	20.8%	3.8%
PERCENT INCREASE OVER BASE YEAR	-	13.6%	37.1%	42.3%

Figure 4.28: Management report that summarizes the annual tuition and fee charges (file TOTCOST).

subsystems, as shown in Figure 4.29, included: (1) the dialog subsystem that controls most of the operations and provides the interface between the user and the system, (2) the data subsystem that provides the memory and information handling functions, and (3) the models subsystem that provides the analytic function for calculating parameter values and solving the goal programming model. The software for the DSS was developed using four programming languages/systems: EXEC 2, FORTRAN, LINDO, and IFPS.

The system was designed to be comprehensive yet easy to use. As this chapter describes, the DSS not only formulates and solves the GP model, it provides a vehicle for the user to effectively consider alternatives and perform sensitivity analyses. The system is intended to support administrative judgements and not replace them. Likewise, the DSS was designed to improve the effectiveness of the decision-making process and just improve its efficiency.

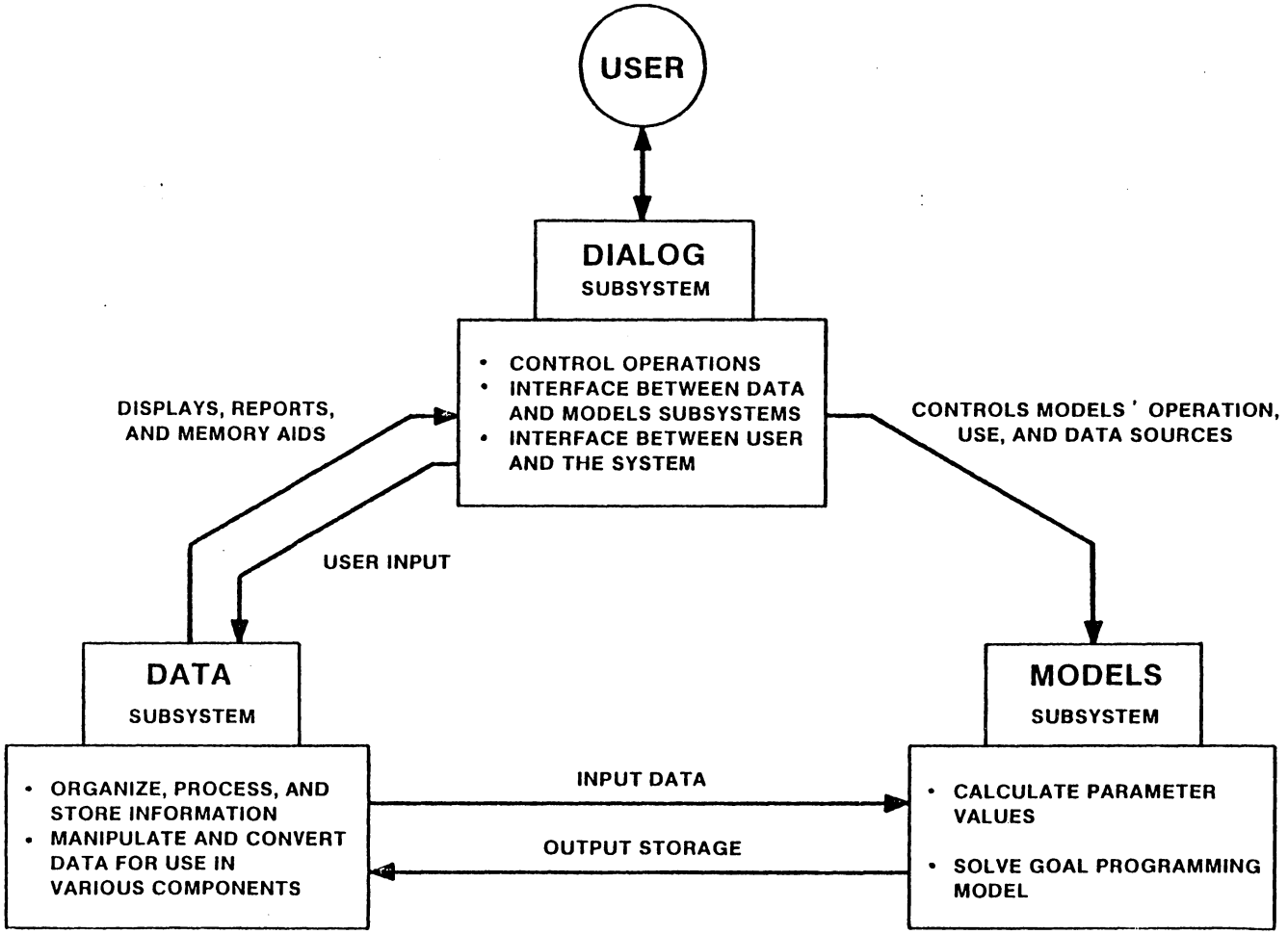


Figure 4.29: Summary of the DSS's three subsystems and their functions.

Chapter V

IMPLEMENTATION, CONCLUSIONS, AND RECOMMENDATIONS

The DSS for tuition-and-fee policy analysis was developed and implemented at Virginia Polytechnic Institute and State University. This chapter describes the implementation process and provides the conclusions and recommendations that have resulted from this research.

5.1 IMPLEMENTATION

The implementation of the DSS has actually been an ongoing process that has existed throughout the development of the system. Since one of the objectives of this research was to test and implement the system at Virginia Tech, user participation and involvement was considered very important and was solicited from the outset. The integration of the DSS into the organization is discussed in the following four sections: (1) evolution of the system, (2) technical testing of the system's components, (3) demonstration of the systems's use and capabilities, and (4) application of the DSS to a real problem situation.

5.1.1 Evolution of the system

At the beginning of this study, four university administrators were identified as being the primary contacts and supporters of this research effort. They were selected because they are directly involved in the tuition-and-fee decision-making process and because they are the ones that will most likely use the system. The "support" group, all from Virginia Tech, included: Mr. Minnis E. Ridenour - Vice-President for Finance, Dr. Laurence J. Moore - Director of Financial Planning and Analysis, Mrs. Katherine M. Johnston - University Budget Director, and Mr. Wayland H. Winstead - Assistant Director of the Office of Institutional Research.

The concept for the development of a DSS for tuition-and-fee policy anlysis evolved from several tuition studies performed at Virginia Tech by the Office of Financial Planning and Analysis and the Office of Institutional Research between June 1982 and June 1983. The studies were primarily concerned with assessing the future impact of changes in state policy and the effect of various capital projects on total student charges. It became apparent from these studies that the process of establishing tuition charges was becoming both increasingly more critical and more complex. It was also recognized that the problem of

establishing tuition charges is a multi-criteria resource allocation problem.

Subsequently, a mathematical model was constructed to represent the allocation of the institution's multi-year revenue requirements to individual student charges. The modeling approach employed was a version of mathematical programming referred to as goal programming. The task of identifying the type of goals that needed to be included in the model involved the review of memos, reports, and correspondence within the university, personal conversations, and a literature search. This information was assimilated into a set of goals that describe the university's tuition philosophy and policy in mathematical form.

The goals that are available in the system were reviewed with the support group on several occassions. The goals were also presented at a meeting of the Virginia Association for Institutional Research on November 15, 1983 as part of the seminar "DSS in College and University Administration". The description of the goals, included as Appendix G, was last reviewed and approved by Mr. Ridenour on February 8, 1984.

As the model was being formulated, it became evident that it would not be utilized to the fullest extent possible

if the administration had to formulate and solve the goal programming model every time a question arose. The administration did not have the resources or the time available to do this unless the process was automated. In addition to formulating and solving the model, the parameter values have to be calculated; this requires a significant number of computations and a considerable amount of data. Recognizing that the decision-making environment in higher education is dynamic, it was obvious that many of these parameter values would change frequently. There was also a strong need to perform "what-if" analyses on a quick turnaround basis. Hence, a comprehensive interactive on-line decision support system would be a necessity if the GP model was to be used.

Most of the DSS was designed and developed between July 1983 and March 1984, with the heaviest concentration of the effort taking place from August 1983 through January 1984. The system was demonstrated to the support group in February 1984 and applied to a real problem in March. The system was modified some during the demonstration and application phases; these phases are discussed in more detail below. Most of the documentation work occurred between March and May 1984.

5.1.2 Technical testing of the system's components

The GP solution algorithm, as described in Section 3.5.3.3, was validated by applying the algorithm to a simple "textbook" problem and checking the results. While the solution of a goal programming model by the Sequential Linear Goal Programming (SLGP) method has been tested, the use of this method in conjunction with LINDO has not. The application of the solution procedure outlined in Figure 3.4 provided the same results as "proven" GP solution methods.

A new solution routine was developed for the DSS because an interactive GP solution package is currently not available. As described in Chapter 3, the Multiphase method for solving GP models is limited to modest size problems. MPS III and MPS X have been used to solve SLGP problems but they are not as conducive to performing interactive operations and generating solutions "on-the-fly"[12] as LINDO.

The interfaces and models within the system were tested as the system was developed. The DSS's modular structure greatly facilitated the development process, since small manageable portions of the system could be developed and debugged before being integrated into the overall system. The system's modularity also makes it conducive to modification, extension, and updating at a later time.

---

[12] solving the sequence of linear programs in one execution of the system.

The philosophy used in the development process was to have the large complex system evolve from a series of small operable and proven components. The system's development was iterative in that quite often the components had to be modified after they were integrated into the system. Modification was necessary because all of the requirements of the system could not be foreseen initially and improvements were identified as the development process progressed. The modular structure again facilitated this process since the component that had to be modified could oftentimes be changed and tested apart from the overall system.

Each of the models that were developed to calculate the GP model's parameter values were tested separately, before being integrated into the system. Likewise, the programs that formulate the goals in MPS format, translate the user's priority structure into an achievement function, and generate displays and reports were all developed and tested outside of the DSS; they were then integrated into the overall system, once they were found to be operating properly. Even the component control programs (CCPs) were developed and tested independently of the executive control program (ECP). The links between the various programs and the interactive interface between the system and the user underwent an evolutionary development as well.

5.1.3 Demonstration of the system's use and capability

Once the DSS had been developed and tested, it was presented to each member of the support group in order to obtain their comments and recommendations. Each presentation involved a "live" demonstration of the system using a hypothetical, but representative set of data, goals, and priorities. A hypothetical example was used in the demonstrations so that as many of the system's features could be highlighted as possible. The comments and suggestions that resulted from each demonstration are summarized below. All of the changes that the support group recommended were incorporated into the DSS.

The strategy for the demonstrations was to execute the system on the computer terminal, but to also provide each person with a hardcopy handout version of the session. The handout made it easier to follow what was happening on the terminal screen and to see the details of the displays. It also provided a means to review and study the output from the system after the session was over. A complete copy of the session handout is included as Appendix H. The first part of the session gave the viewer an overview of the structure and capability of the system. This was accomplishewd by answering "no" to all of the questions posed by the system. Following the overview, the process

was repeated but this time each question posed by the system was answered "yes", thereby exhibiting the details of the system. The session handouts referenced each display that results from an affirmative response to a question posed by the system with a "circled letter". The "circled letter" corresponds to the page in the handout that contains an example of the display.

The system was demonstrated first to Dr. Moore on January 12, 1984. He felt that the system performed well and was ready for review by the other members of the support group. He provided several suggestions for clarifying the displays, instructions, and reports generated by the system. He also indicated that a report should be available to the user that summarizes the level of goal achievement attained by the solution.

The system was next demonstrated to Mr. Ridenour, Dr. Moore, and Dr. Clayton on February 1, 1984. The 2-1/2 hour demonstration resulted in several important modifications to the system. Mr. Ridenour wanted an alternative method for calculating the dorm-and-dining charges included in the system. Initially the charges were directly entered for each dorm group; whereas, he suggested that the overall operating expenses be input and the charges for each group be determined by an allocation based on a set of user-

specified ratios. The demonstration also revealed that the procedure for handling the enrollment data was very confusing. The procedure was re-worked and subsequently found to be much clearer. It was also evident from the meeting that the user should have the choice of starting a session at various points in the process. For example, the user should be able to proceed immediately to the priority structure or to the goal target values without having to step through the entire process.

The system was demonstrated to Mr. Winstead and Dr. Moore on February 8, 1984. The details of the various models, logic of the DSS, assumptions inherent in the solution process, and possible uses of the system were all extensively explored in the three-hour meeting. Mr. Winstead was very enthusiastic about the potential uses of the system in the tuition-and-fee decision-making process. He especially liked having the state recovery rate model incorporated into a system that also considers the individual institution's tuition goals.

The system was introduced to Mrs. Johnston on February 20, 1984. She felt that the system would be very helpful for establishing the upcoming tuition charges since it could quickly generate a series of alternatives. It would also let her assess the effect on tuition charges of changes in

certain variables, e.g. the effect on the amount of revenue required due to a difference between the actual and projected enrollment level and mix in certain student categories.

The system was demonstrated to the dissertation committee - Drs. Clayton, Houck, Moore, Rees, and Taylor - on May 23, 1984. At one point during the session, the system ran out of space on the A-disk. Dr. Clayton suggested that the system, whenever accessed, should test to see if there is enough space available to execute all of the routines. If there is not enough space, then a message should be generated to inform the user of this condition. The suggested modification was incorporated.

5.1.4 Application of the DSS to a real problem

The DSS was integrated into the decision-making process by using it to establish the 1984-85 student charges at Virginia Polytechnic Institute and State University. A meeting was held with Mrs. Johnston on March 3, 1984 to discuss the assumptions, data requirements, and strategy for the application of the DSS to the actual situation. It was decided that the analysis had to be completed within two weeks in order to permit time for an internal review before the Board of Visitors meeting on April 1. It was also

decided that the initial application of the system would be kept similar to the previous approach. Mrs. Johnston wanted to keep the process as simple as possible initially, since there was not sufficient time to educate many of the people involved on the new capabilities incorporated in the DSS (ranking of the goals, direct consideration of peer institution values, etc.). It was decided to perform a one-year analysis and consider a limited number of goals.

The data items needed to run the DSS were discussed and plans were made to meet on March 8 in order to make the initial run of the model. One modification of the model became apparent from this meeting - the system needed to have the capability to handle two sets of enrollment data. Oftentimes there is a difference in the expected enrollment level or mix between the time that the recovery rate is determined and the time that the tuition charges are established. In this case the recovery rate was based on projections made during the Fall quarter of 1983, 1-1/2 years prior to when the tuition charges are actually established.

The March 8 meeting included: Mrs. Johnston, Dr. Moore, and Mr. Purdy. The data were input into the system and the initial priority structure was established. The goals included: (1) meet the state revenue requirement,

(2) maintain the present ratio of tuition charges between student categories,[13] and (3) keep the percentage of average total cost close to 25% for in-state students and below 75% for non-resident students. An error in the system prevented the model from being solved. The problem was in the translation of the priority structure to the achievement function. The meeting was adjourned and we planned to meet again once the error was corrected.

The DSS was now ready to be used to establish the tuition charges for the 1984-85 academic year. During the next week the system was used extensively to incorporate updated information and to consider alternative goal ranking schemes. Some of the models were modified to provide a more detailed summary of the revenue calculations and to change the rounding procedure. After considering several different enrollment projections and several alternative priority structures, a satisfactory solution was found. Mrs. Johnston selected the solution, the tuition charge structure, that she felt best met the goals and needs of the university. Several of the model's values were verified by hand calculations, i.e. total tuition and fee revenue based on the recommended charges and expected enrollment. The

---

[13] Relative to resident undergraduate: non-resident undergraduate - 2.078, on-campus graduate - 1.122, and off-campus graduate - 1.262.

recommended tuition charges were presented to the Board of Visitors on April 1 and were subsequently approved. The solution is discussed below, along with an analysis of the level of goal attainment.

The final solution involved 22 goals and four priority levels. The final solution is provided in Table 5.1; the model's goals, including the target value, level attained, and ranking, are summarized in Table 5.2. As seen in Table 5.2, the first priority was to meet the state revenue requirement plus $300,000 in supplemental revenue for a "safety cushion". This represents a 17.1% increase over the previous year. Since the goals were included in the first priority level, they were achieved.

The goals included in the second priority level were intended to balance the percentage increase in tuition charges across student categories. All of the categories, except for resident undergraduate, increased by the targeted 15.5%; the resident undergraduate only increased 14.8%. These increases all came out below the 17.1% increase in the institution's revenue requirements because of a shift in enrollment. The revenue requirement value is based on enrollment projections that were made 1-1/2 years prior to the current estimate that is used to establish the charge structure.

TABLE 5.1

Summary of the implemented model's solution .

Student category	Decision Variable	Solution Value ($/credit hr.)
Resident Undergrad	X_{1111}	52.79
Non-Resident Undergrad	X_{1211}	113.79
Resident Graduate	X_{2111}	67.76
Non-Resident Graduate	X_{2211}	74.69
Resident Graduate Off-Campus	X_{2121}	79.70
Non-Resident Graduate Off-Campus	X_{2221}	86.63

TABLE 5.2

Summary of the implemented model's level of goal attainment.

Prior. Level	Goal	Goal Target	Level Attained
1	State revenue requirement[1]	At least 45101	45101
1	Supplemental revenue rqmt.[1]	Equal to 300	300
	Change-over-time goals: (percentage increase)		
2	Res UG	Equal to 15.5	14.8
2	Non-Res UG	Equal to 15.5	15.5
2	Res Grad	Equal to 15.5	15.5
2	Non-Res Grad	Equal to 15.5	15.5
2	Res Grad O-C	Equal to 15.5	15.5
2	Non-Res Grad O-C	Equal to 15.5	15.5
	Absolute goals: (dollars per credit hour)		
2	Non-Res Grad - Res Grad	Equal to 6.00	5.07
2	Res Grad O-Camp - Res Grad	Equal to 10.77	9.61
2	Non-Res Grad - Res Grad - Res Grad O-C - Non-Res Grad O-C	Equal to 0.00	0.00
	Relative goals: (unitless)		
3	Non-Res UG / Res UG	Equal to 2.078	2.155
3	Res Grad / Res UG	Equal to 1.122	1.284
3	Non-Res Grad / Res UG	Equal to 1.122	1.415
3	Res Grad O-C / Res UG	Equal to 1.262	1.508
3	Non-Res Grad O-C / Res UG	Equal to 1.262	1.641
	Percentage-of-average-total-cost goals: (percent)		
4	Res UG	Less than 25.0	29.3
4	Non-Res UG	Less than 75.0	63.1
4	Res Graduate	Less than 25.0	37.6
4	Non-Res Graduate	Less than 75.0	41.5
4	Res Graduate O-C	Less than 25.0	44.2
4	Non-Res Graduate O-C	Less than 75.0	48.1

[1] in thousands of dollars.

Also included in the second priority level, as shown in Table 5.2, were the goals to: (1) maintain a $6.00 per credit-hour difference between non-resident and resident graduate students, (2) establish a $10.77 per credit-hour difference between off-campus and on-campus graduate programs, and (3) maintain the same difference between resident and non-resident graduate students in both the on-campus and off-campus programs. Only the third goal was achieved; the actual difference for the first goal was $5.07 and the difference for the second goal was $9.61.

The goals included in the third priority level attempted to maintain for each student category the same relative tuition charge as the previous year (e.g. ratio of non-resident to resident undergraduate charge was 2.078 to 1). As can be seen from Table 5.2, none of these goals were completely achieved.

The final priority level was used to limit the percentage of average total cost paid by the various student categories. The goals to keep resident charges below 25% were not met but the goals to keep the non-resident charges below 75% were achieved.

The goal summary in Table 5.2 highlights two potential problems. The first is that the percentage of average total cost actually attained for the in-state student categories are significantly higher than the goal target values (fourth priority goals). Likewise, the tuition charges for each student category, relative to the resident undergraduate charge, are also significantly higher than the targets (third priority goals). The administration should review these conditions and experiment with alternative charge structures that bring these measures more in line with the targets. It would be benefical to use a multi-year model in order to see if maintaining the current policies will cause these measures to diverge beyond acceptable limits.

5.2 CONCLUSIONS

This dissertation demonstrates how computer hardware and software technology can be combined with management science modeling and analysis to provide an extremely powerful and effective package for making sound and effective tuition-and-fee policy decisions in an on-line interactive mode. The product of this research is a decision support system that provides a generalized and systematic approach to tuition analysis. It not only shows that management science techniques can be applied to

problems in higher education, but that such techniques and systems can be implemented and used directly by administrators to support their planning, decision-making, and policy-setting processes.

Conclusions are drawn from three areas: (1) the use of goal programming to model the tuition process, (2) the use of an automated system to formulate and solve the GP model, and (3) the translation of the GP model into a valuable decision-making aid. Each of these are described in more detail in the following sections.

5.2.1 The use of goal programming to model the tuition process

This research demonstrates that a goal programming model of the tuition and fee process provides the administration with a rational and tractable, yet flexible, means to analyze and establish tuition charges. The GP model provides a systematic means to allocate the university's revenue requirements to various student categories based on the institution's goals, limitations, and priorities. While the GP model provides structure to the tuition and fee allocation problem, it also provides the user with considerable flexibility. The user controls which goals are included in the model, the data used to calculate the model's parameter values, and the ranking of each goal in the order of importance to the university.

The GP model provides a medium for effectively dealing with the complex and critical process of establishing tuition charges. Since the model integrates and systematically considers a set of goals, it provides a way for the administration to balance the stark financial considerations with university philosophical considerations. The model's goals include: state revenue requirements, supplemental revenue needs, charges at peer institutions, change in tuition charges over time, students' share of the institution's cost of operation, and a wide range of relative and absolute relationships between student categories.

5.2.2 The use of an automated system to formulate and solve GP models

Through the research of this dissertation, a method was developed to solve goal programming models with the commercial interactive linear programming package LINDO. A FORTRAN "buffer" code was written to interface with LINDO and to control the solution procedure. The code translates the priority structure provided by the user into an achievement function, decomposes the GP model by priority level into a series of single-objective problems, and subsequently solves it by the simplex algorithm included in LINDO. The code also provides several reports for use by

the decision maker to assess and analyze the solution. One report provides a summary of the goal achievements, i.e. the target value and the level attained by each goal in the model's solution. Another report provides a trace of the decision-variable values at each priority level; this illustrates both the degree of conflict that exists between the goals and the sensitivity of the solution to the priority structure.

The dissertation provides a means to "generate" the goal programming model described by the user. A system was developed that not only solves the GP model, but formulates it and calculates its parameter values. The arduous task of model formulation and solution is performed by the system; whereas, the user is responsible for defining the problem framework, selecting the goals, establishing the priority structure, and assessing the solution.

5.2.3 Translation of the GP model into a decision-making aid

The primary focus of this dissertation was to translate the GP model into a valuable decision-making aid by making it directly and readily accessible to the administration. This involved the development of a system that formulates the goals selected by the user, calculates the model's parameter values, provides a means to establish the priority

structure, solves the model, and summarizes the solution. In order for these tasks to be controlled by a user that is not necessarily well versed in management science and computers, the system had to be designed to: (1) provide the decision maker with an active role in the final solution, (2) include a series of analytical techniques and models, (3) provide a user-friendly interface between the models and data and the user, (4) process and store a significant amount of data, (5) provide immediate feedback, (6) integrate the functional routines, (7) provide displays, memory aids, and reports, (8) be flexible in order to permit modification, extension, and updating, and (9) control the overall process.

The decision support system places the goal programming approach into the hands of the decision maker. The decision making process is controlled directly by the user. Tuition and fee problems are addressed at a detailed and sophisticated level without directly handling large amounts of data, structuring the problem, formulating and solving complex analytical models, and composing a series of reports to document the results.

This dissertation demonstrates how an interactive user-friendly decision support system with an imbedded multi-year GP model can be used to:

1. forecast or project future values,
2. deal with complex dynamic multi-dimensional problems in a rational manner,
3. generate and evaluate alternative plans and courses of action,
4. perform sensitivity analyses and address "what if" questions,
5. "design" tuition charge structures,
6. decide how to phase in new policies or changes in the tuition charge structure, and
7. assess the impact of policy changes and analyze the effect of changes in factors that influence tuition charges.

This research also showed that a comprehensive system, such as the tuition-and-fee DSS, cannot be developed using a single language. The decision support system was developed using four programming languages/systems: EXEC 2, IFPS, FORTRAN, and LINDO. The best features and capabilities of each language were utilized in the system's software design and development.

Even IFPS, a powerful interactive financial planning language that is often touted as a language for DSSs, did not provide all of the features needed to develop this system. Although IFPS includes command, control, and logic

statements, they are not as powerful or as flexible as those included in the IBM command programming language EXEC 2. The use of EXEC 2 provides better interactive communication with the user, a more convenient means to interface with CMS, more capability for manipulating data files, and more flexibility for process control and logical operations.

IFPS also does not have the capability to solve multi-criteria optimization problems. Therefore, the commercial mathematical programming package LINDO was used to provide the software for the simplex solution algorithm and a FORTRAN subroutine was used to provide the logic to solve the goal programming model as a sequence of linear programming problems. FORTRAN was found to be the easiest language to use for formulating the GP model in MPS format. Also, some of the data files and reports were created using FORTRAN.

IFPS was especially useful for developing models to calculate the parameter values. The modeling task was facilitated by several of IFPS's prominent programming characteristics, including its: natural non-procedural language, spread-sheet format, incorporation of leading and lagging variables, and built-in functions and subroutines (amortization, simple linear regression, rounding, etc.). IFPS also provides a convenient means to create, modify, and

use data files. Its report generating capability was very useful for creating displays and management reports.

In summary, it was found that EXEC 2 was best for the purpose of user interface, process control, and data file manipulation. IFPS was best for defining the relationships and performing the financial/quantitative computations and projections that are necessary to generate the GP model's parameter values. IFPS was also used to generate the system's reports and to organize, change, manipulate, and update data files. The DSS utilized the existing mathematical programming software in LINDO to assist in the solution of the GP model. FORTRAN was used primarily to formulate and solve the GP model. It was also used to create and modify some of the display, data, and report files.

5.3 RECOMMENDATIONS

As discussed in a previous section, the DSS was used to establish the 1984-85 tuition charges at Virginia Polytechnic Institute and State University. Based on this successful application of the system, the university should continue to use the system to establish its charges in the future. The system should not only be used at an operational level, but should be used as a strategic

planning tool. It is strongly recommended that the multi-year feature of the model be utilized. This will enhance the planning process, identify potential problems with present policies, provide a means to decide how to implement changes, and provide a vehicle to assess the impact of changes in factors that effect tuition charges (state policy, enrollment level and mix, inflation, etc.)

It is also recommended that other public colleges and universities should consider the use of the decision support system with its imbedded goal programming model to allocate their resource requirements to student categories. It should not only be used to establish their annual tuition charge structure but should be used as a tool to enhance their planning, decision-making, and policy-setting processes.

With regard to foregoing research, several areas could use additional work; such areas for future research include:

1. Integrating the budget model for the institution into the DSS, thereby providing a means to directly assess the effect of changes in budget variables on tuition charges.
2. Additional experimentation with the system should be conducted to gain further insight into the effects of changes in various variables and factors that influence tuition charges.

3. Additional research to develop functional relationships that directly incorporate the effect of price changes on demand would be beneficial.
4. The current system could be expanded to include tuition charge functions other than the "per-credit-hour-to-plateau" function.
5. The administrators that will use the system should experiment with the priority structure so that they can gain insight into the sensitivity of the goal rankings on tuition charges. They need to better understand the effects of using a preemptive priority structure.

In summary, this research represents only the beginning of what could develop into a comprehensive and sophisticated computer-based financial planning and analysis system. The opportunities for continued productive research and development in this area are impressive and exciting.

BIBLIOGRAPHY

1. Alavi, M. "An Assessment of the Concept of Decision Support Systems as Viewed by Senior-Level Executives" MIS Quarterly. December 1982, p. 1-9.

2. Awerbuch, S., Ecker, J. G., Wallace, W. A. "A Note: Hidden Nonlinearities in the Application of Goal Programming" Management Science. April 1976, vol. 22, no. 8, p. 918-920.

3. Bennett, J. L. "User Oriented Graphics, Systems for Decision Suppport in Unstructured Tasks" in User-Oriented Design of Interactive Graphics Systems, S. Treu (ed.) New York: Association for Computing Machinery, 1977, p. 3-11.

4. Bennett, J. L. (Ed.) Building Decision Support Systems. Reading, MA: Addison-Wesley Publishing Company, 1983.

5. Blanning, R. W. "The Function of a Decision Support System" Information and Management. 1979, vol. 2, no. 3, p. 87-93.

6. Bloomfield, S. D., & Updegrove, D. A. "A Modeling System for Higher Education" Decision Sciences. April 1981, vol. 12, no. 2, p. 310-321.

7. Bloomfield, S. D., & Updegrove, D. A. "Modeling for Insight, Not Numbers" In J. A. Wilson (Ed.) Management Science Applications to Academic Administration. (New Directions for Higher Education Series, vol. IX, no. 3, September 1981), San Francisco, CA: Jossey-Bass, Inc. Publishers, p. 93-104.

8. Bruno, F. & Swanson, L. H. "A Resource Planning Model for the Determination of Faculty Size and Student Admission Targets in Institutions of Higher Education" In Proceedings. Atlanta, GA: American Institute for Decision Sciences, 1977.

9. Carbone, R. F. Alternative Tuition Systems. Iowa City, Iowa: The American College Testing Program, 1974.

10. Carlson, E. D., Grace, B. F., Sutton, J. A. "Case Studies of the End User Requirements for Interactive Problem-Solving Systems" Management Information Systems Quarterly. 1977, vol. 1, p. 51-63.

11. Carlson, E. D. "An Approach for Designing Decision Suport Systems" Data Base. Winter 1979, p. 3-15.

12. Carnegie Commission on Higher Education Higher Education: Who Pays? Who Benefits? Who Should Pay? New York, NY: McGraw-Hill Book Company, 1973.

13. Carter, E. E. College Financial Management. Lexington, MA: Lexington Books, D.C. Heath and Company, 1980.

14. Centra, J. A. "College Enrollments in the 1980s, Projections and Possibilities" Journal of Higher Education. 1980, vol. 51, no. 1, p.18-39.

15. Chambers, M. M. Higher Education: Who Pays? Who Gains? Danville, Illinois: The Interstate Printers & Publishers, Inc. 1968.

16. The Chronicle of Higher Education, August 3, 1983.

17. Cyert, R. M. "Management Science and University Management" In J. A. Wilson (Ed.) Management Science Applications to Academic Administration. (New Directions for Higher Education Series, vol. IX, no. 3, September 1981), San Francisco, CA: Jossey-Bass, Inc. Publishers, p. 27-40.

18. Dauer, J. P., & Krueger, R. J. "An Interactive Approach to Goal Programming" Operational Research Quarterly, 1977, vol. 28, no. 3, ii, p. 671-681.

19. EXECUCOM IFPS User Defined FORTRAN User's Guide (Revision D). Austin, TX: EXECUCOM Systems Corporation, 1982.

20. EXECUCOM IFPS User's Manaul (Release 9.0). Austin, TX: EXECUCOM Systems Corporation, 1983.

21. Frances, C. The Short Run Economic Outlook for Higher Education. Washington, D. C.: American Council on Education, 1980.

22. Franz, L. S., Lee, S. M., & VanHorn, J. C. "An Adaptive Decision Support System for Academic Resource Planning" Decision Sciences. April 1981, vol. 12, no. 2, p. 276-293.

23. Fuller, B. "A Framework for Academic Planning" Journal of Higher Education. January/February 1976, vol. 47, no. 1, p. 65-77.

24. Geoffrion, A. M., Dyer, J. S., & Feinberg, A. "An Interactive Approach to Multicriteria Optimization, With an Application to the Operation of an Academic Department" Management Science. December, Part I, 1972, vol. 19, no. 4, p. 357-368.

25. Glenny, L. A. "Demographic and Related Issues for Higher Education in the 1980s" Journal of Higher Education. 1980, vol. 51, no. 4, p. 363-380.

26. Gorry, G. A., & Scott Morton, M. S. "A Framework for Mangement Information Systems" Sloan Management Review. Fall 1971, p. 55-70.

27. Gray, P. Student Guide to IFPS. New York: McGraw-Hill Book Company, 1983.

28. Hannan, E. L. "Effects of Substituting a Linear Goal for a Fractional Goal in the Goal Programming Problem" Management Science. September 1977, vol. 24, no. 1, p. 105-107.

29. Hannan, E. L. "Allocation of Library Funds for Books and Standing Orders - A Multiple Objective Formulation" Computers & Operations Research, 1978, vol. 5, p. 109-114.

30. High School Graduates: Projections for the Fifty States. Boulder, CO: Western Interstate Commission for Higher Education (WICHE), November 1979.

31. "Higher Education Prices and Price Indexes: 1982 Update" Research Associates of Washington. Washington, D. C., October 1982.

32. Hopkins, D. S. P., Larreche, J., & Massy, W. F. "Constrained Optimization of a University Administrator's Preference Function" Management Science. December 1977, vol. 24, no. 4, p. 365-377.

33. Hopkins, D. S. P. "Computer Models Employed in University Administration: The Stanford Experience" INTERFACES. February 1979, vol. 9, no. 2, pt. 1, p. 13-23.

34. Hopkins, D. S. P., & Massy, W. F. Planning Models for Colleges and Universities. Stanford, CA: Stanford University Press, 1981.

35. Hopkins, D. S. P., Landry, L. L., Sonenstein, B. & Tschectelin, J. D. "Financial Modeling: Four Success Stories" EDUCOM Bulletin. Fall 1982, vol. 17, no. 3, p. 11-15.

36. Hwang, C. L., & Masud, A. S. M. Multiple Objective Decision Making - Methods and Applications. Lecture Notes in Economics and Mathematical Systems, vol. 164, New York: Springer-Verlag, 1979.

37. Ignizio, J. P. Goal Programming and Extensions. Lexington, MA: D. C. Heath and Company, 1976.

38. Ignizio, J. P., "A Review of Goal Programming: A Tool for Multiobjective Analysis", Journal of the Operational Research Society. 1978, vol. 29, p. 1109-1119.

39. Ignizio, J. P., "Sequential Linear Goal Programming: Implementation Via MPSX", Computers and Operations Research. 1979, vol. 6, p. 141-145.

40. Ignizio, J. P. Linear Programming in Single and Multiple Objectives. Englewood Cliffs, NJ: Prentice-Hall, 1982.

41. IBM Virtual Machine / System Product: EXEC 2 Reference Program Number 56C4-167. Endicott, NY: IBM Corp., 1982.

42. Johnson, J. L. "Setting Tuition Levels at Public Institutions: The Case of the University of Washington" Journal of Higher Education. March/April 1976, vol XLVII, no. 2, p. 125-139.

43. Johnson, J. L. "An Analysis of the Relationship Between Instructional Costs and Differential Tuition Levels" Journal of Higher Education. 1979, vol. 50, no. 3, p. 280-288.

44. Johnstone, J. J. "Mathematical Models Developed for Use in Educational Planning: A Review" Review of Educational Research. 1976, vol. 44, no. 2, p. 177-201.

45. Keen, P. G., & Wagner, G. R. "DSS: An Executive Mind-Support System" Datamation November 1979, p. 117-122.

46. Keen, P. G. W. "Decision Support Systems Translating Analytic Techniques into Useful Tools" Sloan Management Review. Spring 1980, vol. 21, p. 33-44.

47. Keen, P. G. W. "Decision Support Systems: Lessons for the 80s" EDUCOM Bulletin. Fall 1982, vol. 17, no. 3, p. 17-21.

48. Keller, G. Academic Strategy - The Management Revolution in American Higher Education. Baltimore, MD: The Johns Hopkins University Press, 1983.

49. Kendall, K. E., & Luebbe, R. L. "Management of College Recruiting Activities Using Goal Programming" Decision Sciences. April 1981, vol. 12, no. 2, p. 193-205.

50. Keown, A. J., Taylor, B. W., & Pinkerton, J. M. "Multiple Objective Capital Budgeting Within the University" Computers & Operations Research. 1981, vol, 8, p. 59-70.

51. King, W. R. "Using and Evaluating Administrative Decision Support Systems" In J. A. Wilson (Ed.) Management Science Applications to Academic Administration. (New Directions for Higher Education Series, vol. IX, no. 3, September 1981), San Francisco, CA: Jossey-Bass, Inc. Publishers, p. 63-71.

52. Klein, R. "Computer-Based Financial Modeling" Journal of Systems Management. May 1982, p. 6-13.

53. Knodle, L. L., Noval, L., Powell, J., Rogers, F., & Updegrove, D. "EFPM: User's Experiences at Purdue University, Oberlin College, and Carnegie-Mellon University" EDUCOM Bulletin. Summer 1979, vol. 14, no. 2, p. 5-11.

54. Koch, J. V. "A Linear Programming Model of Resource Allocation in a University" Decision Sciences. 1973, vol. 4, p. 494-504.

55. Lawrence, G. B., & Service, A. L. (Eds.) Quantitative Approaches to Higher Education Management (ERIC/Higher Education Research Report No. 4, 1977). Washington, D. C.: American Association for Higher Education, 1977.

56. Lee, S. M. Goal Programming for Decision Analysis. Philadelphia, PA: Auerbach Publishers, 1972.

57. Lee, S. M. & Clayton, E. R. "A Goal Programming Model for Academic Resource Allocation" Management Science. April 1972, vol. 18, no. 8, p. d395-B408.

58. Lee, S. M., & Moore, L. J. "Optimizing University Admissions Planning" Decision Sciences. 1974, vol. 5, p. 405-414.

59. Lee, S. M., & Moore, L. J. "Multi-criteria school busing models" Management Science March 1977, vol. 23, no. 7, p. 704-715.

60. Lee, S. M., Van Horn, J., & Brisch, H. "A Multiple Criteria Analysis Model for Academic Policies, Priorities, and Budgetary Constraints" In G. Fandel & T. Gal (Eds.), Multiple Criteria Decision Making Theory and Applications. Lecture Notes in Economics and Mathematical Systems vol. 177. New York: Springer-Verlag, 1980, p. 218-237.

61. Leslie, L. L. "The Financial Prospects for Higher Education in the 1980s" Journal of Higher Education. 1980, vol. 51, no. 1, p. 1-17.

62. Lin, W. T. "A Survey of Goal Programming Applications", Journal of the Operational Research Society. vol. 8, no. 1, 1980, p. 115-122

63. McKeown, M. P. "State Policies on Tuition and Fees for Public Higher Education", Journal of Education Finance. Summer 1982, vol. 8, p. 1-19.

64. McNamara, J. F. "Mathematical Programming Models in Educational Planning" Review of Educational Research. 1971, vol. 41, no. 5, p. 419-446.

65. McNamara, J. F. "Mathematical Programming Applications in Educational Planning" Socio-Economic Planning Sciences. 1973, vol. 7, p. 19-35.

66. Massy, W. F. "Reflections on the Application of a Decision Science Model to Higher Education" Decision Sciences. 1978, vol. 9, 362-369.

67. Massy, W. F. "Decision Science in Academic Administration" Decision Sciences. April 1981, vol. 12, no. 2, p. 167-174.

68. Moore, J. H., & Chang, M. G. "Design of Decision Support Systems" Data Base Fall 1980, vol. 10, no. 1 and 2, p. 8-13.

69. Moyer, R. C., & Kretlow, W. J. "The Resource Allocation Decision in U.S. Colleges and Universities: Practice, Problems and Recommendations" Higher Education. 1978, vol. 7, p. 35-46.

70. Muriel A. C. "A Management Support Model for Community Colleges" Research in Higher Education. 1982, vol. 16, no. 3, p. 209-223.

71. Myers, M. M. Fact Book on Higher Education in the South 1981 and 1982. Atlanta, GA: Southern Regional Education Board, 1982.

72. National Association of State Universities and Land-Grant Colleges through 1983-84 Student Charges at State and Land-Grant Universities. Washington, 1977-1983.

73. Naylor, T. H., & Mann, M. H. (Eds.) Computer Based Planning Systems Oxford, OH: Planning Executives Institute, 1982.

74. Pollack, M. "Interactive Models in Operations Research - An Introduction and Some Future Research Directions" Computers and Operations Research 1976, vol. 3, p. 305-312.

75. Predicasts, Inc. Predicasts Forecasts. Issue 94, 2nd quarter. Cleveland, OH: Predicasts, Inc., January 18, 1984.

76. Rusk, J. J., & Leslie, L. L. "The Setting of Tuition in Public Higher Education" Journal of Higher Education. 1978, vol, 49, no. 6, p. 531-547.

77. Said, K. E. A Budgetary Model for an Institution of Higher Education. Austin, Texas: Bureau of Business Research, Graduate School of Business, The University of Texas at Austin, 1974.

78. Saupe, J. L. & Blagg, R. E. "Student Charges at Twenty Major Universities: Can the Data be Compared ?" Research and Planning for Higher Education. In R. H. Fenske and P. J. Staskey (Eds.), Seventeenth Annual Forum Proceedings, The Association for Institutional Research, Montreal, Quebec, 1977, p. 59-62.

79. Schietinger, E. F. Tuition Policy for the Eighties. Atlanta, GA: Southern Region Education Board, 1981.

80. Schrage, L. User's Manual for LINDO. Palo Alto, CA: The Scientific Press, 1981

81. Schrage, L. Linear Programming Models With LINDO. Palo Alto, CA: The Scientific Press, 1981

82. Schroeder, R. G. "A Survey of Management Science in University Operations" Management Science. April 1973, vol. 19, no. 8, 895-906.

83. Schroeder, R. G. "Resource Planning in University Management by Goal Programming" Operations Research. July-August 1974, p. 700-710.

84. Schroeder, R. G. & Adams, C. R. "The Effective Use of Management Science in University Administration" Review of Educational Research. Winter 1976, vol. 46, no. 1, p. 117-131.

85. Sprague, R. H., Jr. & Carlson, E. D. Building Effective Decision Support Systems. Englewood Cliffs, NJ: Prentice-Hall, 1982.

86. Spronk, J. Interactive Multiple Goal Programming. Boston, MA: Martinus Nijhoff Publishing, 1981.

87. Stewart, I. R., & Dickason, D. G. "Hard Times Ahead" American Demographics. June 1979, p. 12-23.

88. Troutt, M. D. "Deciding Tuition Structure with Linear Programming" Research in Higher Education. 1983, vol. 18, no. 3, p. 359-371.

89. U. S. Council of Economic Advisers. Supplement to Economic Indicators (Historical and Descriptive Background). Washington: Government Printing Office, 1980.

90. U. S. Council of Economic Advisers. Economic Indicators. February 1976 - February 1984, Washington: Government Printing Office, 1976-84.

91. Van Alstyne, C. "Rationales for Setting Tuition Levels at Public Institutions" Educational Record. Winter 1977, vol. 58, p. 66-82.

92. Vazsonyi, A. "Information Systems in Management Science" INTERFACES. November 1978, vol. 9, no. 1, p. 72-77.

93. Viehland, D. W., Kaufman, N. S., & Krauth, B. M. "Indexing Tuition to Cost of Education: The Impact on Students and Institutions" Research in Higher Education. 1982, vol. 17, no. 4, p. 333-343.

94. Walhaus, R. A. "Institutional Research in the Eighties" In P. J. Staskey (Ed.), Issues for the Eighties. Nineteenth Annual Forum Proceedings No. 2, The Association for Institutional Research, San Diego, 1979, p. 21-26.

95. Walters, A., Mangold, J. & Haran, E. G. P. "A Comprehensive Planning Model for Long-Range Academic Strategies" Management Science. March 1976, vol. 22, no. 7, p. 727-738.

96. Wehrung, D. A., Hopkins, D. S. P., & Massy, W. F. "Interactive Preference Optimization for University Administrators" Management Science. 1978, vol. 24, p. 599-611.

97. Welzenbach, L. F. (Ed.) College & University Business Administration (4th Edition). Washington, D. C.: National Association of College and Business Officers, 1982.

98. Wilson, J. A., (Ed.) Management Science Applications to Academic Administration. (New Directions for Higher Education Series, vol. IX, no. 3, September 1981), San Francisco, CA: Jossey-Bass, Inc. Publishers, 1981.

99. Wyatt, J. B., Emery, J. C., & Landis, C. P. (Eds.) Financial Planning Models: Concepts and Cases in Colleges and Universities Princeton, NJ: EDUCOM, 1979.

100. Zeleny, M. Multiple Criteria Decision Making. New York, NY: McGraw-Hill Book Company, 1982.

Appendix A

PROGRAM LISTINGS FOR THE EXECUTIVE CONTROL PROGRAM (ECP)

```
&TRACE
*************************************************************************
*************************************************************************
*                                                                       *
* EXECUTIVE CONTROL PROGRAM                                             *
*                                                                       *
*************************************************************************
*************************************************************************
*                                                                       *
*       VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY             *
* DECISION SUPPORT SYSTEM FOR THE TUITION AND FEE POLICY ANALYSIS       *
*                                                                       *
*************************************************************************
*************************************************************************
*                                                                       *
* DEFINE STORAGE, LINK TO LINDO AND IFPS                                *
* CHECK TO SEE IF THERE IS ENOUGH DISK SPACE                            *
*                                                                       *
*************************************************************************
*CP DEF STOR 1400K
*CP IPL CMS
*EXEC LDISK LINDO
*EXEC LDISK IFPS
SET CMSTYPE HT
STACKCMS Q DISK A
&READ VARS
&READ VARS &A &B &C &D &E &F &G &H &I &J
SET CMSTYPE RT
DESBUF
&IF &J GT 50   &GOTO -GO
&TYPE NOTE !!!!!!!
&TYPE THERE ARE ONLY &J BLOCKS LEFT WHICH MAY NOT BE ENOUGH TO RUN
&TYPE THE SYSTEM - YOU MAY WANT TO OBTAIN ADDITIONAL DISK SPACE
&TYPE
&TYPE DO YOU WANT TO CONTINUE THIS SESSION ?
&TYPE >>> ENTER YES OR NO
&TYPE
-CHK &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -GO
&IF &RESPONSE EQ NO &GOTO -BYE
&IF &RESPONSE EQ Y &GOTO -GO
&IF &RESPONSE EQ N &GOTO -BYE
&TYPE
&TYPE >>> ENTER YES OR NO
&TYPE
&GOTO -CHK
-GO
*************************************************************************
*                                                                       *
* SET PARAMETERS / INITIALIZATIONS                                      *
*                                                                       *
*************************************************************************
```

```
*
*&STARTYEAR = FIRST YEAR OF DATA
* &MAXNYEAR = MAXIMUM NUMBER OF YEARS AVAILABLE IN THE PLANNING HORIZON
* &UNO = UNIT NUMBER, USED BY LINDO, TO IDENTIFY THE FILE CONTAINING
*    THE PROBLEM IN MPS FORMAT.
*    DO NOT USE 1-5, 10-15.
* &FILETYPE = FILETYPE OF THE FILE CONTAINING THE PROBLEM IN MPS FORMAT
* &NREPORTS = NUMBER OF MANAGEMENT REPORTS AVAILABLE
* &DDOPTION = DORM AND DINING CALCULATION OPTION; DEFAULTS TO
*    OPTION 2 - ALLOCATE OPERATING EXPENSES TO INDIVIDUAL UNITS.
&STARTYEAR = 1978
&MAXNYEAR = 6
&UNO = 08
&UN1 = &CONCAT OF FT &UNO
&FILETYPE = &CONCAT OF &UN1 F001
&NREPORTS = 16
&DDOPTION = 2
-START
&ABSCHGIND = 0
&COMPIND = 0
&COSTIND = 0
&ENROLLIND = 0
&PHCHGIND = 0
&SKIPMPS = 0
*
*************************************************************************
*                                                                       *
* DETERMINE EXISTING PROBLEM FRAMEWORK                                  *
*        BY READING IN DATA FROM FILE SETUP DSS A                       *
*                                                                       *
*************************************************************************
*                                                                       *
*  READ-IN THE FIRST 11 VALUES, THEN READ-IN THE TWO ENROLLMENT         *
*    FILE NAMES (PROJECTED FOR STATE RECOVERY RATE AND ACTUAL FOR       *
*    EXPECTED REVENUE), THEN READ-IN THE LAST ITEM - TIME SPAN          *
*    OF THE MODEL.                                                      *
*                                                                       *
&NSET = 13
&NSET2 = &NSET - 2
EXECIO &NSET DISKR SETUP DSS A (FINIS
&LOOP -LOOP1 &NSET2
&READ VARS &NAME &EQUAL &VALUE
*&TYPE &NAME &EQUAL &VALUE
&&NAME = &VALUE
-LOOP1
&READ VARS &NAME &EQUAL &PENROLL &AENROLL
&READ VARS &NAME &EQUAL &VALUE
&&NAME = &VALUE
*&TYPE &PENROLL &AENROLL &ENROLL
*                                                                       *
*  READ-IN THE REMAINING ITEMS FROM FILE SETUP DSS - STUDENT            *
*    NAMES AND ABBREVIATIONS, TYPE OF RELATIVE AND ABSOLUTE GOALS.      *
*                                                                       *
&POINTER = &NSET + 1
&NACMG = &N1ABG + &N2ABG + &N4ABG
```

```
&NSCAT2 = &MULT OF &NSCAT 2
&NREAD = &NRCMG + &NACMG + &NSCAT2
EXECIO &NREAD DISKR SETUP DSS A &POINTER (FINIS
&LOOP -LOOP2 &NSCAT2
&READ VARS &NAME &VALUE
&&NAME = &VALUE
-LOOP2
&R1 = 0
&LOOP -RD1 &NRCMG
&R1 = &R1 + 1
&NM = &CONCAT OF RELG &R1
&NM1 = &CONCAT OF RG &R1
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&READ VARS &NM &&NM1A &&NM2A
*&TYPE VARS &NM &&NM1A &&NM2A
-RD1
&A1 = 0
&A2 = 0
&A4 = 0
&LOOP -RD1A &N1ABG
&A1 = &A1 + 1
&NM = &CONCAT OF ABS1 &A1
&NM1 = &CONCAT OF AG1 &A1
&READ VARS &NM &&NM1
*&TYPE VARS &NM &&NM1
-RD1A
&LOOP -RD2 &N2ABG
&A2 = &A2 + 1
&NM = &CONCAT OF ABS2 &A2
&NM1 = &CONCAT OF AG2 &A2
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&READ VARS &NM &&NM1A &&NM2A
*&TYPE VARS &NM &&NM1A &&NM2A
-RD2
&LOOP -RD4 &N4ABG
&A4 = &A4 + 1
&NM = &CONCAT OF ABS4 &A4
&NM1 = &CONCAT OF AG4 &A4
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&NM3A = &CONCAT OF &NM1 3
&NM4A = &CONCAT OF &NM1 4
&READ VARS &NM &&NM1A &&NM2A &&NM3A &&NM4A
*&TYPE VARS &NM &&NM1A &&NM2A &&NM3A &&NM4A
-RD4
&NCATNID = &MULT OF &NSCAT &NINDG
&SUMROW = &NAGRG + &NCATNID + &NACMG + &NRCMG
&NROWS = &NOVRG + &MULT OF &SUMROW &NYEAR
&NGOAL = &NOVRG + &SUMROW
* &LASTYEAR = LATEST POSSIBLE YEAR IN THE PLANNING HORIZON
&LASTYEAR = &STARTYEAR + &ISPAN - 1
*********************************************************************************
*                                                                               *
```

```
* PROVIDE THE OPTION TO 1) GO DIRECTLY TO THE PRIORITY STRUCTURE,        *
*  2) GO DIRECTLY TO THE GOAL TARGET VALUES, OR 3) STEP THROUGH THE      *
*  PROCESS AND HAVE THE OPPORTUNITY TO VIEW AND CHANGE THE BASIC DATA    *
*                                                                        *
**************************************************************************
&BEGTYPE -END1

 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
 $                                                                       $
 $     VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY               $
 $ DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS           $
 $                                                                       $
 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

-END1
&BEGTYPE 11


 .. IF YOU WANT TO GO DIRECTLY TO THE PRIORITY STRUCTURE AND NOT CHANGE
    ANY OF THE MODEL'S GOAL TARGETS OR BASIC DATA, >>> ENTER PRIORITY

 .. IF YOU WANT TO GO DIRECTLY TO THE GOAL TARGET VALUES AND NOT CHANGE
    ANY OF THE MODEL'S BASIC DATA, >>> ENTER TARGET

 .. IF YOU WANT TO STEP THROUGH THE PROCESS AND HAVE THE OPPORTUNITY TO
    VIEW OR CHANGE THE MODEL'S BASIC DATA, >>> ENTER STEP

-PS &READ VARS &RESPONSE
&IF &RESPONSE EQ STEP &GOTO -NYEAR1
&IF &RESPONSE EQ PRIORITY &GOTO -PREPRIOR
&IF &RESPONSE EQ TARGET &GOTO -SETUP
&IF &RESPONSE EQ S &GOTO -NYEAR1
&IF &RESPONSE EQ P &GOTO -PREPRIOR
&IF &RESPONSE EQ T &GOTO -SETUP
&TYPE
&TYPE >>> ENTER PRIORITY, TARGET, OR STEP
&TYPE
&GOTO -PS
-NYEAR1
&PHIND = 0
**************************************************************************
*                                                                        *
* DISPLAY PRESENT NUMBER OF YEARS IN PLANNING HORIZON, CHANGE IF DESIRED*
*                                                                        *
**************************************************************************
&BEGTYPE 9


            ****************************************
            *                                      *
 ********** DEFINITION OF THE  PLANNING HORIZON  ******************
            *                                      *
            ****************************************
```

```
&TYPE    .. THE NUMBER OF YEARS IN THE PLANNING HORIZON IS &NYEAR (NYEAR)
&TYPE    ..... (MAXIMUM NUMBER OF YEARS POSSIBLE IS &MAXNYEAR )
&TYPE
&TYPE    .. THE PLANNING HORIZON STARTS IN YEAR &SYEAR (SYEAR)
&TYPE
&TYPE    .. THE LAST YEAR OF THE PLANNING HORIZON MUST BE &LASTYEAR OR PRIOR
&TYPE
&TYPE  >>>  DO YOU WANT TO CHANGE NYEAR OR SYEAR?, ENTER YES OR NO
&TYPE
-INPH &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -CHECKPH
&IF &RESPONSE EQ YES &GOTO -NYEAR2
&IF &RESPONSE EQ N &GOTO -CHECKPH
&IF &RESPONSE EQ Y &GOTO -NYEAR2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INPH
-NYEAR2
&TYPE >>> ENTER: THE NUMBER OF CHANGES THAT YOU WANT TO MAKE
&READ VARS &RESPONSE
&K = 1
&LOOP -YRCHG1 &RESPONSE
&TYPE
&TYPE **** NAME CHANGE &K
&TYPE >>> ENTER: THE VARIABLE NAME, SPACE, DESIRED VALUE
&READ VARS &INPUT1 &INPUT2
&&INPUT1 = &INPUT2
*
* SET THE INDICATOR SO THAT THE FOLLOWING ARE CHANGED AS A RESULT OF THE
*   CHANGE IN THE PROBLEM STRUCTURE:
*     RE-ENTER COMPARISON GOAL TARGETS
*     RE-ENTER AGGREGATE GOAL TARGETS (SUPPLEMENTAL REVENUE REQUIREMENTS)
*     RE-ENTER PERCENTAGE OF AVERAGE COST GOAL TARGETS
*     RE-ENTER MEASURE OF CHANGE IN TUITION CHARGES OVER TIME
*     RE-ENTER GENERAL INDIVIDUAL GOAL TARGETS
*     RE-CALCULATE STATE REVENUE REQUIREMENT COEFFICIENTS & GOAL TARGETS
*     RE-CALCULATE PERCENTAGE OF AVERAGE COST GOAL COEFFICIENTS
*     RE-CALCULATE STUDENT FEES AND DORM & DINING CHARGES
*     RE-CALCULATE PEER INSTITUTION GOAL TARGETS
*
&PHCHGIND = 1
&K = &K + 1
-YRCHG1
*
* UPDATE THE PROBLEM FRAMEWORK BASED ON THE CHANGES IN THE
*   DEFINITION OF THE PLANNING HORIZON.
*
&PHIND = 1
&GOTO -UPDATE
&GOTO -NYEAR1
-CHECKPH &PHIND = 0
*
* CHECK TO SEE IF THE NUMBER OF YEARS IN THE PLANNING HORIZON DOES NOT
*    EXCEED THE MAXIMUM ALLOWED.
*
```

```
* CHECK TO SEE THAT THE PLANNING HORIZON IS NOT DEFINED BEYOND THE LATEST
*    YEAR ALLOWED.
*
&IF &NYEAR LE &MAXNYEAR &GOTO -CHECK2
&TYPE
&TYPE ERROR !!!
&TYPE     NYEAR = &NYEAR IS GREATER THAN THE MAXIMUM ALLOWED, &MAXNYEAR
&TYPE PLEASE CORRECT.
&TYPE
&PHIND = 1
-CHECK2
&ENDYEAR = &SYEAR + &NYEAR - 1
&IF &ENDYEAR LE &LASTYEAR &GOTO -CHECKOUT
&TYPE
&TYPE ERROR !!!
&TYPE     THE LAST YEAR IN THE PLANNING HORIZON, &ENDYEAR ,IS GREATER THAN
&TYPE        THE MAXIMUM ALLOWED, &LASTYEAR
&TYPE PLEASE CORRECT.
&TYPE
&PHIND = 1
-CHECKOUT
&IF &PHIND NE 0 &GOTO -NYEAR1
-ENROLL
*************************************************************************
*
* DISPLAY ENROLLMENT DATA IN EFFECT, CHANGE IF DESIRED                  *
*                                                                       *
*************************************************************************
&IF &PHCHGIND EQ 0 &GOTO -MSG1
&BEGTYPE 3

THE PLANNING HORIZON DEFINITION HAS CHANGED, YOU MAY WANT TO REVIEW
   THE ENROLLMENT DATA VALUES.
-MSG1 &BEGTYPE 9

                          ********************
                          *                  *
 ************************ ENROLLMENT DATA    ************************
                          *                  *
                          ********************

    >>> DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INENROLL &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -FINANCE
&IF &RESPONSE EQ YES &GOTO -ENROLL2
&IF &RESPONSE EQ N &GOTO -FINANCE
&IF &RESPONSE EQ Y &GOTO -ENROLL2
&TYPE
&TYPE >>> ENTER YES OR NO
&GOTO -INENROLL
-ENROLL2
*
* SET INDICATOR FOR CHANGES THAT MUST BE MADE AS A RESULT OF THE CHANGE
*   IN ENROLLMENT DATA:
```

```
*      RE-CALCULATE STATE REVENUE REQUIREMENT GOAL COEFFICIENTS & TARGETS
*      RE-CALCULATE STUDENT FEES
*      RE-CALCULATE AVERAGE COST PER STUDENT
*      RE-CALCULATE TUITION AND FEE RECOVERY RATE
*      RE-CALCULATE PEER INSTITUTION GOAL TARGETS
*
&ENROLLIND = 1
*
* PASS CONTROL TO CCP: ENROLL EXEC; PASS THE ENROLLMENT FILE NAMES
*  THAT ARE IN EFFECT
*
EXEC ENROLL &NSET2 &PENROLL &AENROLL
&NSET3 = &NSET2 + 1
EXECIO 1 DISKR SETUP DSS A &NSET3 (FINIS
&READ VARS &NAME &EQUAL &PENROLL &AENROLL
&GOTO -MSG1
-FINANCE
*************************************************************************
*                                                                       *
* DISPLAY FINANCIAL DATA, CHANGE IF DESIRED                             *
*  COST INFO, STUDENT FEES, DORM & DINING, CAPITAL PROJECTS             *
*                                                                       *
*************************************************************************
&IF &PHCHGIND EQ 0 &GOTO -MSG2
&TYPE PLANNING HORIZON DEFINITION HAS CHANGED
&TYPE
-MSG2 &IF &ENROLLIND EQ 0 &GOTO -MSG3
&TYPE ENROLLMENT DEFINITION HAS CHANGED
&TYPE
-MSG3 &IF &PHCHGIND EQ 0 &ENROLLIND EQ 0 &GOTO -INFIN
&TYPE YOU MAY WANT TO REVIEW THE FINANCIAL DATA
-INFIN &BEGTYPE 11

                  ************************************
                  *                                  *
                  *          FINANCIAL DATA          *
                  *                                  *
                  *   . BASIC COST INFORMATION       *
*****************   . STUDENT FEE DATA               *********************
                  *   . DORM & DINING DATA           *
                  *   . CAPITAL PROJECTS DATA        *
                  *                                  *
                  ************************************
&BEGTYPE 3

       >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INFIN2 &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -FIN2
&IF &RESPONSE EQ NO &GOTO -COMPARISON
&IF &RESPONSE EQ Y &GOTO -FIN2
&IF &RESPONSE EQ N &GOTO -COMPARISON
&TYPE
&TYPE >>> ENTER YES OR NO
&GOTO -INFIN2
```

```
-FIN2
EXEC FINANCE &SYEAR &DDOPTION
*
* SET INDICATOR FOR CHANGES THAT MUST BE MADE AS A RESULT OF THE CHANGE
*  IN THE FINANCIAL DATA:
*     RE-CALCULATE STATE REVENUE REQUIREMENT GOAL COEFFICIENTS & TARGETS
*     RE-CALCULATE STUDENT FEES
*     RE-CALCULATE AVERAGE COST PER STUDENT
*     RE-CALCULATE TUITION AND FEE RECOVERY RATE
*     RE-CALCULATE PEER INSTITUTION GOAL TARGETS
*
&COSTIND = 1
&GOTO -INFIN
*****************************************************************
*                                                               *
* DISPLAY THE RELATIVE AND ABSOLUTE GOALS THAT ARE IN EFFECT    *
*                                                               *
*****************************************************************
-COMPARISON
&ABSCHGIND = 0
&BEGTYPE 24

               ****************************************
               *                                      *
 ************** RELATIVE AND ABSOLUTE COMPARISON GOALS ***************
               *                                      *
               ****************************************

NOTES:
  1) IF YOU WISH TO ADD OR DELETE ANY RELATIVE OR ABSOLUTE GOAL,
    YOU MUST USE THE CHANGE COMMAND.
     MAKING CHANGES UNDER THE DISPLAY OPTION WILL NOT BE RECOGNIZED
    BY THE SYSTEM.
  2) IF YOU USE THE CHANGE OPTION, YOU MUST:
     A) RE-ENTER ALL OF THE RELATIVE AND ABSOLUTE GOAL TARGET VALUES.
     B) RE-ENTER THE ENTIRE PRIORITY STRUCTURE.
   (YOU WILL PROMPTED FOR THESE AT A LATER TIME IN THE SESSION)

       .. IF YOU WANT TO VIEW THE GOALS IN EFFECT,
          >>> ENTER: DISPLAY
       .. IF YOU WANT TO CHANGE THE BASIC GOALS,
          >>> ENTER: CHANGE
       .. IF NEITHER OF THE ABOVE,
          >>> ENTER: NO

-INRAG &RLAD VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -SETUP
&IF &RESPONSE EQ DISPLAY &GOTO -ABSDISP
&IF &RESPONSE EQ CHANGE &GOTO -ABSCHG
&IF &RESPONSE EQ N &GOTO -SETUP
&IF &RESPONSE EQ D &GOTO -ABSDISP
&IF &RESPONSE EQ C &GOTO -ABSCHG
&TYPE
&TYPE >>> ENTER DISPLAY, CHANGE, OR NO.
&GOTO -INRAG
```

```
-ABSCHG
*
* SET &COMPIND = 1: SINCE THE COMPARISON GOALS ARE BEING CHANGED,
*      THE USER MUST RE-ENTER THE GOAL TARGET VALUES IN COMPARE EXEC.
*         AND
*      THE USER MUST RE-ENTER THE PRIORITY STRUCTURE IN PRIORITY EXEC.
*
* SET &ABSCHGIND = 1: SINCE THE COMPARISON GOALS ARE BEING CHANGED,
*      THESE CHANGES MUST BE REFLECTED IN FILE SETUP DSS.
*
&COMPIND = 1
&ABSCHGIND = 1
&TYPE GENERATING RELATIVE AND ABSOLUTE GOALS
*
* GENERATE A NEW FILE, COMPGEN DSS, CONTAINING ALL POSSIBLE RELATIVE
*    AND ABSOLUTE GOALS.
*
SET CMSTYPE HT
ERASE COMPGEN DSS A
*FORTHX COMPGEN (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK COMPGEN DSS A
LOAD COMPGEN (START NOMAP
SET CMSTYPE RT
XEDIT COMPGEN DSS A
&TYPE PROCESSING RELATIVE AND ABSOLUTE GOALS
*
* READ-IN RELATIVE AND ABSOLUTE GOAL INFORMATION:
*    NUMBER AND TYPE
*
&SPACE = 16
&NUM1 = &NSCAT - 1
&NUM2 = &MULT OF &NSCAT &NUM1
&NUMBA2 = &DIV OF &NUM2 2
&NUMBER = &NSCAT + &NUMBA2 + &NUMBA2 + &SPACE + 10
EXECIO &NUMBER DISKR COMPGEN DSS A (FINIS
&R1 = 0
&READ 5 VARS
&LOOP -REL1 &NUMBA2
&READ VARS &A &B &C
&IF &A NE Y &GOTO -REL1
&R1 = &R1 + 1
&N1 = &CONCAT OF RG &R1
&N1A = &CONCAT OF &N1 1
&&N1A = &B
&N2A = &CONCAT OF &N1 2 .
&&N2A = &C
-REL1
&A1 = 0
&A2 = 0
&A4 = 0
&READ 2 VARS
```

```
&LOOP -ABS1 &NSCAT
&READ VARS &A &B &C
&IF &A NE Y &GOTO -ABS1
&A1 = &A1 + 1
&N1 = &CONCAT OF AG1 &A1
&&N1 = &C
-ABS1
&READ 2 VARS
&LOOP -ABS2 &NUMBA2
&READ VARS &A &B &C &D
&IF &A NE Y &GOTO -ABS2
&A2 = &A2 + 1
&N1 = &CONCAT OF AG2 &A2
&N1A = &CONCAT OF &N1 1
&&N1A = &C
&N2A = &CONCAT OF &N1 2
&&N2A = &D
-ABS2
&READ 7 VARS
&LOOP -ABS4 10
&READ VARS &A &B &C &D &E &F
&IF &A NE Y &GOTO -ABS4
&A4 = &A4 + 1
&N1 = &CONCAT OF AG4 &A4
&N1A = &CONCAT OF &N1 1
&&N1A = &C
&N2A = &CONCAT OF &N1 2
&&N2A = &D
&N3A = &CONCAT OF &N1 3
&&N3A = &E
&N4A = &CONCAT OF &N1 4
&&N4A = &F
-ABS4
&NRCMG = &R1
&N1ABG = &A1
&N2ABG = &A2
&N4ABG = &A4
&NACMG = &A1 + &A2 + &A4
&GOTO -UPDATE
-ABSDISP
XEDIT COMPGEN DSS A
&GOTO -COMPARISON
*****************************************************************************
*                                                                           *
* UPDATE PROBLEM FRAMEWORK BASED ON CHANGES AND STORE IN FILE SETUP DSS *
*                                                                           *
*****************************************************************************
-UPDATE &BEGTYPE 3

UPDATING PROBLEM FRAMEWORK BASED ON CHANGES, MAY TAKE A MOMENT

ERASE SETUP DSS A
&NCATNID = &MULT OF &NSCAT &NINDG
&SUMROW = &NACRG + &NCATNID + &NACMG + &NRCMG
&NROWS = &NOVRG + &MULT OF &SUMROW &NYEAR
```

```
&NGOAL = &NOVRG + &SUMROW
&STACK BYEAR = &BYEAR
&STACK SYEAR = &SYEAR
&STACK NYEAR = &NYEAR
&STACK NSCAT = &NSCAT
&STACK NAGRG = &NAGRG
&STACK NINDG = &NINDG
&STACK NRCMG = &NRCMG
&STACK N1ABG = &N1ABG
&STACK N2ABG = &N2ABG
&STACK N4ABG = &N4ABG
&STACK NOVRG = &NOVRG
&STACK ENROLL = &PENROLL &AENROLL
&STACK ISPAN = &ISPAN
&K = 1
&LOOP -LOOP4 &NSCAT
&NAME = &CONCAT OF NAME &K
&VALUE = &&NAME
&STACK &NAME &VALUE
&K = &K + 1
-LOOP4
&K = 1
&LOOP -LOOP5 &NSCAT
&ABBREV = &CONCAT OF ABBR &K
&VALUE2 = &&ABBREV
&STACK &ABBREV &VALUE2
&K = &K + 1
-LOOP5
&R1 = 0
&LOOP -REL2 &NRCMG
&R1 = &R1 + 1
&NM = &CONCAT OF RELG &R1
&NM1 = &CONCAT OF RG &R1
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&STACK &NM &&NM1A &&NM2A
-REL2
&A1 = 0
&A2 = 0
&A4 = 0
&LOOP -ABS1 &N1ABG
&A1 = &A1 + 1
&NM = &CONCAT OF ABS1 &A1
&NM1 = &CONCAT OF AG1 &A1
&STACK &NM &&NM1
-ABS1
&LOOP -ABS2 &N2ABG
&A2 = &A2 + 1
&NM = &CONCAT OF ABS2 &A2
&NM1 = &CONCAT OF AG2 &A2
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&STACK &NM &&NM1A &&NM2A
-ABS2
&LOOP -ABS4 &N4ABG
```

```
&A4 = &A4 + 1
&NM = &CONCAT OF ABS4 &A4
&NM1 = &CONCAT OF AG4 &A4
&NM1A = &CONCAT OF &NM1 1
&NM2A = &CONCAT OF &NM1 2
&NM3A = &CONCAT OF &NM1 3
&NM4A = &CONCAT OF &NM1 4
&STACK &NM &&NM1A &&NM2A &&NM3A &&NM4A
-ABS4
&NACMG = &N1ABG + &N2ABG + &N4ABG
&NPRIOR = &NSET + &NRCMG + &NACMG + &MULT OF &NSCAT 2
EXECIO &NPRIOR DISKW SETUP DSS A 1 F 80 (FINIS
&IF &PHIND = 1 &GOTO -NYEAR1
&IF &ABSCHGIND = 1 &GOTO -COMPARISON
-SETUP
*****************************************************************************
*                                                                           *
* DEFINE THE PROBLEM FRAMEWORK IN IFPS FORMAT                               *
*    AND                                                                    *
* CREATE AN MPS FORMAT FILE FOR THE OPTIMIZATION ROUTINE                    *
*                                                                           *
*****************************************************************************
EXEC SETUP &FILETYPE
-COMPARE
*****************************************************************************
*                                                                           *
* ESTABLISH RELATIVE AND ABSOLUTE COMPARISON GOAL TARGET VALUES             *
*                                                                           *
*****************************************************************************
EXEC COMPARE &PHCHGIND &COMPIND
-AGGREGATE
*****************************************************************************
*                                                                           *
* ESTABLISH AGGREGATE GOALS                                                 *
*                                                                           *
*****************************************************************************
&CALC = 0
* &CALC IS AN INDICATOR THAT EITHER THE PROBLEM STRUCTURE HAS CHANGED
*    (PHCHGIND), THE ENROLLMENT DATA HAS CHANGED (ENROLLIND), OR THE
*    COST DATA HAS CHANGED (COSTIND) AND IT IS NECESSARY TO RECALCULATE
*    THE STATE REVENUE REQUIREMENT COEFFICIENTS AND GOAL TARGETS, THE
*    PERCENTAGE-OF-COST GOAL TARGETS, STUDENT FEES, AND DORM & DINING
&IF &PHCHGIND EQ 1 &CALC = 1
&IF &COSTIND EQ 1 &CALC = 1
&IF &ENROLLIND EQ 1 &CALC = 1
EXEC AGGR &PHCHGIND &CALC &PENROLL &AENROLL &COMPIND &DDOPTION
*****************************************************************************
*                                                                           *
* PERCENTAGE-OF-AVERAGE-COST GOALS                                          *
*                                                                           *
*****************************************************************************
-PERCOST
EXEC PERCOST &PHCHGIND
*****************************************************************************
*                                                                           *
```

```
* CHANGE-IN-TUITION-CHARGES-OVER-TIME GOALS                              *
*                                                                       *
*************************************************************************
-CHGTIME
EXEC CHGTIME &PHCHGIND
-PEER
*************************************************************************
*                                                                       *
* ESTABLISH PEER INSTITUTION GOALS                                      *
*                                                                       *
*************************************************************************
&IF &PHCHGIND EQ 0 &GOTO -INSTPR
&BEGTYPE 5

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS BEEN CHANGED, THE
        PEER INSTITUTION GOALS MUST BE RE-ESTABLISHED.

          WHEN READY TO RE-ENTER: PRESS 'RETURN'
&READ VARS
&GOTO -PEER2
-INSTPR &BEGTYPE 9

                ************************************
                *                                  *
**************** ESTABLISH PEER INSTITUTION GOALS *********************
                *                                  *
                ************************************

      >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INPEER &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -PEER2
&IF &RESPONSE EQ NO &GOTO -PEERCHK
&IF &RESPONSE EQ Y &GOTO -PEER2
&IF &RESPONSE EQ N &GOTO -PEERCHK
&TYPE
&TYPE >>> ENTER YES OR NO
&GOTO -INPEER
-PEER2
EXEC PEERINST &PHCHGIND &NSCAT
&GOTO -PEERCALC
-PEERCHK
&IF &ENROLLIND = 1 &GOTO -PEERCALC
&IF &COSTIND = 1 &GOTO -PEERCALC
&GOTO -GENIND
-PEERCALC
*
* CALCULATE PEER INSTITUTION GOAL TARGETS:
*   PEER INSTITUTION VALUE - FEES - DORM & DINING
*
* SETUP IN MPS FORMAT
*
EXEC PEER
*************************************************************************
*                                                                       *
```

```
* GENERAL INDIVIDUAL GOALS                                                    *
*                                                                            *
******************************************************************************
-GENIND
EXEC GENIND &PHCHGIND
&GOTO -PRIORITY
-PREPRIOR
* SET AN INDICATOR TO GO DIRECTLY TO THE OPTIMIZATION AND SKIP SETTING
*    UP THE FILE IN MPS FORMAT
&SKIPMPS = 1
-PRIORITY
******************************************************************************
*                                                                            *
* SET UP PRIORITY STRUCTURE                                                  *
*                                                                            *
******************************************************************************
&IF &COMPIND EQ 0 &GOTO -INST2
&BEGTYPE 5

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS BEEN CHANGED, YOU MUST
        RE-ENTER THE PRIORITY STRUCTURE.

          WHEN READY TO RE-ENTER: PRESS 'RETURN'
&GOTO -PRIOR2
-INST2 &BEGTYPE 9

                        **********************
                        *                    *
 ********************** PRIORITY STRUCTURE ***************************
                        *                    *
                        **********************

     >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INPRIOR &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -PRIOR2
&IF &RESPONSE EQ NO &GOTO -OPTIMIZE
&IF &RESPONSE EQ Y &GOTO -PRIOR2
&IF &RESPONSE EQ N &GOTO -OPTIMIZE
&TYPE
&TYPE >>> ENTER YES OR NO
&GOTO -INPRIOR
-PRIOR2
* NREAD IS THE NUMBER OF LINES TO BE READ FROM FILE PRIOR IFPS.
*   IT IS PASSED TO EXEC PRIORITY AS PARAMETER 2.
*    NGOAL IS THE NUMBER OF GOALS CONSIDERED IN THE MODEL.
*    NSPACE IS THE NUMBER OF BLANK LINES IN FILE PRIOR IFPS.
*    NINDG IS THE NUMBER OF INDIVIDUAL GOAL TITLE LINES.
*    EACH 4-CATEGORY ABSOLUTE GOAL REQUIRES 2 LINES.
*    THERE IS A TITLE LINE FOR EACH OF THE 3 TYPES OF ABSOLUTE GOALS AND
*      THE RELATIVE GOALS.
*    THERE ARE 11 TOP HEADING LINES
&NSPACE = &NAGRG + &NINDG + &NOVRG + 4 + 4
&NREAD = &NGOAL + &NSPACE + &NINDG + &N4ABG + 11
EXEC PRIORITY &COMPIND &NREAD &NPRIOR
```

```
-OPTIMIZE
*************************************************************************
*                                                                       *
* SET UP THE PROBLEM IN MPS FORMAT, SOLVE USING LINDO                   *
*                                                                       *
*************************************************************************
&BEGTYPE 9

                    ***********************************
                    *                                 *
 ******************* OPTIMIZATION - PROBLEM SOLUTION *******************
                    *                                 *
                    ***********************************

            >>>  ARE YOU READY TO SOLVE ?, ENTER YES OR NO.

-INOPT &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -MPS
&IF &RESPONSE EQ NO &GOTO -REPORTS
&IF &RESPONSE EQ Y &GOTO -MPS
&IF &RESPONSE EQ N &GOTO -REPORTS
&TYPE
&TYPE ENTER YES OR NO
&GOTO -INOPT
-MPS
&IF &SKIPMPS EQ 1 &GOTO -SOLVE
*
* SET UP THE PROBLEM FORMULATION IN A FILE TO BE READ BY LINDO
*
EXEC MPS &FILETYPE
*
* VIEW THE FORMULATION IN EQUATION FORM
*
&TYPE
&TYPE  DO YOU WANT TO VIEW THE PROBLEM FORMULATION IN EQUATION FORM ?
&TYPE
&TYPE     NOTE: YOU MUST ENTER YES IF YOU WANT TO PRINT A HARDCOPY OF
&TYPE           THE PROBLEM FORMULATION.
&TYPE  >>>  ENTER YES OR NO.
-INVIEW &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -VIEW
&IF &RESPONSE EQ NO &GOTO -SOLVE
&IF &RESPONSE EQ Y &GOTO -VIEW
&IF &RESPONSE EQ N &GOTO -SOLVE
&TYPE
&TYPE ENTER YES OR NO
&GOTO -INVIEW
-VIEW
EXEC FORMAT &UNO
-SOLVE
*
* SOLVE THE PROBLEM INTERACTIVELY USING LINDO
*
EXEC GP &UNO
ERASE USER LISTING
```

```
*************************************************************************
*                                                                       *
* VIEW THE TUITION VALUES (SOLUTION)                                    *
*                                                                       *
*************************************************************************
XEDIT TUIT IFPS
*************************************************************************
*                                                                       *
* VIEW THE LEVEL OF GOAL ACHIEVEMENT ATTAINED                           *
*                                                                       *
*************************************************************************
&BEGTYPE 10

                    ********************
                    *                  *
 ******************* GOAL ACHIEVEMENT ********************************
                    *                  *
                    ********************

  >>>  DO YOU WANT TO VIEW THE LEVEL OF GOAL ACHIEVEMENT ATTAINED
         BY THE PRESENT SOLUTION?,  ENTER YES OR NO.

-INGA &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -GOALACH
&IF &RESPONSE EQ NO &GOTO -REPORTS
&IF &RESPONSE EQ Y &GOTO -GOALACH
&IF &RESPONSE EQ N &GOTO -REPORTS
&TYPE
&TYPE ENTER YES OR NO
&GOTO -INGA
-GOALACH
XEDIT DEVVAR IFPS
*************************************************************************
*                                                                       *
* SELECT WHICH MANAGEMENT REPORTS TO PRINT                              *
*                                                                       *
*************************************************************************
-REPORTS
EXEC REPORTS &PENROLL &AENROLL &NREPORTS &DDOPTION
*************************************************************************
*                                                                       *
* CONTINUE OR STOP THE ANALYSIS                                         *
*                                                                       *
*************************************************************************
&BEGTYPE 10

                    ****************************
                    *                          *
 ******************* ?? ANOTHER ANALYSIS ??  ***************************
                    *                          *
                    ****************************

    >>>  DO YOU WANT TO CONSIDER AN ALTERNATIVE OR A VARIATION
           ON THE PROBLEM ?,  ENTER YES OR NO.
```

```
-INANALYSIS &READ VARS &RESPONSE
&IF &RESPONSE EQ YES &GOTO -START
&IF &RESPONSE EQ NO &GOTO -BYE
&IF &RESPONSE EQ Y &GOTO -START
&IF &RESPONSE EQ N &GOTO -BYE
&TYPE
&TYPE ENTER YES OR NO
&GOTO -INANALYSIS
-BYE &BEGTYPE 10

$$$
```

Appendix B

PROGRAM LISTINGS FOR THE COMPONENT CONTROL PROGRAMS (CCP)

B.1 Component Control Program AGGR
B.2 Component Control Program CHGTIME
B.3 Component Control Program COMPARE
B.4 Component Control Program ENROLL
B.5 Component Control Program FINANCE
B.6 Component Control Program FORMAT
B.7 Component Control Program GENIND
B.8 Component Control Program GP
B.9 Component Control Program MPS
B.10 Component Control Program PEER
B.11 Component Control Program PEERINST
B.12 Component Control Program PERCOST
B.13 Component Control Program PRIORITY
B.14 Component Control Program REPORTS
B.15 Component Control Program SETUP

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE AGGREGATE GOALS       *
*                                                                       *
*************************************************************************
*
* ALL OF THE FOLLOWING PARAMETERS ARE PASSED FROM THE ECP, DSS.EXEC
* &1 = INDICATOR AS TO WHETHER AGGREGATE GOALS NEED TO BE RESTABLISHED
*      BECAUSE THE PLANNING HORIZON HAS CHANGED.
* &2 = INDICATOR AS TO WHETHER CERTAIN CALCULATIONS NEED TO BE PERFORMED
*      AS THE RESULT OF CHANGES IN THE PLANNING HORIZON, ENROLLMENT
*      DATA, OR FINANCIAL DATA.
* &3 = "PROJECTED" ENROLLMENT DATA FILE NAME
* &4 = "ACTUAL" ENROLLMENT DATA FILE NAME
* &5 = INDICATOR AS TO WHETHER THE COMPARISON GOALS HAVE CHANGED
* &6 = DORM & DINING CALCULATION OPTION NUMBER
*
&PHCHGIND = &1
&CALC = &2
&PENROLL = &3
&AENROLL = &4
&COMPIND = &5
&DDOPTION = &6
&IF &CALC EQ 0 &GOTO -SUPREV
*
* CALCULATE STATE REVENUE REQUIREMENT GOAL COEFFICIENTS (AGGREGATE GOAL)
*
* GENERATE ENROLLMENT BASED MANAGEMENT REPORTS
*
* CREATE 4 FILES:
*    1. A WORKING FILE OF ENROLLMENT DATA FOR IFPS MODELS (ENRLWORK IFPS)
*    2. A FILE OF STATE REV. RQMT. COEFFICIENTS (STATEREV IFPS)
*    3. A MANAGEMENT REPORT OF ENROLLMENT DATA (ENROLOUT IFPS)
*    4. A MANAGEMENT REPORT OF CREDIT HOUR DATA (CREDOUT IFPS)
*
-GENCOEF
&BEGTYPE 3

GENERATING COEFFICIENTS FOR STATE REVENUE REQUIREMENT GOALS USING IFPS

EXEC IFPSDSS A B C DSS ENRLCRED &PENROLL &AENROLL
*
* CALCULATE STATE REVENUE REQUIREMENT GOAL TARGETS (AGGREGATE GOAL)
*
* CALCULATE PERCENTAGE OF AVERAGE TOTAL COST GOAL COEFFICIENTS
*
* CREATE 3 FILES:
*    1. STATE REVENUE REQUIREMENT GOAL TARGETS APPENDED TO (STATEREV IFPS)
*    2. A FILE OF PERCENT OF COST GOAL COEFFICIENTS (PERCOST IFPS)
*    3. A MANAGEMENT REPORT OF COST DATA (COSTINFO IFPS)
*
```

```
-GENCOST
&BEGTYPE 4

GENERATING THE STATE REVENUE REQUIREMENT GOAL TARGETS AND THE
  COEFFICIENTS FOR THE PERCENTAGE OF AVERAGE TOTAL COST GOALS, USING IFPS

EXEC IFPSDSS A B C DSS COSTCOM
&BEGTYPE 3

FINISHED WITH IFPS FOR NOW

* CONVERT FILE STATEREV IFPS FROM V TO F FORMAT
&STACK SET RECF F
&STACK FILE
XEDIT STATEREV IFPS
* CONVERT FILE PERCOST IFPS FROM V TO F FORMAT
&STACK SET RECF F
&STACK FILE
XEDIT PERCOST IFPS
-FEECALC
*************************************************************************
*                                                                       *
* CALCULATE ANNUAL STUDENT FEES AND DORM & DINING CHARGES USING IFPS    *
*                                                                       *
*************************************************************************
&BEGTYPE 3

CALCULATING ANNUAL STUDENT FEES AND DORM & DINING CHARGES USING IFPS

&DDPARM1 = &CONCAT OF ' &DDOPTION
&DDPARM = &CONCAT OF &DDPARM1 '
EXEC IFPSDSS A B C DSS DDFEECOM &DDPARM
&BEGTYPE 3

FINISHED WITH IFPS FOR NOW

*
* CONVERT FILE DDFEEMPS IFPS FROM V TO F FORMAT
*
&STACK SET RECF F
&STACK FILE
XEDIT DDFEEMPS IFPS
*
* SUPPLEMENTAL REVENUE REQUIREMENT GOALS
*
-SUPREV
&IF &PHCHGIND EQ 0 &GOTO -INSUPREV
&BEGTYPE 6

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS BEEN CHANGED, THE
          SUPPLEMENTAL REVENUE REQUIREMENT GOAL TARGET VALUES
          MUST BE RE-ESTABLISHED.

            WHEN READY TO RE-ENTER: PRESS 'RETURN'
&READ VARS
```

```
&GOTO -SUPREVGEN
-INSUPREV &BEGTYPE 13

         ********************************************************
         *                                                      *
********* SUPPLEMENTAL REVENUE REQUIREMENT GOAL TARGET VALUES ********
         *                                                      *
         ********************************************************

        .. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
           >>> ENTER: DISPLAY
        .. IF YOU WANT TO REGENERATE THE GOAL VALUES,
           >>> ENTER: GENERATE
        .. IF NEITHER OF THE ABOVE,
           >>> ENTER: NO
-INAGGR &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -AGGRCHECK
&IF &RESPONSE EQ DISPLAY &GOTO -SUPREVDISP
&IF &RESPONSE EQ GENERATE &GOTO -SUPREVGEN
&IF &RESPONSE EQ N &GOTO -AGGRCHECK
&IF &RESPONSE EQ D &GOTO -SUPREVDISP
&IF &RESPONSE EQ G &GOTO -SUPREVGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INAGGR
-SUPREVGEN
SET CMSTYPE HT
ERASE SUPLMREV IFPS A
*FORTHX SUPLMREV (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK SUPLMREV IFPS A
LOAD SUPLMREV (START NOMAP
SET CMSTYPE RT
-SUPREVDISP
XEDIT SUPLMREV IFPS A
&CALC = 1
&GOTO -INSUPREV
-AGGRCHECK
&IF &COMPIND EQ 1 &GOTO -AGGRMPS
&IF &CALC EQ 0 &GOTO -OUT
*
* CONVERT AGGREGATE GOALS TO MPS FORMAT
*
-AGGRMPS
&TYPE SETTING UP AGGREGATE GOALS IN MPS FORMAT
SET CMSTYPE HT
ERASE AGRGRHS DSS A
ERASE AGRGX DSS A
ERASE AGRGNP DSS A
*FORTHX AGRGMPS (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
```

```
FILEDEF 03 DISK SETUP DSS A
FILEDEF 02 DISK STATEREV IFPS A
FILEDEF 04 DISK SUPLMREV IFPS A
#OUTPUT FILES
FILEDEF 05 DISK AGRGRHS DSS A
FILEDEF 06 DISK AGRGX DSS A
FILEDEF 07 DISK AGRGNP DSS A
LOAD AGRGMPS (START NOMAP
SET CMSTYPE RT
-OUT &EXIT
```

```
&TRACE
*****************************************************************************
*                                                                           *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE CHANGE-IN-TUITION-        *
*    CHARGES-OVER-TIME GOALS                                                *
*                                                                           *
*****************************************************************************
* &1 = INDICATOR THAT THE PROBLEM STRUCTURE HAS CHANGED (PASSED FROM
*    DSS EXEC
&PHCHGIND = &1
&IF &PHCHGIND EQ 0 &GOTO -INCHGTIME
&BEGTYPE 6

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS CHANGED, THE
          CHANGE-IN-TUITION-CHARGES-OVER-TIME GOAL TARGET VALUES
          MUST BE RE-ESTABLISHED.

            WHEN READY TO RE-ENTER: PRESS 'RETURN'
&READ VARS
&GOTO -CHGTIMEGEN
-INCHGTIME &BEGTYPE 13

        ***********************************************************
        *                                                         *
******* CHANGE-IN-TUITION-CHARGES-OVER-TIME GOAL TARGET VALUES *******
        *                                                         *
        ***********************************************************

        .. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
           >>> ENTER: DISPLAY
        .. IF YOU WANT TO REGENERATE THE GOAL VALUES,
           >>> ENTER: GENERATE
        .. IF NEITHER OF THE ABOVE,
           >>> ENTER: NO
-INCTIME &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ DISPLAY &GOTO -CHGTIMEDISP
&IF &RESPONSE EQ GENERATE &GOTO -CHGTIMEGEN
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ D &GOTO -CHGTIMEDISP
&IF &RESPONSE EQ G &GOTO -CHGTIMEGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INCTIME
-CHGTIMEGEN
&TYPE GENERATING FILE FOR CHANGE-IN-TUITION-CHARGES-OVER-TIME GOAL VALUES
SET CMSTYPE HT
ERASE CHGTIME TEMPS A
*FORTHX CTIMEGEN (NOPRINT
GLOBAL TXTLIB FORTVSTB
*INPUT FILES:
FILEDEF 03 DISK SETUP DSS A
```

```
*OUTPUT FILES
FILEDEF 06 DISK CHGTIME IFPS A
LOAD CTIMEGEN (START NOMAP
SET CMSTYPE RT
-CHGTIMEDISP
*
* VIEW THE ECONOMIC INDICATORS THAT ARE HELPFUL IN SETTING THE GOAL
*     TARGET VALUES
*
XEDIT INDEXCHG IFPS A
*
* SET THE GOAL TARGET VALUES
*
&STACK SET RECF F
XEDIT CHGTIME IFPS A
&TYPE SETTING UP CHANGE-IN-TUITION-OVER-TIME GOALS IN MPS FORMAT
-MPS
*
* CONVERT TO MPS FORMAT
*
SET CMSTYPE HT
ERASE CTIMERHS DSS A
ERASE CTIMEX DSS A
ERASE CTIMENP DSS A
*FORTHX CHGTIME (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 04 DISK CHGTIME IFPS A
FILEDEF 05 DISK TUITHIST DSS A
*OUTPUT FILES
FILEDEF 06 DISK CTIMERHS DSS A
FILEDEF 07 DISK CTIMEX DSS A
FILEDEF 08 DISK CTIMENP DSS A
FILEDEF 09 DISK BASEYR DSS A
FILEDEF 10 DISK ERROR DSS A
LOAD CHGTIME (START NOMAP
SET CMSTYPE RT
STATE ERROR DSS A
&IF &RC NE 0 &GOTO -OUT
EXECIO 1 DISKR ERROR DSS A (FINIS
&READ VAR &BASEYEAR
ERASE ERROR DSS A
&BEGTYPE 2

ERROR  -  THE TUITION CHARGE IS 0.0 FOR AT LEAST ONE STUDENT CATEGORY
&TYPE        IN THE BASE YEAR, &BASEYEAR
&BEGTYPE 2

>>> WHEN YOU ARE READY TO ENTER THE CORRECT VALUE(S), PRESS 'RETURN'
&READ VARS
XEDIT TUITHIST DSS A
&GOTO -MPS
-OUT &EXIT
```

```
&TRACE
*
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE RELATIVE AND          *
*     ABSOLUTE GOAL TARGET VALUES                                       *
*                                                                       *
*************************************************************************
*
* PARAMETERS PASSED FROM THE FCP, DSS EXEC
* &1 = INDICATOR THAT THE PROBLEM STRUCTURE HAS CHANGED
* &2 = INDICATOR THAT THE RELATIVE OR ABSOLUTE GOALS HAVE BEEN CHANGED
&PHCHGIND = &1
&COMPIND = &2
&IND = 0
&IF &PHCHGIND EQ 0 &GOTO -CHK
&IND = 1
-CHK &IF &COMPIND EQ 0 &GOTO -CHK2
&IND = 1
-CHK2 &IF &IND EQ 0 &GOTO -INST1
&BEGTYPE 5

NOTE       SINCE THE STRUCTURE OF THE PROBLEM HAS CHANGED, YOU MUST
        RE-ENTER THE RELATIVE AND ABSOLUTE GOAL TARGET VALUES

           WHEN READY TO RE-ENTER THE GOALS: PRESS 'RETURN'
&READ VARS
&GOTO -SET2
-INST1 &BEGTYPE 13

               ***************************************
               *                                     *
***************  RELATIVE AND ABSOLUTE GOAL TARGET VALUES  ***************
               *                                     *
               ***************************************

  .. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
     >>> ENTER: DISPLAY
  .. IF YOU WANT TO REGENERATE THE GOAL VALUES,
     >>> ENTER: GENERATE
  .. IF NEITHER OF THE ABOVE,
     >>> ENTER: NO
-INCOMP &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ DISPLAY &GOTO -SET3
&IF &RESPONSE EQ GENERATE &GOTO -SET2
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ D &GOTO -SET3
&IF &RESPONSE EQ G &GOTO -SET2
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INCOMP
```

```
-SET2
&TYPE REGENERATING RELATIVE AND ABSOLUTE GOAL VALUES
SET CMSTYPE HT
ERASE COMPSET IFPS A
*FORTHX COMPSET (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK COMPSET IFPS A
LOAD COMPSET (START NOMAP
SET CMSTYPE RT
-SET3
XEDIT COMPSET IFPS A
-MPS
*
* TRANSFER ALL RELATIVE & ABSOLUTE GOALS TO MPS FORMAT, COLUMNS AND RHS
*
&TYPE TRANSFERING ALL RELATIVE AND ABSOLUTE GOALS TO MPS FORMAT
SET CMSTYPE HT
ERASE COMPRHS DSS A
ERASE COMPX DSS A
ERASE COMPNP DSS A
*FORTHX COMPMPS (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 02 DISK COMPSET IFPS A
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK COMPRHS DSS A
FILEDEF 05 DISK COMPX DSS A
FILEDEF 06 DISK COMPNP DSS A
LOAD COMPMPS (START NOMAP
SET CMSTYPE RT
&GOTO -INST1
-OUT &EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM TO HANDLE ENROLLMENT DATA FILE DEFINITION*
*                                                                       *
*************************************************************************
*
*************************************************************************
* PARAMETERS PASSED FROM THE FCP, DSS EXEC                              *
* &1 = &NSET2 - NUMBER OF ITEMS TO "SKIP OVER" IN FILE SETUP DSS        *
*    BEFORE THE ENROLLMENT FILES ARE DEFINED.                           *
* &2 = &PENROLL - NAME OF "PROJECTED" DATA FILE                         *
* &3 = &AENROLL - NAME OF "ACTUAL" DATA FILE                            *
*************************************************************************
&NSET2 = &1
&PENROLL = &2
&AENROLL = &3
&OPTCHGIND = 0
-ENROLLDEF
*
* DETERMINE WHICH OPTION IS IN EFFECT
*
&OPT3COUNT = 0
&INDP = 0
&INDA = 0
&OPT = 3
&TAGP = &PIECE OF &PENROLL 1 1
&IF &TAGP EQ C &INDP = -1
&TAGA = &PIECE OF &AENROLL 1 1
&IF &TAGA EQ C &INDA = -2
&OPTION = &OPT + &INDP + &INDA
&IF &OPTION NE 0 &GOTO -ENROLLOPT
&TYPE ERROR IN DEFINITION OF ENROLLMENT FILES, RESPECIFY
-ENROLLOPT &BEGTYPE 13

  IT IS POSSIBLE TO USE TWO DIFFERENT ENROLLMENT DATA FILES IN
     IN THE MODEL:
  1) "PROJECTED" ENROLLMENT IS USED IN THE RECOVERY RATE CALCULATIONS
  2) "ACTUAL" ENROLLMENT IS USED TO CALCULATE THE EXPECTED NUMBER
       OF CREDIT HOURS BY STUDENT CATEGORY AND STUDENT FEES.

THERE ARE THREE OPTIONS FOR SPECIFYING THE ENROLLMENT DATA FILES:

1) ACTUAL DATA SAME AS PROJECTED (PROJECTED IS THE BASIS)
2) PROJECTED DATA SAME AS ACTUAL (ACTUAL IS THE BASIS)
3) BOTH PROJECTED AND ACTUAL DIFFERENT

&TYPE PRESENTLY OPTION &OPTION IS IN EFFECT, DO YOU WANT TO CHANGE?,
&TYPE >>> ENTER YES OR NO
&TYPE
-INOPT1 &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -INOPT5
```

```
&IF &RESPONSE EQ YES &GOTO -INOPT2
&IF &RESPONSE EQ N &GOTO -INOPT5
&IF &RESPONSE EQ Y &GOTO -INOPT2
&TYPE
&TYPE >>> ENTER YES OR NO
&GOTO -INOPT1
-INOPT2
&OPTCHGIND = 1
&TYPE
&TYPE SPECIFY THE OPTION THAT YOU WANT TO USE
&TYPE
-INOPT3 &READ VARS &RESPONSE
&IF &RESPONSE EQ 1 &GOTO -ENROLLOPT2
&IF &RESPONSE EQ 2 &GOTO -ENROLLOPT2
&IF &RESPONSE EQ 3 &GOTO -ENROLLOPT2
&TYPE
&TYPE >>> ENTER 1 2 OR 3
&GOTO -INOPT3
-ENROLLOPT2
&OPTION = &RESPONSE
-INOPT5
&IF &OPTION = 3 &GOTO -OPT3
&IF &OPTION = 2 &GOTO -OPT2
-OPT1
&AENROLL = CAENROLL
&ENROLLNAME = PROJECTED
&TAG = P
&GOTO -ENROLL2
-OPT2
&PENROLL = CPENROLL
&ENROLLNAME = ACTUAL
&TAG = A
&GOTO -ENROLL2
-OPT3
&OPT3COUNT = &OPT3COUNT + 1
&IF &OPT3COUNT EQ 2 &GOTO -OPT32
&ENROLLNAME = PROJECTED
&TAG = P
&IF &PENROLL NE CPENROLL &GOTO -ENROLL2
&PENROLL = PENROLL1
&GOTO -ENROLL2
-OPT32
&ENROLLNAME = ACTUAL
&TAG = A
&IF &AENROLL NE CAENROLL &GOTO -ENROLL2
&AENROLL = AENROLL1
&GOTO -ENROLL2
-ENROLL2
&TYPE
&TYPE ********** &ENROLLNAME ENROLLMENT DATA  **********
&BEGTYPE 10

  .. IF YOU WANT TO VIEW OR CHANGE THE DATA IN EFFECT,
     >>> ENTER: DISPLAY
  .. IF YOU WANT TO SELECT ANOTHER ENROLLMENT DATA FILE,
```

```
      >>> ENTER: SELECT
   .. IF YOU WANT TO CREATE A NEW ENROLLMENT DATA FILE,
      >>> ENTER: CREATE
   .. IF NONE OF THE ABOVE,
      >>> ENTER: NO

&IF &OPTCHGIND EQ 0 &GOTO -INENROLL2
&TYPE
&TYPE YOU HAVE CHANGED THE ENROLLMENT OPTION; THEREFORE, YOU MUST
&TYPE    EITHER USE THE SELECT OR CREATE OPTION.
&TYPE
-INENROLL2 &READ VAR &RESPONSE
&IF &RESPONSE EQ NO &GOTO -FACTOR
&IF &RESPONSE EQ DISPLAY &GOTO -ENROLLDISP
&IF &RESPONSE EQ SELECT &GOTO -ENROLLSEL
&IF &RESPONSE EQ CREATE &GOTO -ENROLLCREATE
&IF &RESPONSE EQ N &GOTO -FACTOR
&IF &RESPONSE EQ D &GOTO -ENROLLDISP
&IF &RESPONSE EQ S &GOTO -ENROLLSEL
&IF &RESPONSE EQ C &GOTO -ENROLLCREATE
&TYPE
&TYPE >>> ENTER DISPLAY, CHANGE, OR NO
&GOTO -INENROLL2
*
* VIEW OR CHANGE THE ENROLLMENT DATA THAT IS IN EFFECT
*
-ENROLLDISP
&IF &OPTCHGIND = 0 &GOTO -ENROLLDISP2
&TYPE
&TYPE YOU HAVE CHANGED THE ENROLLMENT OPTION; THEREFORE, YOU MUST
&TYPE    EITHER USE THE SELECT OR CREATE OPTION.
&TYPE
&GOTO -ENROLL2
-ENROLLDISP2
&IF &TAG EQ A &ENROLLER = &AENROLL
&IF &TAG EQ P &ENROLLER = &PENROLL
XEDIT &ENROLLER IFPS
&IF &OPTION NE 3 &GOTO -ENROLL2
&IF &OPT3COUNT NE 2 &GOTO -OPT3
&GOTO -ENROLL2
*
* VIEW THE ENROLLMENT DATA FILES THAT ARE AVAILABLE
*
-ENROLLSEL
&ETYPE = &CONCAT OF &TAG ENROLL
&LIST = &CONCAT OF &ETYPE *
&TYPE
&TYPE >>>>    WHEN READY TO REVIEW THE LIST OF ENROLLMENT DATA FILES
&TYPE            AVAILABLE,    PRESS 'RETURN'
&TYPE
&TYPE
&READ VAR
EXEC FILELIST &LIST IFPS
-NEWFILE
&TYPE
```

```
&TYPE ENTER THE NUMBER OF THE DATA FILE THAT YOU WANT USE, I.E. 1, 2,..
&TYPE
&READ VAR &RESPONSE
&TEST = &DATATYPE OF &RESPONSE
&IF &TEST EQ NUM &GOTO -CHECK2
&IF &RESPONSE EQ S &GOTO -ENROLL2
&TYPE ENTER A NUMBER
&GOTO -NEWFILE
-CHECK2
&TEST = &CONCAT OF &TAG ENROLL
&TEST2 = &CONCAT OF &TEST &RESPONSE
STATE &TEST2 IFPS A
&IF &RC EQ 0 &GOTO -CHECK3
&TYPE FILE DOES NOT EXIST
&GOTO -NEWFILE
-CHECK3
&ENROLLER = &TEST2
&GOTO -UPDATE
*
* CREATE A NEW ENROLLMENT DATA FILE
*
-ENROLLCREATE
&BEGTYPE 17

**************  CREATION OF A NEW ENROLLMENT DATA FILE  **********

    YOU MUST SPECIFY:
      1. THE NUMBER OF THE NEW DATA FILE WHICH SHOULD BE BETWEEN
         1 AND 9 AND NOT THE SAME AS ONE THAT IS ALREADY IN EFFECT.
      2. THE NUMBER OF AN EXISTING FILE THAT WILL BE THE BASIS
         FOR THE ONE THAT YOU ARE CREATING, I.E. THE NEW FILE THAT
         WILL BE CREATED WILL BE A COPY OF THE EXISTING FILE THAT
         YOU SPECIFY IN THIS STEP. IF YOU WANT TO USE THE DEFAULT
         FILE, SPECIFY B.

    IF YOU ARE READY TO ENTER THE TWO ITEMS ABOVE,
         >>> ENTER GO
    IF YOU NEED TO SEE THE DATA FILES AVAILABLE,
         >>> ENTER DISPLAY

-INCREATE1 &READ VARS &RESPONSE
&IF &RESPONSE EQ DISPLAY &GOTO -CREATE2
&IF &RESPONSE EQ GO &GOTO -CREATE3
&IF &RESPONSE EQ D &GOTO -CREATE2
&IF &RESPONSE EQ G &GOTO -CREATE3
&IF &RESPONSE EQ S &GOTO -FACTOR
&TYPE
&TYPE >>> ENTER GO OR DISPLAY
&TYPE
&GOTO -INCREATE1
-CREATE2
&ETYPE = &CONCAT OF &TAG ENROLL
&LIST = &CONCAT OF &ETYPE *
EXEC FILELIST &LIST IFPS A
&GOTO -ENROLLCREATE
```

```
-CREATE3
&BEGTYPE 3

   >>> ENTER THE NEW FILE NUMBER

&READ VAR &NEWFILENUM
&NEWNAME = &CONCAT OF &TAG ENROLL
&NEWFILENAME = &CONCAT OF &NEWNAME &NEWFILENUM
STATE &NEWFILENAME IFPS A
*NOTE: RC = 28 MEANS FILE DOES NOT EXIST
&IF &RC EQ 28 &GOTO -CREATE4
&TYPE
&TYPE FILE &NEWFILENAME ALREADY EXITS, TRY AGAIN
&TYPE
&GOTO -CREATE3
-CREATE4
&BEGTYPE 3

   >>> ENTER THE EXISTING FILE NUMBER TO BE USED AS A BASIS

&READ VAR &OLDFILENUM
&OLDNAME = &CONCAT OF &TAG ENROLL
&OLDFILENAME = &CONCAT OF &OLDNAME &OLDFILENUM
-CREATE5
STATE &OLDFILENAME IFPS A
&IF &RC EQ 0 &GOTO -CREATE7
*NOTE: RC = 0 MEANS FILE EXISTS
&TYPE
&TYPE FILE &OLDFILENAME DOES NOT EXIT, TRY AGAIN
&TYPE
&GOTO -CREATE4
-CREATE7
COPYFILE &OLDFILENAME IFPS A &NEWFILENAME = =
&ENROLLER = &NEWFILENAME
&GOTO -UPDATE
-FACTOR
*
* VIEW OR CHANGE THE ENROLLMENT FACTOR DATA THAT IS IN EFFECT
*
&TYPE
&TYPE ****************** ENROLLMENT FACTOR DATA **************************
&TYPE
&TYPE  >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO
-INFACTOR &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ YES &GOTO -FACTOR2
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ Y &GOTO -FACTOR2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INFACTOR
*
* VIEW OR CHANGE THE ENROLLMENT FACTOR DATA USED
*
-FACTOR2
```

```
XEDIT FACTOR IFPS
&GOTO -FACTOR
-UPDATE
*
* STORE NEW ENROLLMENT DATA FILE NAMES IN FILE SETUP DSS
*
&IF &TAG EQ A &AENROLL = &ENROLLER
&IF &TAG EQ P &PENROLL = &ENROLLER
&STACK ENROLL = &PENROLL &AENROLL
&NSET3 = &NSET2 + 1
EXECIO 1 DISKW SETUP DSS A &NSET3 F 80 (FINIS
&IF &OPTION NE 3 &GOTO -RESET1
&IF &OPT3COUNT EQ 2 &GOTO -RESET1
&GOTO -OPT3
-RESET1
&OPTCHGIND = 0
&GOTO -ENROLL2
-OUT &EXIT
```

```
&TRACE
*****************************************************************************
*                                                                           *
* COMPONENT CONTROL PROGRAM USED TO HANDLE FINANCIAL DATA ITEMS -           *
*    COST INFORMATION, STUDENT FEES, DORM & DINING, CAPITAL PROJECTS        *
*                                                                           *
*****************************************************************************
*
* ALL PARAMETER VALUES ARE PASSED FROM THE ECP, DSS EXEC
* &1 = STARTING YEAR OF THE ANALYSIS
* &2 = DORM & DINING CALCULATION OPTION NUMBER
&SYEAR = &1
&DDOPTION = &2
&BYEAR = &SYEAR - 1
-TUITHIST
*****************************************************************************
*                                                                           *
* VIEW OR CHANGE THE BASE-YEAR TUITION CHARGES                              *
*                                                                           *
*****************************************************************************
&BEGTYPE 7

                    ***************************
                    *                         *
****************** * BASE-YEAR TUITION CHARGES * ***************************
                    *                         *
                    ***************************

&TYPE THE BASE-YEAR TUITION CHARGES SHOULD BE ENTERED FOR &BYEAR
&BEGTYPE 3

       >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INHIST &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -COST
&IF &RESPONST EQ YES &GOTO -HIST2
&IF &RESPONSE EQ N &GOTO -COST
&IF &RESPONSE EQ Y &GOTO -HIST2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INHIST
-HIST2
XEDIT TUITHIST DSS
&GOTO -TUITHIST
-COST
*****************************************************************************
*                                                                           *
* VIEW OR CHANGE THE COST DATA THAT IS IN EFFECT                            *
*                                                                           *
*****************************************************************************
&BEGTYPE 9
```

```
                              ************
                              *           *
****************************** COST DATA *********************************
                              *           *
                              ************

      >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INCOST &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -FEES
&IF &RESPONSE EQ YES &GOTO -COST2
&IF &RESPONSE EQ N &GOTO -FEES
&IF &RESPONSE EQ Y &GOTO -COST2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INCOST
-COST2
XEDIT COST IFPS
&GOTO -COST
-FEES
*****************************************************************************
*                                                                           *
* VIEW OR CHANGE THE STUDENT FEE AND DORM & DINING DATA THAT IS IN          *
*      EFFECT                                                               *
*                                                                           *
*****************************************************************************
&BEGTYPE 9

                            ********************
                            *                  *
 ************************** STUDENT FEE DATA ***************************
                            *                  *
                            ********************

       >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INFEES &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -DORMDING
&IF &RESPONSE EQ YES &GOTO -FEES2
&IF &RESPONSE EQ N &GOTO -DORMDING
&IF &RESPONSE EQ Y &GOTO -FEES2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INFEES
-FEES2
&BEGTYPE 3

   >>> WHEN READY TO VIEW THE STUDENT FEES DATA BASE, PRESS 'RETURN'

&READ VAR
XEDIT FEEBASE IFPS
&GOTO -FEES
-DORMDING
&BEGTYPE 9
```

```
                        ************************
                        *                      *
**********************  * DORM AND DINING DATA *  ***************************
                        *                      *
                        ************************

     >>>  DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

-INDD &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -CAPITAL
&IF &RESPONSE EQ YES &GOTO -DD1
&IF &RESPONSE EQ N &GOTO -CAPITAL
&IF &RESPONSE EQ Y &GOTO -DD1
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INDD
-DD1 &BEGTYPE 8

THERE ARE TWO OPTIONS FOR CALCULATING THE DORM & DINING CHARGES:
     1. DIRECTLY INPUT THE CHARGES FOR EACH DORM TYPE
                          OR
     2. INPUT THE OVERALL OPERATING EXPENSES AND THE CHARGES ARE
        ALLOCATED TO THE INDIVIDUAL DORMS BASED ON A SET OF USER-
        SPECIFIED RATIOS.

&TYPE OPTION &DDOPTION IS IN EFFECT, DO YOU WANT TO CHANGE ?
&TYPE >>> ENTER YES OR NO.
&TYPE
-INDD2 &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -DD3
&IF &RESPONSE EQ YES &GOTO -DD2
&IF &RESPONSE EQ N &GOTO -DD3
&IF &RESPONSE EQ Y &GOTO -DD2
&TYPE
&TYPE >>> ENTER YES OR NO.
&GOTO -INDD2
-DD2 &BEGTYPE 3

  >>> INPUT THE OPTION NUMBER THAT YOU WANT TO USE

-INDD3 &READ VARS &RESPONSE
&IF &RESPONSE GT 2 &GOTO -DD2
&DDTEST = &DATATYPE OF &RESPONSE
&IF &DDTEST EQ CHAR &GOTO -DD2
&DDOPTION = &RESPONSE
-DD3
&DDFILE = &CONCAT OF DDBASE &DDOPTION
&BEGTYPE 3

   >>> WHEN READY TO VIEW THE DORM & DINING DATA BASE, PRESS 'RETURN'

&READ VAR
XEDIT &DDFILE IFPS
&GOTO -DORMDING
-CAPITAL
```

```
*************************************************************************
*                                                                       *
* VIEW OR CHANGE THE CAPITAL PROJECTS DATA TO BE USED IN THE FEE CALC.  *
*                                                                       *
*************************************************************************
&BEGTYPE 13

          ********************************************************
          *                                                      *
********** CAPITAL PROJECTS DATA FOR FEE CALCULATION PURPOSES **********
          *                                                      *
          ********************************************************

  .. IF YOU WANT TO VIEW OR CHANGE THE PROJECT DATA VALUES IN EFFECT,
     >>> ENTER: DISPLAY
  .. IF YOU WANT TO REGENERATE THE PROJECT DATA VALUES,
     >>> ENTER: GENERATE
  .. IF NEITHER OF THE ABOVE,
     >>> ENTER: NO
-INCAP &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ DISPLAY &GOTO -CAPINFO
&IF &RESPONSE EQ GENERATE &GOTO -CAPGEN
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ D &GOTO -CAPINFO
&IF &RESPONSE EQ G &GOTO -CAPGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INCAP
-CAPGEN
*
* GENERATE THE CAPITAL PROJECTS DATA FILE, CAPGEN DSS
*
SET CMSTYPE HT
ERASE CAPGEN DSS A
*FORTHX CAPGEN (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
*OUTPUT FILES
FILEDEF 05 DISK CAPGEN DSS A
LOAD CAPGEN (START NOMAP
SET CMSTYPE RT
-CAPINFO
XEDIT CAPGEN DSS
*
* CREATE THE FILE CAPIFPS IFPS FROM THE FILE CAPGEN DSS
*
SET CMSTYPE HT
ERASE CAPGEN IFPS A
*FORTHX CAPIFPS (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 04 DISK CAPGEN DSS A
*OUTPUT FILES
```

```
FILEDEF 05 DISK CAPIFPS IFPS A
LOAD CAPIFPS (START NOMAP
SET CMSTYPE RT
&GOTO -CAPITAL
-OUT &EXIT
```

```
&TRACE OFF
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO TRANSLATE THE PROBLEM               *
*    FORMULATION FROM MPS FORMAT TO EQUATION FORM AND STORE IN THE      *
*    FILE FORMAT TFPS                                                   *
*                                                                       *
*************************************************************************
*
* &1 IS THE UNIT NUMBER CORRESPONDING TO THE FILE CONTAINING THE PROBLEM
*    IN MPS FORMAT
*
&UNO = &1
ERASE FORMAT TFPS
CP SP CONS START TO *
&STACK TERSE
&STACK RMPS
&STACK &UNO
&STACK MIN
&STACK LOOK ALL
&STACK QUIT
LINDO
CP SP CONS STOP CLOSE
READ FORMAT TFPS A
&EXIT
```

```
&TRACE
*********************************************************************
*                                                                   *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE GENERAL           *
*    INDIVIDUAL GOALS                                               *
*                                                                   *
*********************************************************************
*
* &1 = INDICATOR THAT THE PROBLEM STRUCTURE HAS CHANGED (PASSED FROM
*    DSS EXEC)
&PHCHGIND = &1
&IF &PHCHGIND EQ 0 &GOTO -INGENIND
&BEGTYPE 6

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS CHANGED,
            THE GENERAL INDIVIDUAL GOAL TARGET VALUES
                  MUST BE RE-ESTABLISHED.

          WHEN READY TO RE-ENTER: PRESS 'RETURN'
&READ VARS
&GOTO -GENINDGEN
-INGENIND &BEGTYPE 13

                ***********************************
                *                                 *
**************  GENERAL INDIVIDUAL GOAL TARGET VALUES  **************
                *                                 *
                ***********************************

        .. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
           >>> ENTER: DISPLAY
        .. IF YOU WANT TO REGENERATE THE GOAL VALUES,
           >>> ENTER: GENERATE
        .. IF NEITHER OF THE ABOVE,
           >>> ENTER: NO
-INGIND &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ DISPLAY &GOTO -GENINDDISP
&IF &RESPONSE EQ GENERATE &GOTO -GENINDGEN
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ D &GOTO -GENINDDISP
&IF &RESPONSE EQ G &GOTO -GENINDGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INGIND
-GENINDGEN
&TYPE GENERATING FILE FOR GENERAL INDIVIDUAL GOAL TARGET VALUES
SET CMSTYPE HT
ERASE GENIND TEMPS A
*FORTHX GENINGEN (NOPRINT
GLOBAL TXTLIB FORTXTLIB
*INPUT FILES
```

```
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 06 DISK GENIND IFPS A
LOAD GENINGEN (START NOMAP
SET CMSTYPE RT
-GENINDDISP
&STACK SET RECF F
XEDIT GENIND IFPS A
*
* CONVERT GENERAL INDIVIDUAL GOALS TO MPS FORMAT
*
&TYPE SETTING UP GENERAL INDIVIDUAL GOALS IN MPS FORMAT
SET CMSTYPE HT
ERASE GINDRHS DSS A
ERASE GINDX DSS A
ERASE GINDNP DSS A
*FORTHX GENIND (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 05 DISK GENIND IFPS A
*OUTPUT FILES
FILEDEF 06 DISK GINDRHS DSS A
FILEDEF 07 DISK GINDX DSS A
FILEDEF 08 DISK GINDNP DSS A
LOAD GENIND (START NOMAP
SET CMSTYPE RT
-OUT &EXIT
```

```
&TRACE OFF
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO SOLVE THE GOAL PROGRAMMING          *
*     PROBLEM INTERACTIVELY USING LINDO AND THE USER SUBROUTINE         *
*                                                                       *
*************************************************************************
*
* &1 IS THE UNIT NUMBER CORRESPONDING TO THE FILE CONTAINING THE PROBLEM
*    IN MPS FORMAT
&UNO = &1
&BEGTYPE 4

 THE TUITION CHARGES ARE BEING DETERMINED BY THE GOAL PROGRAMMING
    ROUTINE, INTERACTIVELY, USING LINDO

*CP QUERY TIME
SET CMSTYPE HT
ERASE TUITION IFPS A
ERASE TUIT IFPS A
ERASE DEVVAR IFPS A
&STACK TERSE
&STACK RMPS
*&STACK 1
&STACK &UNO
&STACK MIN
*&STACK LOOK ALL
&STACK USER
&STACK QUIT
GLOBAL TXTLIB FORTXLIB
*FORTHX USER (NOPRINT
*INPUT FILES
FILEDEF 10 DISK SETUP DSS A
FILEDEF 11 DISK BASEYR DSS A
FILEDEF 12 DISK PRIOROPT DSS A
*OUTPUT FILES
FILEDEF 14 DISK DEVVAR IFPS A
FILEDEF 15 DISK TUIT IFPS A
FILEDEF 16 DISK TUITION IFPS A
FILEDEF 17 DISK TRACE IFPS A
LOAD LINDO USER (START NOMAP
SET CMSTYPE RT
*CP QUERY TIME
&EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE MPS FORMAT FILE       *
*      USED BY LINDO TO SOLVE THE GOAL PROGRAMMING PROBLEM              *
*                                                                       *
*************************************************************************
* &1 = FILETYPE OF THE FILE CONTAINING THE PROBLEM FORMULATION IN
*    MPS FORMAT (PASSED FROM DSS EXEC
&FILETYPE = &1
&BEGTYPE 3

   ...  THIS MAY TAKE A MOMENT

*
* COMBINE ALL MPS-RELATED FILES INTO FILE FT__F001 TO FORMULATE THE
*      GOAL PROGRAMMING PROBLEM TO BE READ BY LINDO.
*
** NAME CARD AND ROWS
*** ROWS WERE INCLUDED IN FILE FT__F001 IN DSS EXEC
** COLUMN CARDS - DEVIATION VARIABLES
*** COMBINE ALL DEVIATION VARIABLES FROM AGRGNP DSS, PCOSTNP DSS,
***    CTIMENP DSS, PEERNP DSS, GINDNP DSS, AND COMPNP DSS
***      TO DSSNP DSS AND SORT
*** SORT ON DEVIATION VARIABLE NUMBER, THEN ON N OR P.
&TYPE SETTING UP PROBLEM IN MPS FORMAT TO BE USED BY LINDO
SET CMSTYPE HT
&STACK GET AGRGNP DSS A
&STACK BOTTOM
&STACK GET PCOSTNP DSS A
&STACK BOTTOM
&STACK GET CTIMENP DSS A
&STACK BOTTOM
&STACK GET PEERNP DSS A
&STACK BOTTOM
&STACK GET GINDNP DSS A
&STACK BOTTOM
&STACK GET COMPNP DSS A
&STACK FILE
XEDIT DSSNP DSS A
&STACK 10 12 9 9
SORTF DSSNP DSS A DSSNPS = =
ERASE DSSNP DSS A
** COLUMN CARDS - DECISION VARIABLES
*** COMBINE ALL DECISION VARIABLES FROM AGRGX DSS, PCOSTX DSS,
***    CTIMEX DSS, PEERX DSS, GINDNP DSS, AND COMPX DSS
***      TO DSSX DSS AND SORT
*** SORT ON YEAR, THEN STUDENT CATEGORY, THEN ROW NUMBER.
&STACK GET AGRGX DSS A
&STACK BOTTOM
&STACK GET PCOSTX DSS A
&STACK BOTTOM
```

```
&STACK GET CTIMEX DSS A
&STACK BOTTOM
&STACK GET PEERX DSS A
&STACK BOTTOM
&STACK GET GINDX DSS A
&STACK BOTTOM
&STACK GET COMPX DSS A
&STACK FILE
XEDIT DSSX DSS A
&STACK 12 12 11 11 15 22
SORTF DSSX DSS A DSSXS = =
ERASE DSSX DSS A
** RIGHT-HAND SIDE CARDS
*** COMBINE ALL RIGHT-HAND SIDE VALUES FROM AGRGRHS DSS, PCOSTRHS DSS,
***   CTIMERHS DSS, PEERRHS DSS, GINDRHS DSS, AND COMPRHS DSS
***     TO DSSRHS DSS AND SORT
&STACK GET AGRGRHS DSS A
&STACK BOTTOM
&STACK GET PCOSTRHS DSS A
&STACK BOTTOM
&STACK GET CTIMERHS DSS A
&STACK BOTTOM
&STACK GET PEERRHS DSS A
&STACK BOTTOM
&STACK GET GINDRHS DSS A
&STACK BOTTOM
&STACK GET COMPRHS DSS A
&STACK FILE
XEDIT DSSRHS DSS A
&STACK 1 50
SORTF DSSRHS DSS A DSSRHSS = =
ERASE DSSRHS DSS A
** COMBINE ROWS, DEVIATION VARIABLES, DECISION VARIABLES, AND RHS
**     INTO AN MPS FILE, FILE FT__F001, TO BE READ BY LINDO
&STACK BOTTOM
&STACK INPUT
&STACK COLUMNS
&STACK
&STACK GET DSSNPS DSS A
&STACK BOTTOM
&STACK GET DSSXS DSS A
&STACK BOTTOM
&STACK INPUT
&STACK RHS
&STACK
&STACK GET DSSRHSS DSS A
&STACK BOTTOM
&STACK INPUT
&STACK ENDATA
&STACK
&STACK FILE
XEDIT FILE &FILETYPE
ERASE DSSNPS DSS A
ERASE DSSXS DSS A
ERASE DSSRHSS DSS A
SET CMSTYPE RT
&EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO:                                    *
*   1) ESTABLISH THE PEER INSTITUTION GOAL TARGET VALUES FOR EACH       *
*      STUDENT CATEGORY BY YEAR.                                        *
*                                                                       *
*   2) CONVERT THE PEER INSTITUTION GOALS TO MPS FORMAT                 *
*                                                                       *
*************************************************************************
SET CMSTYPE HT
*FORTHX PEERCALC (NOPRINT
ERASE PEERX DSS
ERASE PEERNP DSS
ERASE PEERRHS DSS
ERASE PEER TFPS
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 01 DISK SETUP DSS A
FILEDEF 02 DISK PEERINFO DSS A
FILEDEF 03 DISK PEERCHG TFPS A
FILEDEF 04 DISK DDTEEMPS TFPS A
*OUTPUT FILES
FILEDEF 06 DISK PEER TFPS A
FILEDEF 07 DISK PEERRHS DSS A
FILEDEF 08 DISK PEERX DSS A
FILEDEF 09 DISK PEERNP DSS A
LOAD PEERCALC (START NOMAP
SET CMSTYPE RT
&EXIT
```

```
&TRACE
*********************************************************************
*                                                                   *
* COMPONENT CONTROL PROGRAM USED TO ESTIMATE THE TUITION CHARGES AT  *
*   PEER INSTITUTIONS                                               *
*                                                                   *
*********************************************************************
*
*&1 = INDICATOR THAT THE PROBLEM STRUCTURE HAS CHANGED (PASSED FROM DSS
*&2 = NSCAT (PASSED FROM DSS EXEC
&PHCHGIND = &1
&NSCAT = &2
-MSG1 &BEGTYPE 8

*** INFORMATION REQUIRED TO ESTIMATE THE CHARGES AT PEER INSTITUTIONS ***
  .. IF YOU WANT TO VIEW OR CHANGE THE INFORMATION IN EFFECT,
     >>> ENTER: DISPLAY
  .. IF YOU WANT TO REGENERATE THE PEER INFORMATION,
     >>> ENTER: GENERATE
  .. IF NEITHER OF THE ABOVE,
     >>> ENTER: NO
-INPEER &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -AROUND
&IF &RESPONSE EQ DISPLAY &GOTO -PEERINFO
&IF &RESPONSE EQ GENERATE &GOTO -PEERGEN
&IF &RESPONSE EQ N &GOTO -AROUND
&IF &RESPONSE EQ D &GOTO -PEERINFO
&IF &RESPONSE EQ G &GOTO -PEERGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INPEER
-PEERGEN
*
* GENERATE PEER INSTITUTION INFORMATION FILE, PEERGEN IFPS
*
&TYPE SETTING UP PEER INSTITUTION INFORMATION FILE
SET CMSTYPE RT
ERASE PEERGEN IFPS A
*FORTHX PEERGEN (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 05 DISK PEERGEN IFPS A
LOAD PEERGEN (START NOMAP
SET CMSTYPE RT
-PEERINFO
XEDIT PEERGEN IFPS A
&TYPE PROCESSING PEER INSTITUTION GOAL INFORMATION
*
* SETUP THE FILE PEERINFO DSS TO BE READ BY THE FORTRAN PROGRAM PEERMCRO
*
```

```
* &N1 IS THE NUMBER OF TITLE LINES THAT MUST BE SKIPPED (TITLE LINES)
*     IN FILE PEERGEN IFPS
&N1 = 9
ERASE PEERINFO DSS A
&N = 0
&LOOP -INPUT2 &NSCAT
&N = &N + 1
&N2 = &N1 + &N
EXECIO 1 DISKR PEERGEN IFPS A &N2 (FINIS
&READ VARS &A &B &C &D &E &F &G &H &J
&IF &D EQ MN &D = -1
&IF &D EQ DI &D = 00
&IF &E EQ Y &E = 1
&IF &E EQ N &E = 0
&IF &F EQ Y &F = 1
&IF &F EQ N &F = 0
&IF &G EQ Y &G = 1
&IF &G EQ N &G = 0
&IF &H EQ Y &H = 1
&IF &H EQ N &H = 0
&IF &J EQ Y &J = 1
&IF &J EQ N &J = 0
&STACK &B &C &D &E &F &G &H &J
EXECIO 1 DISKW PEERINFO DSS A &N F 80 (FINIS
-INPUT2
*
* READ INFORMATION FROM FILE PEERGEN IFPS
* SETUP AN IFPS DATAFILE FOR EACH STUDENT CATEGORY BASED ON THE DATA
*     SOURCE REQUESTED IN FILE PEERGEN IFPS (I.E. LAND GRANT, NCHEMS 1A,
*     VIRGINIA, ETC.
*
&N = 0
&ERRORCOUNT = 0
&ERROR1 = 0
&ERROR2 = 0
&ERROR3 = 0
&ERROR4 = 0
&ERROR5 = 0
&ERROR6 = 0
&FN1 = RUON
&FN2 = NUON
&FN3 = RGON
&FN4 = NGON
&FN5 = RGOF
&FN6 = NGOF
SET CMSTYPE HT
&LOOP -INPUT &NSCAT
&N = &N + 1
&N2 = &N1 + &N
EXECIO 1 DISKR PEERGEN IFPS A &N2 (FINIS
&READ VARS &A &B &C  &D &E &F &G &H &J
&A = &CONCAT OF FN &N
&IF &J = Y &GOTO -TOTAL
&FILEIN = &CONCAT OF &&A &C
&FILEOUT = &CONCAT OF PEER &N
```

```
&GOTO -A
-TOTAL
&FILEINA = &CONCAT OF &&A &C
&FILEIN = &CONCAT OF &FILEINA TO
&FILEOUT = &CONCAT OF PEER &N
-A
&IF &C EQ DI &GOTO -B
STATE &FILEIN IFPS A
&IF &RC EQ 0 &GOTO -B
*
* RECORD THE FILE NAME THAT DOES NOT EXIST
*
&ERRORCOUNT = &ERRORCOUNT + 1
&AA = &CONCAT OF ERROR &ERRORCOUNT
&&AA = &FILEIN
&C = DI
-B
*&TYPE &A &FILEIN &FILEOUT &C
ERASE &FILEOUT IFPS A
&IF &C EQ DI &GOTO -DIRINP
COPYFILE &FILEIN IFPS A &FILEOUT = =
&GOTO -IN
-DIRINP
COPYFILE DIRECTIN IFPS A &FILEOUT = =
-IN &STACK INPUT
&STACK NAME
&STACK PERCENTILE
&STACK
&STACK FILE
XEDIT &FILEOUT IFPS A
-INPUT
SET CMSTYPE RT
&IF &ERRORCOUNT EQ 0 &GOTO -OK
&TYPE
&TYPE ERROR - &ERRORCOUNT OF THE FILES THAT YOU SPECIFIED FOR THE
&TYPE          PEER INSTITUTION VALUES DO NOT EXIST
&TYPE
&TYPE          THE FOLLOWING FILE(S) ARE NOT AVAILABLE:
&N = 0
&LOOP -FILECHK &NSCAT
&N = &N + 1
&ERRORN = &CONCAT OF ERROR &N
&IF &&ERRORN EQ 0 &GOTO -FILECHK
&TYPE                &&ERRORN
-FILECHK
&TYPE
&TYPE          THEY WILL BE HANDLED AS DIRECT INPUT, IF THIS OK, ENTER OK
&TYPE
&TYPE          IF YOU WANT TO RESPECIFY THE FILE'S SOURCE, ENTER AGAIN
&TYPE
-INERROR &READ VARS &RESPONSE
&IF &RESPONSE EQ OK &GOTO -OK
&IF &RESPONSE EQ AGAIN &GOTO -PEERINFO
&IF &RESPONSE EQ 0 &GOTO -OK
&IF &RESPONSE EQ A &GOTO -PEERINFO
```

```
&TYPE
&TYPE >>> ENTER OK OR AGAIN
&GOTO -INERROR
-OK
*
* CREATE THE MICRO STATEMENTS FOR THE IFPS FILES (STUDENT CATEGORY NAME
*     AND MEASURE TO BE USED)
*
SET CMSTYPE HT
*FORTHX PEERMCRO (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 04 DISK PEERINFO DSS A
*OUTPUT FILES
FILEDEF 05 DISK PEERMCRO DSS A
LOAD PEERMCRO (START NOMAP
SET CMSTYPE RT
*
* PLACE THE MICRO STATEMENTS IN THE APPROPRIATE IFPS DATAFILE
*
&N = 0
&LOOP -MICRO &NSCAT
&N2 = &MULT OF &N 2
&NN = &N2 + 1
&N = &N + 1
&NAME = &CONCAT OF PEER &N
EXECIO 2 DISKR PEERMCRO DSS A &NN (FINIS
EXECIO 2 DISKW &NAME IFPS A 1 (FINIS
-MICRO
*
* CALCULATE PEER INSTITUTION VALUES USING IFPS
*
&BEGTYPE 3

CALCULATING PEER INSTITUTION VALUES USING IFPS

&NSCAT2 = &CONCAT OF ' &NSCAT
&NSCAT3 = &CONCAT OF &NSCAT2 '
EXEC IFPSDSS A B C DSS PEERCOM &NSCAT3
&BEGTYPE 3

FINISHED WITH IFPS FOR NOW

*
* DISPLAY PEER INSTITUTION VALUES; USER SUPPLIES DIRECT INPUT DATA
*
&BEGTYPE -DIRIN

****** DISPLAY OF THE PEER INSTITUTION THAT ARE TO BE USED **********
    ... THE QUANTITIES WHICH ARE 0, ARE TO BE SUPPLIED BY THE USER.

    ... WHEN READY TO VIEW/INPUT PEER INSTITUTION VALUES,
        >>> PRESS 'RETURN'
```

```
-DIRIN
&READ VARS
XEDIT PEERCHG IFPS
*
*CONVERT THE IFPS FILE FORMAT FROM V TO F
*
&STACK SET RECF F
&STACK FILE
XEDIT PEERCHG IFPS
&GOTO -OUT
-AROUND &IF &PHCHGIND EQ 0 &GOTO -OUT
&BEGTYPE 4

THE PLANNING HORIZON HAS CHANGED, THEREFORE YOU CANNOT USE THE "NO"
   OPTION, ENTER EITHER "DISPLAY" OR "GENERATE"

&GOTO -MSG1
-OUT &EXIT
```

```
&TRACE
*********************************************************************
*                                                                   *
* COMPONENT CONTROL PROGRAM USED TO ESTABLISH THE PERCENTAGE-OF-    *
*     AVERAGE-TOTAL-COST GOALS                                      *
*                                                                   *
*********************************************************************
*
* &1 = INDICATOR THAT THE PROBLEM STRUCTURE HAS CHANGED (PASSED FROM
*    DSS EXEC)
&PHCHGIND = &1
&IF &PHCHGIND EQ 0 &GOTO -INPERCOST
&BEGTYPE 6

NOTE!!!!  SINCE THE STRUCTURE OF THE PROBLEM HAS CHANGED, THE
          PERCENTAGE-OF-AVERAGE-TOTAL-COST GOAL TARGET VALUES
          MUST BE RE-ESTABLISHED.

            WHEN READY TO RE-ENTER: PRESS 'RETURN'
&READ VARS
&GOTO -PERCOSTGEN
-INPERCOST &BEGTYPE 13

         ***************************************************
         *                                                 *
******** PERCENTAGE OF AVERAGE TOTAL COST GOAL TARGET VALUES ********
         *                                                 *
         ***************************************************

         .. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
            >>> ENTER: DISPLAY
         .. IF YOU WANT TO REGENERATE THE GOAL VALUES,
            >>> ENTER: GENERATE
         .. IF NEITHER OF THE ABOVE,
            >>> ENTER: NO
-INPCOST &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ DISPLAY &GOTO -PERCOSTDISP
&IF &RESPONSE EQ GENERATE &GOTO -PERCOSTGEN
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ D &GOTO -PERCOSTDISP
&IF &RESPONSE EQ G &GOTO -PERCOSTGEN
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INPCOST
-PERCOSTGEN
&TYPE GENERATING FILE FOR PERCENTAGE-OF-AVG-TOTAL-COST GOAL TARGET VALUES
SET CMSTYPE HT
ERASE PERCOST2 LISTS A
*FORTHX PCOSTGEN (NOPRINT
GLOBAL TXTLIB FORTXTLIB
*INPUT LISTS
```

```
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 06 DISK PERCOST2 IFPS A
LOAD PCOSTGEN (START NOMAP
SET CMSTYPE RT
&STACK SET RECF F
-PERCOSTDISP
XEDIT PERCOST2 IFPS A
*
* CONVERT PERCENTAGE-OF-AVERAGE-COST GOALS TO MPS FORMAT
*
&TYPE SETTING UP PERCENTAGE-OF-AVERAGE-COST GOALS IN MPS FORMAT
SET CMSTYPE HT
ERASE PCOSTRHS DSS A
ERASE PCOSTX DSS A
ERASE PCOSTNP DSS A
*FORTHX PERCOST (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 04 DISK PERCOST IFPS A
FILEDEF 05 DISK PERCOST2 IFPS A
*OUTPUT FILES
FILEDEF 06 DISK PCOSTRHS DSS A
FILEDEF 07 DISK PCOSTX DSS A
FILEDEF 08 DISK PCOSTNP DSS A
LOAD PERCOST (START NOMAP
SET CMSTYPE RT
-OUT &EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO PROCESS THE PRIORITY STRUCTURE      *
*      AND CONVERT IT TO A FORM THAT IS READABLE BY THE OPTIMIZATION    *
*      ROUTINE AND LINDO                                                *
*                                                                       *
*************************************************************************
*&1 = INDICATOR THAT PROBLEM STRUCTURE HAS CHANGED (FROM DSS EXEC)
*&2 = &NREAD, FROM DSS EXEC
*&3 = &NPRIOR, FROM DSS EXEC
&IF &1 EQ 0 &GOTO -CHOICE
&TYPE REGENERATING PRIORITY STRUCTURE
&GOTO -PRI
-CHOICE
*    .. YOU WILL NOW BE ABLE TO CHANGE THE PRIORITY STRUCTURE
*    .. WHEN THE PRIORITY STRUCTURE IS DISPLAYED,
*       >>> CHANGE THE PRIORITY LEVEL BY PLACING THE CURSOR OVER THE
*           EXISTING PRIORITY LEVEL AND MAKE THE DESIRED CHANGE.
*       >>> TO CHANGE THE GOALS NOT SEEN ON THE SCREEN PRESS PF8
*       >>> WHEN ALL CHANGES HAVE BEEN MADE: PRESS 'RETURN', TYPE 'FILE',
*           PRESS 'RETURN'.
*-----------------------------------------------------------------------
&BEGTYPE 9

********************** PRIORITY STRUCTURE **************************
        .. IF YOU WANT TO VIEW OR CHANGE THE PRIORITY STRUCTURE IN EFFECT,
           >>> ENTER: DISPLAY
        .. IF YOU WANT TO REGENERATE THE PRIORITY STRUCTURE
           >>> ENTER: GENERATE
        .. IF YOU DO NOT WANT TO VIEW OR CHANGE THE PRIORITY STRUCTURE,
           >>>ENTER: NO

&READ VAR &RESPONSE
&IF &RESPONSE EQ DISPLAY &GOTO -DISPLAY
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ GENERATE &GOTO -PRI
&IF &RESPONSE EQ D &GOTO -DISPLAY
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ G &GOTO -PRI
&GOTO -CHOICE
-PRI
&TYPE REGENERATING THE PRIORITY STRUCTURE
SET CMSTYPE HT
ERASE PRIOR TEPS
*FORTHX PRIOR (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK PRIOR TEPS A
LOAD PRIOR (START NOMAP
```

```
SET CMSTYPE RT
&TYPE SETTING UP PRIORITY STRUCTURE
-DISPLAY
XEDIT PRIOR IFPS A
&1 = 0
* NREAD IS THE NUMBER OF LINES TO BE READ FROM FILE PRIOR IFPS.
*  IT IS PASSED FROM DSS EXEC AS PARAMETER 2.
*   NGOAL IS THE NUMBER OF GOALS CONSIDERED IN THE MODEL.
*   NSPACE IS THE NUMBER OF BLANK LINES IN FILE PRIOR IFPS.
*   NINDG IS THE NUMBER OF INDIVIDUAL GOAL TITLE LINES.
*   EACH 4-CATEGORY ABSOLUTE GOAL REQUIRES 2 LINES.
*   THERE IS A TITLE LINE FOR EACH OF THE 3 TYPES OF ABSOLUTE GOALS AND
*     THE RELATIVE GOALS.
*   THERE ARE 11 TOP HEADING LINES.
*
*TRANSFORM PRIORITY STRUCTURE, GIVEN IN FILE PRIOR IFPS, TO A FORM
*   READABLE BY LINDO S/R USER.
ERASE PRIOROPT DSS
&NREAD = &2
&NPRIOR = &3
&KOUNT = 0
EXECIO &NREAD DISKR PRIOR IFPS A (MARGIN 45 70 FINIS
&LOOP -GET2 &NREAD
&READ VARS &DEV1 &DEV2 &WGT1 &WGT2
&TEST = &TYPE OF &DEV1
&IF &TEST EQ CHAR &GOTO -GET2
&STACK &DEV1 &DEV2 &WGT1 &WGT2
&KOUNT = &KOUNT + 1
-GET2
ERASE PRIOROPT DSS A
EXECIO &KOUNT DISKW PRIOROPT DSS A 1 F 80 (FINIS
*
* CHECK TO SEE IF THE PRIORITY LEVELS HAVE BEEN ENTERED SEQUENTIALLY
*   I.E. ALL LEVELS, FROM 1 TO NPL, HAVE BEEN ENTERED.
*
SET CMSTYPE HT
*FORTHX PRIORCHK (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
FILEDEF 04 DISK PRIOROPT DSS A
*OUTPUT FILES
FILEDEF 05 DISK ERROR DSS A
LOAD PRIORCHK (START NOMAP
SET CMSTYPE RT
STATE ERROR DSS A
&IF &RC NE 0 &GOTO -OUT
ERASE ERROR DSS A
&BEGTYPE 4

ERROR !!!!! - YOU HAVE OMITTED AT LEAST ONE PRIORITY LEVEL IN
    THE SEQUENCE
 WHEN READY TO CORRECT THE PRIORITY STRUCTURE, PRESS 'RETURN'.
&READ VARS
&GOTO -DISPLAY
-OUT &EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROLL PROGRAM USED TO SELECT WITCH MANAGEMENT            *
*      REPORTS TO PRINT AND ON WHAT PRINTER                             *
*                                                                       *
*************************************************************************
* ALL OF THE FOLLOWING PARAMLETERS ARE PASSED FROM THE ECP, DSS EXEC
* &1 = "PROJECTED" ENROLLMENT DATA FILE NAME
* &2 = "ACTUAL" ENROLLMENT DATA FILE NAME
* &3 = NUMBER OF DIFFERENT REPORTS AVAILABLE
* &4 = DORM AND DINING CALCULATION OPTION
* &NTLINES = NUMBER OF TITLE LINES AT THE TOP OF FILE REPORTS DSS
&PENROLL = &1
&AENROLL = &2
&NREPORTS = &3
&DDOPTION = &4
&NTLINES = 11
&BEGTYPE 10

                      **************************
                      *                        *
**********************    MANAGEMENT REPORTS    **************************
                      *                        *
                      **************************

        .. DO YOU WANT TO PRINT ANY MANAGEMENT REPORTS ?
                        >>> ENTER: YES OR NO

-INREP &READ VARS &RESPONSE
&IF &RESPONSE EQ NO &GOTO -OUT
&IF &RESPONSE EQ YES &GOTO -REPDISP
&IF &RESPONSE EQ N &GOTO -OUT
&IF &RESPONSE EQ Y &GOTO -REPDISP
&TYPE
&TYPE >>> ENTER DISPLAY, GENERATE, OR NO
&GOTO -INREP
-REPDISP
*
* ESTABLISH BOX NUMBER WHERE OUTPUT IS TO BE SENT
*
&TYPE
&TYPE >>> ENTER THE NUMBER OF THE BOX WHERE THE OUPUT IS TO BE SENT
&READ VARS &A
&TESTA = &DATATYPE OF &A
&IF &TESTA NE NUM &GOTO -REPDISP
&IF &A EQ 0 &GOTO -B4
&IF &A GT 10 &GOTO -B1
&BOX = &CONCAT OF B000 &A
&GOTO -SELECT
-B1 &IF &A GT 100 &GOTO -B2
&BOX = &CONCAT OF B00 &A
```

```
&GOTO -SELECT
-B2 &IF &A GT 1000 &GOTO -B3
&BOX = &CONCAT OF B0 &A
&GOTO -SELECT
-B3 &IF &A GT 10000 &GOTO -B4
&BOX = &CONCAT OF B &A
&GOTO -SELECT
-B4 &BOX = B1305
-SELECT VTTAG DEVICE PRINTER DIST &BOX
XEDIT REPORTS DSS
*
* READ THE USER'S SELECTION FROM THE FILE REPORTS DSS
*
*
* PRINT ALL REPORTS GOING TO THE REGULAR PRINTER
*
&K = 0
&KR = 0
&LOOP -REGPRT &NREPORTS
&K = &K + 1
&LINE = &NTLINES + &K
EXECIO 1 DISKR REPORTS DSS A &LINE (FINIS
&READ VARS &A &B
&IF &A NE R &GOTO -REGPRT
&KR = &KR + 1
&IF &B NE TOTCOST &GOTO -PRINTR
*
* GENERATE STUDENT CHARGES REPORT BY CATEGORY
*
&BEGTYPE 3

GENERATING THE STUDENT CHARGES REPORTS USING IFPS

EXEC IFPSDSS A B C DSS TUITCOM &PENROLL &AENROLL &DDOPTION
&BEGTYPE 3

FINISHED WITH IFPS FOR NOW

-PRINTR
PRINT &B IFPS
-REGPRT
*
* PRINT ALL REPORTS GOING TO THE LASER PRINTER
*
* READ FROM THE FILE REPORTS DSS
*
VTTAG DEVICE PRINTER DEST RMT48
&K = 0
&KL = 0
&LOOP -LASPRT &NREPORTS
&K = &K + 1
&LINE = &NTLINES + &K
EXECIO 1 DISKR REPORTS DSS A &LINE (FINIS
&READ VARS &A &B
&IF &A NE L &GOTO -LASPRT
```

```
&KL = &KL + 1
&IF &B NE TOTCOST &GOTO -PRINTL
*
* GENERATE THE STUDENT CHARGES REPORT BY CATEGORY
*
&BEGTYPE 3

GENERATING THE STUDENT CHARGES REPORTS USING IFPS

EXEC IFPSDSS A B C DSS TUITCOM &ENROLL &CHANGE &DDOPTION
&BEGTYPE 3

FINISHED WITH IFPS FOR NOW

-PRINTL
PRINT &B IFPS A (NOH
-LASPRT
VTTAG DEVICE PRINTER DEST
&TYPE
&TYPE &KR REPORTS SENT TO BOX &BOX FROM THE REGULAR PRINTER
&TYPE &KL REPORTS SENT TO BOX &BOX FROM THE LASER PRINTER
-OUT &EXIT
```

```
&TRACE
*************************************************************************
*                                                                       *
* COMPONENT CONTROL PROGRAM USED TO DEFINE THE PROBLEM FRAMEWORK        *
*    IN TFPS FORMAT AND TO CREATE AN MPS-FORMAT FILE FOR THE            *
*    OPTIMIZATION ROUTINE.                                              *
*                                                                       *
*************************************************************************
* &1 IS THE FILETYPE OF THE FILE CONTAINING THE PROBLEM FORMULATION
*    IN MPS FORMAT (PASSED FROM DSS EXEC
&FILETYPE = &1
*************************************************************************
*                                                                       *
* ESTABLISH THE COLUMNS STATEMENT (BASED ON THE NUMBER OF YEARS         *
*    IN THE PLANNING HORIZON AND THE START YEAR) AND THE STUDENT        *
*    CATEGORY NAMES.                                                    *
* BOTH ARE USED BY THE TFPS MODELS. THE FILENAME IS SETUP TFPS          *
*                                                                       *
*************************************************************************
SET CMSTYPE HT
ERASE SETUP TFPS
*FORTHX SETUP (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 04 DISK SETUP TFPS A
LOAD SETUP (START NOMAP
SET CMSTYPE RT
*************************************************************************
*                                                                       *
* GENERATE ROWS SECTION OF MPS INPUT FILE, CREATE FILE FT__F001         *
*     TO BE READ BY LINDO OPTIMIZATION ROUTINE                          *
*                                                                       *
*************************************************************************
-ROWS STATE FILE &FILETYPE A
SET CMSTYPE HT
&IF &RC NE 0 &GOTO -ROW2
ERASE FILE &FILETYPE A
-ROW2
* FORTHX ROWS (NOPRINT
GLOBAL TXTLIB FORTXLIB
*INPUT FILES
FILEDEF 03 DISK SETUP DSS A
*OUTPUT FILES
FILEDEF 12 DISK FILE &FILETYPE A
LOAD ROWS (START NOMAP
SET CMSTYPE RT
&EXIT
```

Appendix C

PROGRAM LISTINGS FOR THE FORTRAN PROGRAMS

C.1 FORTRAN Program AGRGMPS
C.2 FORTRAN Program CAPGEN
C.3 FORTRAN Program CAPIFPS
C.4 FORTRAN Program CHGTIME
C.5 FORTRAN Program COMPGEN
C.6 FORTRAN Program COMPMPS
C.7 FORTRAN Program COMPSET
C.8 FORTRAN Program CTIMEGEN
C.9 FORTRAN Program GENIND
C.10 FORTRAN Program GENINGEN
C.11 FORTRAN Program PCOSTGEN
C.12 FORTRAN Program PEERCALC
C.13 FORTRAN Program PEERGEN
C.14 FORTRAN Program PEERMCRO
C.15 FORTRAN Program PERCOST
C.16 FORTRAN Program PRIOR
C.17 FORTRAN Program PRIORCHK
C.18 FORTRAN Program ROWS
C.19 FORTRAN Program SETUP
C.20 FORTRAN Program SUPLMREV
C.21 FORTRAN Program USER

```
C
C CONVERTS THE AGGREGATE GOALS TO MPS FORMAT
C
C MAXIMUM NUMBER OF STUDENT CATEGORIES ALLOWED IS 9
C MAXIMUM NUMBER OF YEARS IN PLANNING HORIZON ALLOWED IS 6
C
      DIMENSION CREDIT(10,10), RQMT(30)
      DIMENSION VNP(2),DV(2)
      DATA VNP/1.,-1./, DV/'N','P'/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
   10 FORMAT(8X,I1)
   11 FORMAT(7X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
C
C AGGREGATE GOALS INCLUDE:
C       1) STATE REVENUE REQUIREMENT GOALS - BY YEAR,
C       2) SUPPLEMENTAL REVENUE REQUIREMENTS GOALS - BY YEAR, AND
C       3) SUPPLEMENTAL REVENUE REQUIREMNETS GOALS - OVERALL
C
C READ THE DATA
C   * EXPECTED CREDIT HOUR DATA FOR THE COEFFICIENTS OF THE STATE
C     REVENUE REQUIREMENT GOALS, BY YEAR AND BY STUDENT CATEGORY,
C     READ FROM THE FILE STATEREV IFPS.
       DO 25 J=1,4
   25  READ(2,15) A
       DO 40 I=1,NYEAR
   40  READ(2,45) (CREDIT(I,J), J=1,NSCAT)
   45  FORMAT(8X,6F11.3)
C
C   * GOAL TARGET VALUES FOR THE STATE REVENUE REQUIREMENT GOALS, BY YEAR
C     READ FROM FILE THE STATEREV IFPS.
       DO 70 J=1,2
   70  READ(2,15) A
       DO 75 I=1,NYEAR
   75  READ(2,80) RQMT(I)
   80  FORMAT(8X,F11.3)
C
C   * SUPPLEMENTAL REVENUE REQUIREMENTS, BY YEAR AND OVERALL, READ
C     FROM THE FILE SUPLMREV IFPS, CONVERTED FROM DOLLARS TO THOUSANDS
C     OF DOLLARS.
```

```
      DO 100 I=1,5
  100 READ(4,15) A
      DO 110 I=1,NYEAR
      J = NYEAR + I
      READ(4,115) VALUE
  110 RQMT(J) = VALUE/1000.
  115 FORMAT(16X,F10.0)
C     ** SUPPLEMENTAL REVENUE REQUIREMENTS, OVERALL
      READ(4,15) A
      J = 2*NYEAR + 1
      READ(4,115) VALUE
      RQMT(J) = VALUE/1000.
C SETUP RHS IN MPS FORMAT
      NUM = 2*NYEAR
      NUM2 = NYEAR*(NAGRG +NINDG*NSCAT +NRCMG +N1ABS +N2ABS +N4ABS) +2
      NUM3 = NUM + 1
      DO 150 K=1,NUM
      KK = K + 1
      IF(KK .GT. 9) GOTO 130
  120 WRITE(5,125) KK, RQMT(K)
  125 FORMAT(4X,'RHS',7X,'ROW1000',I1,3X,F10.3)
      GOTO 150
  130 WRITE(5,135) KK, RQMT(K)
  135 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
  150 CONTINUE
      IF(NUM2 .GT. 99) GOTO 170
  160 WRITE(5,135) NUM2, RQMT(NUM3)
      GOTO 200
  170 WRITE(5,175) NUM2, RQMT(NUM3)
  175 FORMAT(4X,'RHS',7X,'ROW10',I3,3X,F10.3)
C COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
C   STATE REVENUE RQMT DECISION VARIABLES
  200 DO 230 I=1,NYEAR
      DO 230 K=1,NSCAT
      II = I + 1
      IF(II .GT. 9) GOTO 220
  210 WRITE(6,215) K, I, II, CREDIT(I,K)
  215 FORMAT(9X,'X',2I1,2X,'ROW1000',I1,3X,F10.3)
      GOTO 230
  220 WRITE(6,225) K, I, II, CREDIT(I,K)
  225 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
  230 CONTINUE
C   DEVIATION VARIABLES FOR STATE REVENUE RQMTS
      DO 250 I=1,NYEAR
      DO 250 L=1,2
      II = I + 1
  250 WRITE(7,260) DV(L),I,II,VNP(L)
  260 FORMAT(8X,A1,'00',I1,2X,'ROW1000',I1,3X,F10.3)
C SUPPLEMENTAL REVENUE GOALS - YEARLY
      DO 340 I=1,NYEAR
      II = NYEAR + I + 1
      IF(II .GT. 9) GOTO 320
  300 WRITE(7,310) I, II, VNP(1)
  310 FORMAT(8X,'P','00',I1,2X,'ROW1000',I1,3X,F10.3)
      GOTO 340
```

```
  320 WRITE(7,330) I, II, VNP(1)
  330 FORMAT(8X,'P','00',I1,2X,'ROW100',I2,3X,F10.3)
  340 CONTINUE
C DEVIATION VARIABLES FOR THE YEARLY SUPPLEMENTAL REVENUE GOALS
      DO 400 I=1,NYEAR
      II = NYEAR + I
      III = II + 1
      DO 350 L=1,2
      IF( II .EQ. 9) GOTO 360
      IF( II .GT. 9) GOTO 370
  350 WRITE(7,260) DV(L),II,III,VNP(L)
      GOTO 400
  360 WRITE(7,365) DV(L),II,III,VNP(L)
  365 FORMAT(8X,A1,'00',I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 400
  370 WRITE(7,375) DV(L),II,III,VNP(L)
  375 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
  400 CONTINUE
C SUPPLEMENTAL REVENUE GOAL - OVERALL
      IF(NUM2 .GT. 99) GOTO 420
      DO 405 I=1,NYEAR
  405 WRITE(7,410) I, NUM2, VNP(1)
  410 FORMAT(8X,'P00',I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 445
  420 DO 430 I=1,NYEAR
  430 WRITE(7,440) I, NUM2, VNP(1)
  440 FORMAT(8X,'P00',I3,2X,'ROW10',I3,3X,F10.3)
  445 NUM4 = NUM2 - 1
C DEVIATION VARIABLES FOR THE OVERALL SUPPLEMENTAL REVENUE GOAL
      DO 450 L=1,2
      IF(NUM2 .GT. 99) GOTO 470
  450 WRITE(7,460) DV(L),NUM4,NUM2,VNP(L)
  460 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 500
  470 DO 480 L=1,2
  480 WRITE(7,490) DV(L),NUM4,NUM2,VNP(L)
  490 FORMAT(8X,A1,I3,2X,'ROW10',I3,3X,F10.3)
  500 CONTINUE
      STOP
      END
```

```
C
C GENERATES THE CAPITAL PROJECTS INPUT FILE, CAPGEN DSS
C
C WRITE HEADING TITLES
      WRITE(5,50)
   50 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(5,55)
   55 FORMAT(9X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(5,60)
   60 FORMAT(11X,'CAPITAL PROJECTS DATA RELATED TO STUDENT FEES',/)
  101 FORMAT('ATHLETIC ASSOCIATION')
  102 FORMAT('BUS SYSTEM')
  103 FORMAT('STUDENT CENTER')
  104 FORMAT('HEALTH SERVICE')
  105 FORMAT('DORM AND DINING')
C WRITE THE INPUT SECTION WITH DEFAULT VALUES.
      DO 250 I=1,5
      IF(I .EQ. 1) WRITE(5,101)
      IF(I .EQ. 2) WRITE(5,102)
      IF(I .EQ. 3) WRITE(5,103)
      IF(I .EQ. 4) WRITE(5,104)
      IF(I .EQ. 5) WRITE(5,105)
      WRITE(5,110)
  110 FORMAT('    PROJECT IDENTIFIER           XXXXXX  XXXXXX  XXXXXX')
      WRITE(5,120)
  120 FORMAT('    AMORTIZED AMOUNT   (000)     00000.  00000.  00000.')
      WRITE(5,130)
  130 FORMAT('    UNAMORTIZED AMOUNT (000)     00000.  00000.  00000.')
      WRITE(5,140)
  140 FORMAT('    INTEREST RATE                0.0000  0.0000  0.0000')
      WRITE(5,150)
  150 FORMAT('    PROJECT LIFE - YEARS         00.000  00.000  00.000')
      WRITE(5,160)
  160 FORMAT('    PAYMENTS PER YEAR              01      01      01  ')
      WRITE(5,170)
  170 FORMAT('    YEAR OF PROJECT START        1900    1900    1900 ')
  250 CONTINUE
      STOP
      END
```

```
C
C CREATES AN IFPS-READABLE FILE, CAPIFPS IFPS, FROM THE CAPITAL PROJECTS
C   INPUT FILE, CAPGEN DSS, WHENEVER IT IS CHANGED.
C
      INTEGER BYEAR
      DIMENSION CAP(10,10,10), SNAME(2,6),AREA(5)
      DATA SNAME/'AMOR','I   ','UNAM','ORT ','INT ','RATE','PR I','IFE '
     1,'PPY ','    ','STAR','I   '/
C     DATA SNAME/'ABCD','EFGH','IJKL','MNOP','QRST','UVWX','YZ A','BCDE'
C    1,'FGHI','JKLM','NOPQ','RSTU'/
      DATA AREA/'AA','BU','SC','HS','DD'/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
   10 FORMAT(8X,I1)
   11 FORMAT(7X,I2)
   12 FORMAT(8X,I4)
C
C READ PROJECT DATA FROM FILE, CAPGEN DSS INTO ARRAY CAP(I,J,K) WHERE:
C     I = FINANCIAL DATA ITEM FOR PROJECT J IN AREA K    (NFIN)
C     J = PROJECT IDENTIFIER IN FUNCTIONAL AREA K     (NPROJ)
C     K = FUNCTIONAL AREA; I.E. STUDENT CENTER, DORM & DINING    (NAREAS)
C NFIN   = NUMBER OF FINANCIAL DATA ITEMS PER PROJECT
C NPROJ  = NUMBER OF PROJECTS PER FUNCTIONAL AREA
C NAREAS = NUMBER OF FUNCTIONAL AREAS
      NFIN = 6
      NPROJ = 3
      NAREAS = 5
C BYPASS THE TITLE HEADINGS
      DO 50 I=1,4
   50 READ(4,110) A
C READ IN THE DATA FOR EAC PROJECT
      DO 200 K=1,NAREAS
      I = 0
      READ(4,110) A
      READ(4,110) A
  110 FORMAT(A4)
C * AMORTIZED AMOUNT
      I = I + 1
      READ(4,120) (CAP(I,J,K),J=1,NPROJ)
  120 FORMAT(31X,3(F6.0,2X))
C * UNAMORTIZED AMOUNT
      I = I + 1
      READ(4,120) (CAP(I,J,K),J=1,NPROJ)
C * INTEREST RATE
      I = I + 1
      READ(4,130) (CAP(I,J,K),J=1,NPROJ)
  130 FORMAT(31X,3(F6.4,2X))
C * PROJECT LIFE - YEARS
      I = I + 1
```

```
      READ(4,140) (CAP(I,J,K),J=1,NPROJ)
  140 FORMAT(31X,3(F6.3,2X))
C * PAYMENTS PER YEAR
      I = I + 1
      READ(4,150) (CAP(I,J,K),J=1,NPROJ)
  150 FORMAT(33X,3(F2.0,6X))
C * YEAR OF PROJECT START
      I = I + 1
      READ(4,160) (CAP(I,J,K),J=1,NPROJ)
  160 FORMAT(32X,3(F4.0,4X))
C TRANSLATE YEAR OF PROJECT START TO COLUMN OF PROJECT START
      DO 180 J=1,NPROJ
      CAP(I,J,K) = CAP(I,J,K) - BYEAR + 1
  180 CONTINUE
  200 CONTINUE
C
C WRITE THE DATA TO FILE CAPIFPS IFPS
C * WRITE THE HEADING TITLES
      WRITE(5,210)
  210 FORMAT('*',7X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
     *')
      WRITE(5,220)
  220 FORMAT('*',4X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY
     *ANALYSIS')
      WRITE(5,230)
  230 FORMAT('*',10X,'CAPITAL PROJECTS DATA RELATED TO STUDENT FEES')
C
C * WRITE THE DATA STATEMENTS
C
      DO 400 K=1,NAREAS
      DO 400 J=1,NPROJ
      DO 400 I=1,NFIN
      IF((CAP(1,J,K) .EQ. 0.) .AND. (CAP(2,J,K) .EQ. 0.)) GOTO 400
      WRITE(5,350) AREA(K), J, (SNAME(N,I),N=1,2), CAP(I,J,K)
  350 FORMAT(A2,I1,1X,2A4,' = ',F10.4)
  400 CONTINUE
      STOP
      END
```

```
C
C CONVERTS THE CHANGE-IN-TUITION-CHARGES-OVER-TIME GOALS TO MPS FORMAT
C
      INTEGER BYEAR, SYEAR
      REAL MEASUR(10,10)
      DIMENSION HIST(10,10), TARGET(10,10), COEF(10,10), TITLE(10,6)
      DIMENSION ABBR(10,3), VNP(2),DV(2)
      DATA VNP/1.,-1./, DV/'N','P'/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NTNDG
      READ(3,10) NRCMG
      READ(3,10) N1ABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
      READ(3,15) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
      DO 16 I=1,NSCAT
   16 READ(3,17) (TITLE(I,J),J=1,8)
   17 FORMAT(5X,8A4)
      DO 18 I=1,NSCAT
   18 READ(3,19) (ABBR(I,J),J=1,3)
   19 FORMAT(5X,3A4)
C
C READ THE MEASURE OF CHANGE FROM THE PREVIOUS YEAR FROM THE FILE
C   CHGTIME IFPS.
C
C * BYPASS THE TITLES
      DO 20 J=1,6
   20 READ(4,15) A
C * READ THE DATA
      ERROR = 0.0
      DO 25 I=1,NSCAT
   25 READ(4,30) (MEASUR(I,J), J=1,NYEAR)
   30 FORMAT(13X,6(2X,F6.2))
C
C READ IN THE BASE-YEAR TUITION CHARGE, BY STUDENT CATEGORY, FROM THE
C  FILE TUITHIST DSS
C CHECK TO SEE IF THE VALUES ARE NON-ZERO. IF THEY ARE ZERO THEN PRINT
C  AN ERROR MESSAGE.
C
C * BYPASS THE TITLES
      DO 35 J=1,6
```

```
   35 READ(5,15) A
C * READ THE DATA
      INDEX = SYEAR - BYEAR
      DO 40 I =1, NSCAT
   40 READ(5,41) (HIST(I,J),J=1,6)
   41 FORMAT(13X,6(2X,F6.2))
      IF(INDEX .LE. 6) GOTO 50
C * BYPASS THE TITLES
      DO 43 J=1,3
   43 READ(5,15) A
C * READ THE DATA
      INDEX = INDEX - 6
      DO 45 I=1,NSCAT
   45 READ(5,46) (HIST(I,J), J=1,6)
   46 FORMAT(13X,6(2X,F6.2))
C SETUP RHS IN MPS FORMAT
   50 NROW = NYEAR*(NAGRG + NSCAT) + 1
      NDV = NROW - 1
      NUM = NYEAR*NSCAT
      K = NROW
      DO 80 J=1,NYEAR
      DO 80 I=1,NSCAT
      K = K + 1
      TARGET(I,J) = 0.
      IF(J .NE. 1) GOTO 55
      TARGET(I,J) = HIST(I,INDEX)*(MEASUR(I,J)/100. + 1.)
      IF(HIST(I,INDEX) .EQ. 0.0) ERROR = 1.
   55 IF(K .LE. 9) GOTO 70
      WRITE(6,60) K, TARGET(I,J)
   60 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
      GOTO 80
   70 WRITE(6,75) K, TARGET(I,J)
   75 FORMAT(4X,'RHS',7X,'ROW1000',I1,3X,F10.3)
   80 CONTINUE
C ** ERROR TRAP: BASE-YEAR TUITION CHARGE IS ZERO, DISCONTINUE THE
C       CONVERSION OF GOALS TO MPS FORMAT UNTIL CORRECTED BY THE USER.
      IF(ERROR .EQ. 0.0) GO TO 84
      IYEAR = SYEAR - 1
      WRITE(10,81) IYEAR
   81 FORMAT(I4)
      GOTO 500
C COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
   84 DO 85 I=1,NSCAT
      DO 85 J=1,NYEAR
   85 COEF(I,J) = -1.*(MEASUR(I,J)/100. + 1.)
C  85 WRITE(6,*) COEF(I,J)
      DO 300 I=1,NYEAR
      DO 300 J=1,NSCAT
      NROW = NROW + 1
      NDV = NDV + 1
      IF(I .GT. 1) GOTO 130
      COEF(J,I) = 1.
      IF(NROW .GT. 99) GOTO 100
      WRITE(7,90) J,I,NROW, COEF(J,I)
   90 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
```

```
      GOTO 200
  100 WRITE(7,110) J,I,NROW, COEF(J,I)
  110 FORMAT(9X,'X',2I1,2X,'ROW10',I3,3X,F10.3)
      GOTO 200
  130 IF(NROW .GT. 99) GOTO 170
      II = I - 1
      WRITE(7,90) J,II,NROW,COEF(J,I)
      COEF(J,I) = 1.
      WRITE(7,90) J,I,NROW,COEF(J,I)
      GOTO 200
  170 II = I - 1
      WRITE(7,110) J,II,NROW,COEF(J,I)
      COEF(J,I) = 1.
      WRITE(7,110) J,I,NROW,COEF(J,I)
  200 DO 300 L=1,2
      IF(NDV .EQ. 99) GOTO 230
      IF(NDV .GT. 99) GOTO 240
      IF(NDV .EQ. 9) GOTO 250
      WRITE(8,220) DV(L),NDV,NROW,VNP(L)
  220 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 300
  230 WRITE(8,235) DV(L),NDV,NROW,VNP(L)
  235 FORMAT(8X,A1,'0',I2,2X,'ROW10',I3,3X,F10.3)
      GOTO 300
  240 WRITE(8,245) DV(L),NDV,NROW,VNP(L)
  245 FORMAT(8X,A1,I3,2X,'ROW10',I3,3X,F10.3)
      GOTO 300
  250 WRITE(8,255) DV(L),NDV,NROW,VNP(L)
  255 FORMAT(8X,A1,'00',I1,2X,'ROW100',I2,3X,F10.3)
  300 CONTINUE
C
C CREATE A FILE OF THE BASE YEAR TUITION CHARGES
C
      DO 350 I=1,NSCAT
  350 WRITE(9,360) (ABBR(I,J),J=1,3),HIST(I,INDEX)
  360 FORMAT(3A4,F10.2)
  500 STOP
      END
```

```
C
C CREATES THE FILE, COMPGEN.DSS, WHENEVER THE PROBLEM STRUCTURE IS
C   CHANGED.
C LISTS ALL OF THE POSSIBLE COMPARISON GOALS, ABSOLUTE AND RELATIVE.
C
      DIMENSION TITLE(10,8), ABBR(10,3), IZERO(10)
      DATA IZERO/10*0/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP.DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,23) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('     ')
      DO 22 I=1,NSCAT
   22 READ(3,23) (ABBR(I,J),J=1,3)
   23 FORMAT(5X,3A4)
C WRITE HEADING TITLES
      WRITE(4,25)
   25 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(4,26)
   26 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
C
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
C
      WRITE(4,27)
   27 FORMAT(13X,'ALL OF THE POSSIBLE COMPARISON GOALS')
      WRITE(4,20)
C ALL POSSIBLE PAIRWISE COMBINATIONS OF STUD. CAT. FOR RELATIVE GOALS
      WRITE(4,30)
   30 FORMAT('RELATIVE-COMPARISON GOALS')
      DO 40 J=1,NSCAT
      DO 40 I=1,NSCAT
      IF(I.LE.J) GOTO 40
      WRITE(4,35) I,J,(ABBR(I,II),II=1,3),(ABBR(J,JJ),JJ=1,3)
   35 FORMAT('N',4X,2I3,6X,3A4,' / ',3A4,' = VALUE   ')
   40 CONTINUE
```

```
      WRITE(4,20)
C LIST ALL STUDENT CATEORIES FOR SINGLE-CATEGORY ABSOLUTE GOALS
      WRITE(4,45)
   45 FORMAT('SINGLE-CATEGORY ABSOLUTE GOALS')
      DO 50 I=1,NSCAT
   50 WRITE(4,51) I, (ABBR(I,II),II=1,3)
   51 FORMAT('N 1',2X,I3,9X,3A4,15X,' = VALUE')
      WRITE(4,20)
C ALL POSSIBLE PAIRWISE COMBINATIONS OF STUD. CAT. FOR ABSOLUTE GOALS
      WRITE(4,60)
   60 FORMAT('PAIRWISE-COMPARISON ABSOLUTE GOALS')
      DO 100 J=1,NSCAT
      DO 100 I=1,NSCAT
      IF(I.LE.J) GOTO 100
      WRITE(4,75) I,J,(ABBR(I,II),II=1,3),(ABBR(J,JJ),JJ=1,3)
   75 FORMAT('N 2',2X,2I3,6X,3A4,' - ',3A4,' = VALUE  ')
  100 CONTINUE
      WRITE(4,20)
C LIST 10 FOUR-CATEGORY ABSOLUTE GOALS
      WRITE(4,150)
  150 FORMAT('FOUR-CATEGORY ABSOLUTE COMPARISON GOALS:')
      WRITE(4,151)
  151 FORMAT('USING THE NUMBERS THAT CORRESPOND TO THE STUDENT CATEGORIE
     1S LISTED BELOW,')
      WRITE(4,152)
  152 FORMAT('CONSTRUCT FOUR-CATEGORY ABSOLUTE GOALS.  FOR EXAMPLE, TO
     1MAINTAIN THE')
      WRITE(4,153)
  153 FORMAT('SAME DIFFERENTIAL CHARGE BETWEEN RESIDENT  AND NON-RESIDEN
     1T GRADUATE ')
      WRITE(4,154)
  154 FORMAT('STUDENTS, BOTH ON-CAMPUS AND OFF-CAMPUS, ENTER +4 -3 -6 +5
     1.')
      WRITE(4,155)
  155 FORMAT('THIS CORRESPONDS TO (NRES_GR - RES_GR) - (NRES_GR_OFC - RE
     1S_GR_OFC)')
      DO 160 J=1,10
  160 WRITE(4,170)
  170 FORMAT('N 4   +0 -0 -0 +0')
      WRITE(4,20)
      WRITE(4,175)
  175 FORMAT('NUMBER  CATEGORY')
      DO 180 I=1,NSCAT
  180 WRITE(4,190) I, (ABBR(I,II), II=1,3)
  190 FORMAT(2X,I2,4X,3A4)
      STOP
      END
```

```
C
C CONVERTS THE COMPARISON GOALS, ABSOLUTE AND RELATIVE, TO MPS FORMAT
C
      DIMENSION TNA(25,4), VALUE(25)
      DIMENSION VNP(2),DV(2)
      DATA VNP/1.,-1./, DV/'N','P'/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
      READ(3,15) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
C BYPASS HEADINGS
      DO 16 J=1,4
   16 READ(2,15) A
C PROCESS THE RELATIVE GOALS FIRST
      GOTO 605
   20 IF(N1ABS .NE. 0) GOTO 24
      DO 21 J=1,2
   21 READ(2,15) A
      GOTO 200
C
C SINGLE-CATEGORY ABSOLUTE GOALS
C
C * READ THE DATA FROM THE FILE COMPSET TFPS
   24 DO 25 J=1,2
   25 READ(2,15) A
      K=0
      DO 50 J=1,N1ABS
      READ(2,15) A
      K = K + 1
      READ(2,30) TNA(J,1),VALUE(K)
   30 FORMAT(13,40X,F7.0)
      IF(NYEAR .EQ. 1) GOTO 50
      DO 50 I=2,NYEAR
      K = K + 1
      READ(2,40) VALUE(K)
   40 FORMAT(43X,F7.0)
   50 CONTINUE
C * SETUP RHS IN MPS FORMAT
```

```
      NROW = NYEAR*(NAGRG + NINDG*NSCAT +NRCMG) + 1
      NDV = NROW - 1
      NUM = NYEAR*N1ABS
      K = NROW
      DO 60 J=1,NUM
      K = K + 1
      IF(K .GT. 99) GOTO 70
   60 WRITE(4,65) K, VALUE(J)
   65 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
      GOTO 80
   70 WRITE(4,75) K, VALUE(J)
   75 FORMAT(4X,'RHS',7X,'ROW10',I3,3X,F10.3)
C * COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
   80 DO 150 J=1,N1ABS
      DO 150 I=1,NYEAR
      NROW = NROW + 1
      NDV = NDV + 1
      IF(NROW .GT. 99) GOTO 100
      WRITE(5,90) INA(J,1),I,NROW,VNP(1)
   90 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 120
  100 WRITE(5,110) INA(J,1),I,NROW,VNP(1)
  110 FORMAT(9X,'X',2I1,2X,'ROW10',I3,3X,F10.3)
  120 DO 150 L=1,2
      IF(NDV .EQ. 99) GOTO 130
      IF(NDV .GT. 99) GOTO 140
      WRITE(6,125) DV(L),NDV,NROW,VNP(L)
  125 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 150
  130 WRITE(6,135) DV(L),NDV,NROW,VNP(L)
  135 FORMAT(8X,A1,'0',I2,2X,'ROW10',I3,3X,F10.3)
      GOTO 150
  140 WRITE(6,145) DV(L),NDV,NROW,VNP(L)
  145 FORMAT(8X,A1,I3,2X,'ROW10',I3,3X,F10.3)
  150 CONTINUE
  200 IF(N2ABS .NE. 0) GOTO 205
      DO 202 J=1,2
  202 READ(2,15) A
      GOTO 400
C
C PAIRWISE COMPARISON ABSOLUTE GOALS
C
C * READ THE DATA FROM THE FILE COMPSET IFPS
  205 DO 210 J=1,2
  210 READ(2,15) A
      K=0
      DO 250 J=1,N2ABS
      READ(2,15) A
      K = K + 1
      READ(2,230) INA(J,1),INA(J,2),VALUE(K)
  230 FORMAT(2I3,37X,F7.0)
      IF(NYEAR .EQ. 1) GOTO 250
      DO 250 I=2,NYEAR
      K = K + 1
      READ(2,240) VALUE(K)
```

```
  240 FORMAT(43X,F7.0)
  250 CONTINUE
      NROW = NYEAR*(NAGRG + NINDG*NSCAT + NRCMG + N1ABS) + 1
      NDV = NROW - 1
C * SETUP RHS IN MPS FORMAT
      NUM = NYEAR*N2ABS
      K = NROW
      DO 260 J=1,NUM
      K = K + 1
      IF(K .GT. 99) GOTO 270
  260 WRITE(4,65) K, VALUE(J)
      GOTO 280
  270 WRITE(4,75) K, VALUE(J)
C * COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
  280 DO 350 J=1,N2ABS
      DO 350 I=1,NYEAR
      NROW = NROW + 1
      NDV = NDV + 1
      DO 350 L=1,2
      IF(NROW .GT. 99) GOTO 300
      WRITE(5,90) INA(J,L),I,NROW,VNP(L)
      GOTO 310
  300 WRITE(5,110) INA(J,L),I,NROW,VNP(L)
  310 IF(NDV .EQ. 99) GOTO 330
      IF(NDV .GT. 99) GOTO 340
      WRITE(6,125) DV(L),NDV,NROW,VNP(L)
      GOTO 350
  330 WRITE(6,135) DV(L),NDV,NROW,VNP(L)
      GOTO 350
  340 WRITE(6,145) DV(L),NDV,NROW,VNP(L)
  350 CONTINUE
  400 IF(N4ABS .NE. 0) GOTO 405
      GOTO 1000
C
C FOUR-CATEGORY ABSOLUTE GOALS
C
C * READ THE DATA FROM THE FILE COMPSET IFPS
  405 DO 410 J=1,2
  410 READ(2,15) A
      K=0
      DO 450 J=1,N4ABS
      READ(2,15) A
      READ(2,430) INA(J,1),INA(J,2),INA(J,3),INA(J,4)
  430 FORMAT(4I3)
      DO 450 I=1,NYEAR
      K = K + 1
      READ(2,440) VALUE(K)
  440 FORMAT(43X,F7.0)
  450 CONTINUE
      NROW = NYEAR*(NAGRG + NINDG*NSCAT + NRCMG + N1ABS + N2ABS) + 1
      NDV = NROW - 1
C * SETUP RHS IN MPS FORMAT
      NUM = NYEAR*N4ABS
      K = NROW
      DO 460 J=1,NUM
```

```
      K = K + 1
      IF(K .GT. 99) GOTO 470
  460 WRITE(4,65) K, VALUE(J)
      GOTO 480
  470 WRITE(4,75) K, VALUE(J)
C * COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
  480 DO 600 J=1,N4ABS
      DO 600 I=1,NYEAR
      NROW = NROW + 1
      NDV = NDV + 1
      DO 530 L=1,2
      IF(NDV .EQ. 99) GOTO 500
      IF(NDV .GT. 99) GOTO 510
      WRITE(6,125) DV(L),NDV,NROW,VNP(L)
      GO TO 530
  500 WRITE(6,135) DV(L),NDV,NROW,VNP(L)
      GO TO 530
  510 WRITE(6,145) DV(L),NDV,NROW,VNP(L)
  530 CONTINUE
      DO 580 L=1,4
      VAL = 1.0
      IX = INA(J,L)
      IF(INA(J,L) .LT. 0) IX = IABS(INA(J,L))
      IF(INA(J,L) .LT. 0) VAL= -1.0
      IF(NROW .GT. 99) GOTO 550
      WRITE(5,90) IX,I,NROW,VAL
      GOTO 580
  550 WRITE(5,110) IX,I,NROW,VAL
  580 CONTINUE
  600 CONTINUE
      GOTO 1000
  605 IF(NRCMG .NE. 0) GOTO 610
      READ(2,15) A
      GOTO 20
C
C RELATIVE COMPARISON GOALS
C
C * READ THE DATA FROM THE FILE COMPSET DSS
  610 READ(2,15) A
      K=0
      DO 650 J=1,NRCMG
      READ(2,15) A
      K = K + 1
      READ(2,630) INA(J,1),INA(J,2),VALUE(K)
  630 FORMAT(2I3,37X,F7.0)
      IF(NYEAR .EQ. 1) GOTO 650
      DO 650 I=2,NYEAR
      K = K + 1
      READ(2,640) VALUE(K)
  640 FORMAT(43X,F7.0)
  650 CONTINUE
      NROW = NYEAR*(NAGRG + NINDG*NSCAT) + 1
      NDV = NROW - 1
C * SETUP RHS IN MPS FORMAT, RHS = 0
      NUM = NYEAR*NRCMG
```

```
      K = NROW
      ZERO = 0.
      DO 660 J=1,NUM
      K = K + 1
      IF(K .GT. 99) GOTO 670
  660 WRITE(4,65) K, ZERO
      GOTO 680
  670 WRITE(4,75) K, ZERO
C * COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
C     COEFFICIENT OF INA(J,1) = 1
C     COEFFICIENT OF INA(J,2) = - VALUE(K)
  680 KK = 0
      DO 750 J=1,NRCMG
      DO 750 I=1,NYEAR
      KK = KK + 1
      NROW = NROW + 1
      NDV = NDV + 1
      DO 750 L=1,2
      IF(NROW .GT. 99) GOTO 710
      IF(L .EQ. 2) GOTO 705
      WRITE(5,90) INA(J,L),I,NROW,VNP(1)
      GOTO 720
  705 COEF = - VALUE(KK)
      WRITE(5,90) INA (J,L),I,NROW,COEF
      GOTO 720
  710 IF(L .EQ. 2) GOTO 715
      WRITE(5,110) INA(J,L),I,NROW,VNP(1)
      GOTO 720
  715 COEF = - VALUE(KK)
      WRITE(5,110) INA (J,L),I,NROW,COEF
  720 IF(NDV .EQ. 99) GOTO 730
      IF(NDV .GT. 99) GOTO 740
      WRITE(6,125) DV(L),NDV,NROW,VNP(L)
      GOTO 750
  730 WRITE(6,135) DV(L),NDV,NROW,VNP(L)
      GOTO 750
  740 WRITE(6,145) DV(L),NDV,NROW,VNP(L)
  750 CONTINUE
C PROCESS ABSOLUTE GOALS
      GOTO 20
 1000 CONTINUE
      STOP
      END
```

```
C
C GENERATES A FILE, COMPSET DSS, FOR INPUTING THE COMPARISON GOAL TARGET
C   VALUES, ABSOLUTE AND RELATIVE, BASED ON THE USER'S SELECTION OF
C   GOALS IN FILE COMPGEN DSS.
C CALLED WHENEVER THE PROBLEM STRUCTURE IS CHANGED.
C
      INTEGER SYEAR
      DIMENSION TITLE(10,8), ABBR(10,3), INA(50,4), INR(50,2)
      DIMENSION SIGN(4),S(1)
      DATA SIGN/'   ',' + ',' + ',' + '/,S/' - '/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,12) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 21 I=1,NSCAT
   21 READ(3,22) (ABBR(I,J),J=1,3)
   22 FORMAT(5X,3A4)
C WRITE HEADING TITLES
      WRITE(4,24)
   24 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(4,25)
   25 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
      WRITE(4,26)
   26 FORMAT(17X,'COMPARISON GOAL TARGET VALUES',/)
      WRITE(4,28)
   28 FORMAT('RELATIVE COMPARISON GOALS')
      IF(NRCMG .EQ. 0) GOTO 51
      DO 50 J=1,NRCMG
      READ(3,30) INA(J,1),INA(J,2)
   30 FORMAT(5X,2I2)
      J1 = INA(J,1)
      J2 = INA(J,2)
      WRITE(4,35) J1,J2, (ABBR(J1,K),K=1,3),(ABBR(J2,K),K=1,3),SYEAR
```

```
 35 FORMAT(/,2I3,7X,3A4,' / ',3A4,' = 0000000 ',3X,'FOR YEAR ',I4)
    IF(NYEAR .EQ. 1) GOTO 50
    DO 50 I=2,NYEAR
    NYR = SYEAR + I - 1
    WRITE(4,40) NYR
 40 FORMAT(40X,' = 0000000 ',3X,'FOR YEAR ',I4)
 50 CONTINUE
 51 WRITE(4,52)
 52 FORMAT(/,'SINGLE-CATEGORY ABSOLUTE GOALS')
    IF(N1ABG .EQ. 0) GOTO 68
    DO 65 J=1,N1ABG
    READ(3,53) INA(J,1)
 53 FORMAT(5X,I2)
    J1 = INA(J,1)
    WRITE(4,55) J1,(ABBR(J1,K),K=1,3),SYEAR
 55 FORMAT(/,I3,10X,3A4,15X,' = 0000000 ',3X,'FOR YEAR ',I4)
    IF(NYEAR .EQ. 1) GOTO 65
    DO 65 I=2,NYEAR
    NYR = SYEAR + I - 1
    WRITE(4,60) NYR
 60 FORMAT(40X,' = 0000000 ',3X,'FOR YEAR ',I4)
 65 CONTINUE
 68 WRITE(4,70)
 70 FORMAT(/,'PAIRWISE-COMPARISON ABSOLUTE GOALS')
    IF(N2ABG .EQ. 0) GOTO 110
    DO 100 J=1,N2ABG
    READ(3,75) INA(J,1),INA(J,2)
 75 FORMAT(5X,2I2)
    J1 = INA(J,1)
    J2 = INA(J,2)
 80 WRITE(4,85) J1,J2, (ABBR(J1,K),K=1,3),(ABBR(J2,K),K=1,3),SYEAR
 85 FORMAT(/,2I3,7X,3A4,' - ',3A4,' = 0000000 ',3X,'FOR YEAR ',I4)
    IF(NYEAR .EQ. 1) GOTO 100
    DO 100 I=2,NYEAR
    NYR = SYEAR + I - 1
    WRITE(4,90) NYR
 90 FORMAT(40X,' = 0000000 ',3X,'FOR YEAR ',I4)
100 CONTINUE
110 WRITE(4,120)
120 FORMAT(/,'FOUR-CATEGORY ABSOLUTE COMPARISON GOALS')
    IF(N4ABG .EQ. 0) GOTO 200
    DO 200 J=1,N4ABG
    READ(3,130) INA(J,1),INA(J,2),INA(J,3),INA(J,4)
130 FORMAT(5X,4I3)
    J1 = INA(J,1)
    IF(J1 .LT. 0) SIGN(1)=S(1)
    J2 = INA(J,2)
    IF(J2 .LT. 0) SIGN(2)=S(1)
    J3 = INA(J,3)
    IF(J3 .LT. 0) SIGN(3)=S(1)
    J4 = INA(J,4)
    IF(J4 .LT. 0) SIGN(4)=S(1)
    K1 = IABS(J1)
    K2 = IABS(J2)
    K3 = IABS(J3)
```

```
      K4 = IABS(J4)
      WRITE(4,140) J1,J2,J3,J4, SIGN(1),(ABBR(K1,K),K=1,3),SIGN(2),
     1 (ABBR(K2,K),K=1,3),SIGN(3),(ABBR(K3,K),K=1,3),SIGN(4),
     2 (ABBR(K4,K),K=1,3)
  140 FORMAT(/,4I3,A3,3A4,A3,3A4,A3,3A4,A3,3A4)
      WRITE(4,145) SYEAR
  145 FORMAT(40X,' = 0000000 ',3X,'FOR YEAR ',I4)
      IF(NYEAR .EQ. 1) GOTO 200
      DO 200 I=2,NYEAR
      NYR = SYEAR + I - 1
      WRITE(4,190) NYR
  190 FORMAT(40X,' = 0000000 ',3X,'FOR YEAR ',I4)
  200 CONTINUE
      STOP
      END
```

```
C
C GENERATES THE FILE CHGTIME.DSS FOR INPUTTING THE TARGETED CHANGE IN
C    TUITION CHARGES BETWEEN YEARS, BY STUDENT CATEGORY.
C CALLED WHEN THE PROBLEM STRUCTURE CHANGES.
C
      INTEGER BYEAR, SYEAR
      DIMENSION TITLE(10,8), ABBR(10,3), ICHEAD(10)
C READ IN THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP.DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NTNDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,26) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 25 I=1,NSCAT
   25 READ(3,26) (ABBR(I,J),J=1,3)
   26 FORMAT(5X,3A4)
C WRITE THE HEADING TITLES
      WRITE(6,50)
   50 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(6,55)
   55 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *SIS')
      WRITE(6,60)
   60 FORMAT(6X,'TARGETED PERCENTAGE CHANGE IN TUITION OVER PREVIOUS YEA
     *R')
      WRITE(6,20)
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
      DO 70 I=1,NYEAR
      K = I - 1
   70 ICHEAD(I) = SYEAR + K
      WRITE(6,80) (ICHEAD(I),I=1,NYEAR)
   80 FORMAT(12X,6(4X,I4))
      WRITE(6,20)
      DO 90 I=1,NSCAT
   90 WRITE(6,95) (ABBR(I,K),K=1,3)
   95 FORMAT(3A4,6(2X,' 00.00'))
      STOP
      END
```

```
C
C CONVERTS THE GENERAL INDIVIDUAL GOALS TO MPS FORMAT
C
      DIMENSION COEF(10,10), TARGET(10,10)
      DIMENSION VNP(2),DV(2)
      DATA VNP/1.,-1./, DV/'N','P'/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NTNDG
      READ(3,10) NRCMG
      READ(3,10) NTABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
      READ(3,15) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
C SET THE COEFFICIENTS OF THE DECISION VARIABLES EQUAL TO 1.0
      DO 30 I=1,NYEAR
      DO 30 J=1,NSCAT
   30 COEF(I,J) = 1.
C READ IN THE GOAL TARGET VALUES FROM THE FILE GENIND IFPS
C * BYPASS THE TITLES
      DO 40 J=1,6
   40 READ(5,15) A
C * READ THE DATA
      DO 50 I=1,NSCAT
   50 READ(5,55) (TARGET(I,J), J=1,NYEAR)
   55 FORMAT(13X,6(2X,F6.2))
C SETUP RHS IN MPS FORMAT
      NROW = NYEAR*(NAGRG + NSCAT*(NTNDG - 1)) + 1
      NDV = NROW - 1
      NUM = NYEAR*NSCAT
      K = NROW
      DO 80 J=1,NYEAR
      DO 80 I=1,NSCAT
      K = K + 1
      IF(K .LE. 9) GOTO 70
      WRITE(6,60) K, TARGET(I,J)
   60 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
      GOTO 80
   70 WRITE(6,75) K, TARGET(I,J)
   75 FORMAT(4X,'RHS',7X,'ROW1000',I1,3X,F10.3)
   80 CONTINUE
C COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
```

```
      DO 150 I=1,NYEAR
      DO 150 J=1,NSCAT
      NROW = NROW + 1
      NDV = NDV + 1
      IF(NROW .LE. 9) GOTO 100
      WRITE(7,90) J,I,NROW, COEF(I,J)
   90 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 120
  100 WRITE(7,110) J,I,NROW, COEF(I,J)
  110 FORMAT(9X,'X',2I1,2X,'ROW1000',I1,3X,F10.3)
  120 DO 150 L=1,2
      IF(NDV .EQ. 9) GOTO 130
      IF(NDV .LT. 9) GOTO 140
      WRITE(8,125) DV(L),NDV,NROW,VNP(L)
  125 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 150
  130 WRITE(8,135) DV(L),NDV,NROW,VNP(L)
  135 FORMAT(8X,A1,'00',I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 150
  140 WRITE(8,145) DV(L),NDV,NROW,VNP(L)
  145 FORMAT(8X,A1,'00',I1,2X,'ROW1000',I1,3X,F10.3)
  150 CONTINUE
      STOP
      END
```

```
C
C GENERATES THE FILE GENIND ITPS FOR INPUTTING THE GENERAL INDIVIDUAL
C    GOAL TARGETS FOR EACH STUDENT CATEGORY, BY YEAR.
C CALLED WHENEVER THE PROBLEM STRUCTURE IS CHANGED.
C
      INTEGER BYEAR, SYEAR
      DIMENSION TITLE(10,8), ABBR(10,3), ICHEAD(10)
C READ IN THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRGMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,26) ENROLL
      READ(3,11) SPANE
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 25 I=1,NSCAT
   25 READ(3,26) (ABBR(I,J),J=1,3)
   26 FORMAT(5X,3A4)
C WRITE THE HEADING TITLES
      WRITE(6,50)
   50 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(6,55)
   55 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *SIS')
      WRITE(6,60)
   60 FORMAT(6X,'GOAL TARGETS FOR GENERAL INDIVIDUAL GOALS')
      WRITE(6,20)
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
      DO 70 I=1,NYEAR
      K = I - 1
   70 ICHEAD(I) = SYEAR + K
      WRITE(6,80) (ICHEAD(I),I=1,NYEAR)
   80 FORMAT(12X,6(4X,I4))
      WRITE(6,20)
      DO 90 I=1,NSCAT
   90 WRITE(6,95) (ABBR(I,K),K=1,3)
   95 FORMAT(3A4,6(2X,'000.00'))
      STOP
      END
```

```
C
C GENERATES THE FILE PERCOST2 IFPS FOR INPUTTING THE TARGETED PERCENTAGE
C    OF AVERAGE TOTAL COST FOR EACH STUDENT CATEGORY, BY YEAR.
C CALLED WHENEVER THE PROBLEM STRUCTURE IS CHANGED.
C
      INTEGER BYEAR, SYEAR
      DIMENSION TITLE(10,8), ABBR(10,3), ICHEAD(10)
C READ IN THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
C     WRITE(6,*) SYEAR
      READ(3,10) NYEAR
C     WRITE(6,*) NYEAR
      READ(3,10) NSCAT
C     WRITE(6,*) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,26) ENROLL
      READ(3,11) SPANE
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 25 I=1,NSCAT
   25 READ(3,26) (ABBR(I,J),J=1,3)
   26 FORMAT(5X,3A4)
C WRITE TITLES
      WRITE(6,50)
   50 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(6,55)
   55 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *SIS')
      WRITE(6,60)
   60 FORMAT(6X,'GOAL TARGETS FOR PERCENTAGE OF AVG. TOTAL COST GOALS')
      WRITE(6,20)
C WRITE COLUMN TITLES
      DO 70 I=1,NYEAR
      K = I - 1
   70 ICHEAD(I) = SYEAR + K
      WRITE(6,80) (ICHEAD(I),I=1,NYEAR)
   80 FORMAT(12X,6(4X,I4))
      WRITE(6,20)
      DO 90 I=1,NSCAT
   90 WRITE(6,95) (ABBR(I,K),K=1,3)
   95 FORMAT(3A4,6(2X,' 00.00'))
      STOP
      END
```

```
C
C CALCULATES THE PEER INSTITUTION TARGET VALUES AND CONVERTS THE PEER
C   INSTITUTION GOALS TO MPS FORMAT.
C
C MAXIMUM NUMBER OF STUDENT CATEGORIES ALLOWED IS 9
C MAXIMUM NUMBER OF YEARS IN PLANNING HORIZON ALLOWED IS 6
C
C SUBSCRIPT DEFINITION: I IS FOR STUDENT CATEGORIES, (I=1,NSCAT)
C    J IS FOR YEARS, (J=1,NYEAR) AND K IS FOR FEE CATEGORIES, (K=1,NFEE)
C
      INTEGER SYEAR
      DIMENSION CRHR(10), INFO(10,10), CHARGE(10,10), FEES(10,10)
      DIMENSION TOTFEE(10,10), PGOAL(10,10), DD(10,10), DORDIN(10,10)
      DIMENSION VNP(2),DV(2), TITLE(10,8), ABBR(10,3), IYEAR(10)
      DATA VNP/1.,-1./, DV/'N','P'/
C
C NFEE IS THE NUMBER OF FEE CATEGORIES: ATHLETIC, BUS, STUDENT CENTER,
C   HEALTH SERVICE, AND DORM & DINING
      NFEE = 5
C
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
C
      READ(1,12) BYEAR
      READ(1,12) SYEAR
      READ(1,10) NYEAR
      READ(1,10) NSCAT
      READ(1,10) NAGRG
      READ(1,10) NTNDG
      READ(1,10) NRCMG
      READ(1,10) N1ABS
      READ(1,10) N2ABS
      READ(1,10) N4ABS
      READ(1,10) NOVRG
      READ(1,15) ENROLL
      READ(1,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
      DO 25 I=1,NSCAT
   25 READ(1,26) (TITLE(I,J),J=1,8)
   26 FORMAT(6X,8A4)
      DO 30 I=1,NSCAT
   30 READ(1,35) (ABBR(I,J),J=1,3)
   35 FORMAT(6X,3A4)
C FOR EACH STUDENT CATEGORY READ: 1) THE NUMBER OF CREDIT HOURS ON WHICH
C   THE GOAL IS BASED AND 2) THE INDICATORS AS TO WHICH "OTHER CHARGES"
C   ARE TO BE INCLUDED, FROM THE FILE PEERINFO DSS.
      DO 50 I=1,NSCAT
   50 READ(2,55) CRHR(I), (INFO(I,K),K=1,NFEE)
   55 FORMAT(F2.0,7X,5I2)
C  60 WRITE(6,*) CRHR(I), (INFO(I,K),K=1,NFEE)
```

```
C READ THE PEER INSTITUTION CHARGES FOR EACH STUDENT CATEGORY BY YEAR
C   FROM THE FILE PEERCHG IFPS
      DO 110 K=1,4
  110 READ(3,15) A
      DO 120 I=1,NSCAT
  120 READ(3,125) (CHARGE(I,J), J=1,NYEAR)
  125 FORMAT(12X,6(3X,F6.0))
C READ THE "OTHER CHARGES" BY YEAR FROM THE FILE FEESMPS IFPS
      DO 160 M=1,4
  160 READ(4,15) A
      NFEE1 = NFEE - 1
      DO 170 K=1,NFEE1
  170 READ(4,175) (FEES(K,J), J=1,NYEAR)
  175 FORMAT(15X,8F7.0)
      DO 180 L=1,2
  180 READ(4,175) (DD(L,J),J=1,NYEAR)
C CALCULATE THE PEER INSTITUTION GOAL TARGET VALUE FOR EACH STUDENT
C   CATEGORY IN EACH YEAR.
      DO 200 I=1,NSCAT
      DO 200 J=1,NYEAR
  200 TOTFEE(I,J) = 0.
      DO 250 I=1,NSCAT
      DO 250 J=1,NYEAR
      DO 250 K=1,NFEE1
  250 TOTFEE(I,J) = TOTFEE(I,J) + INFO(I,K)*FEES(K,J)
      DO 300 I=1,NSCAT
      DO 300 J=1,NYEAR
      L = 1
      IF(I .GT. 2) L = 2
      DORDIN(I,J) = INFO(I,NFEE)*DD(L,J)
  300 PGOAL(I,J) = CHARGE(I,J) - TOTFEE(I,J) - DORDIN(I,J)
C
C WRITE A SUMMARY REPORT - PEER IFPS
C
C * WRITE THE HEADING TITLES
      WRITE(6,310)
  310 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(6,320)
  320 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(6,330)
  330 FORMAT(16X,'PEER INSTITUTION GOAL TARGET VALUES',/)
C * WRITE THE REPORT
      DO 332 J=1,NYEAR
  332 IYEAR(J) = SYEAR + J - 1
      WRITE(6,335) (IYEAR(J), J=1,NYEAR)
  335 FORMAT('YEAR',16X,6(4X,I4),/)
      DO 400 I=1,NSCAT
      WRITE(6,350) (TITLE(I,J),J=1,8)
  350 FORMAT(/,'****',10X,8A4)
      WRITE(6,360) (CHARGE(I,J),J=1,NYEAR)
  360 FORMAT('EXPECTED PEER CHARGES',6(2X,F6.0))
      WRITE(6,370) (TOTFEE(I,J),J=1,NYEAR)
  370 FORMAT('  LESS FEE CHARGES   ',6(2X,F6.0))
      WRITE(6,380) (DORDIN(I,J),J=1,NYEAR)
```

```
  380 FORMAT('  LESS DORM & DINING ',6(2X,F6.0))
      WRITE(6,390) (PGOAL(I,J), J=1,NYEAR)
  390 FORMAT('PEER INST GOAL TARGET',6(2X,F6.0))
  400 CONTINUE
C
C CONVERT THE PEER INSTITUTION GOALS TO MPS FORMAT
C
C * SETUP RHS IN MPS FORMAT
      NROW = NYEAR*(NAGRG + 2*NSCAT) + 1
      NDV = NROW - 1
      NUM = NYEAR*NSCAT
      K = NROW
      DO 500 J=1,NYEAR
      DO 500 I=1,NSCAT
      K = K + 1
      IF(K .GT. 99) GOTO 540
  500 WRITE(7,520) K, PGOAL(I,J)
  520 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
      GOTO 600
  540 WRITE(7,560) K, PGOAL(I,J)
  560 FORMAT(4X,'RHS',7X,'ROW10',I3,3X,F10.3)
C COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
C
C * DECISION VARIABLES
  600 DO 800 J=1,NYEAR
      DO 800 I=1,NSCAT
      NROW = NROW + 1
      NDV = NDV + 1
      IF(NROW .GT. 99) GOTO 640
      WRITE(8,620) I,J,NROW,CRHR(I)
  620 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 680
  640 WRITE(8,660) I,J,NROW,CRHR(I)
  660 FORMAT(9X,'X',2I1,2X,'ROW10',I3,3X,F10.3)
C * DEVIATION VARIABLES
  680 DO 800 L=1,2
      IF(NDV .EQ. 99) GOTO 720
      IF(NDV .GT. 99) GOTO 760
      WRITE(9,700) DV(L),NDV,NROW,VNP(L)
  700 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 800
  720 WRITE(9,740) DV(L),NDV,NROW,VNP(L)
  740 FORMAT(8X,A1,'0',I2,2X,'ROW10',I3,3X,F10.3)
      GOTO 800
  760 WRITE(9,780) DV(L),NDV,NROW,VNP(L)
  780 FORMAT(8X,A1,I3,2X,'ROW10',I3,3X,F10.3)
  800 CONTINUE
      STOP
      END
```

```
C
C GENERATES THE FILE PEERGEN.IPS, FROM WHICH THE USER PROVIDES CERTAIN
C    INFORMATION USED TO SET THE PEER INSTITUTION GOAL TARGET VALUES.
C
      DIMENSION TITLE(10,8), ABBR(10,3), IZERO(10)
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP.DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,35) INROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 30 I=1,NSCAT
   30 READ(3,35) (ABBR(I,J),J=1,3)
   35 FORMAT(5X,3A4)
C GENERATE THE FILE PEERGEN.DSS
C WRITE THE TITLE HEADINGS
      WRITE(5,40)
   40 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(5,45)
   45 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(5,50)
   50 FORMAT(7X,'INFORMATION REQUIRED TO SET THE PEER INSTITUTION GOALS
     1')
C WRITE THE COLUMN HEADINGS
      WRITE(5,55)
   55 FORMAT(/,25X,'BASIS FOR',7X,'-------- OTHER CHARGES --------')
      WRITE(5,60)
   60 FORMAT(14X,'ANNUAL',4X,'GOAL TARGET',6X,'-------- FEES --------')
      WRITE(5,65)
   65 FORMAT(3X,'STUDENT',4X,'CREDIT',2X,'----------',16X,'STUD',2X
     1'HLTH',5X,'ROOM')
      WRITE(5,70)
   70 FORMAT(2X,'CATEGORY',4X,'HOURS',3X,'SOURCE',2X,'MEASURE',4X,
     1'ATHL',2X,'BUS',3X,'CNTR',2X,'SERV',4X,'& BRD')
      WRITE(5,80)
   80 FORMAT('----------',4X,'-----',3X,'------',2X,'-------',4X,
```

```
     1'----',2X,'---',3X,'----',2X,'----',4X,'-----')
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
      DO 150 J=1,NSCAT
  150 WRITE(5,155) (ABBR(J,K),K=1,3)
  155 FORMAT(3A4,3X,'00.',6X,'DI',7X,'DI',7X,'Y',5X,'Y',6X,'Y',5X,'Y',
     18X,'Y')
C WRITE THE DESCRIPTION OF THE OPTION ABBREVIATIONS
      WRITE(5,200)
  200 FORMAT(/,5X,'DEFINITIONS OF "BASIS FOR GOAL TARGET" OPTIONS:')
      WRITE(5,210)
  210 FORMAT('SOURCE:  LG = LAND-GRANT COLLEGES & UNIVERSITIES')
      WRITE(5,220)
  220 FORMAT(9X,'NC = NCHEMS 1A ISTITUTIONS')
      WRITE(5,230)
  230 FORMAT(9X,'VA = VIRGINIA 4-YEAR COLLEGES & UNIVERSITIES')
      WRITE(5,240)
  240 FORMAT(9X,'DI = DIRECT INPUT')
      WRITE(5,310)
  310 FORMAT('MEASURE: MN = ARITHMETIC MEAN')
      WRITE(5,320)
  320 FORMAT(9X,'01 - 99 = PERCENTILE')
      WRITE(5,340)
  340 FORMAT(9X,'DI = DIRECT INPUT')
      WRITE(5,400)
  400 FORMAT('NOTE: IF DI, DIRECT INPUT, IS SPECIFIED FOR EITHER SOURCE
     1OR MEASURE,')
      WRITE(5,410)
  410 FORMAT(5X, 'THEN IT MUST BE SPECIFIED FOR BOTH')
      STOP
      END
```

```
C
C CREATE THE IFPS MICRO STATEMENTS FOR EACH STUDENT CATEGORY - STUDENT
C   CATEGORY NAME AND STATISTICAL MEASURE - USED IN THE PEER INSTITUTION
C   DATA FILES
C
      DIMENSION TITLE(10,8), ABBR(10,3), PREFIX(10), XMEAS(10)
      DIMENSION SCNAME(3,6)
      DATA SCNAME/'RES ','UG  ','    ','NRES',' UG ','    ','RES ',
     1'GR  ','    ','NRES',' GR ','    ','RES ','GR O','FC  ','NRES',
     2' GR ','OFC '/
C READ PROBLEM STRUCTURE FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,23) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 22 I=1,NSCAT
   22 READ(3,23) (ABBR(I,J),J=1,3)
   23 FORMAT(5X,3A4)
C READ THE PEER INFORMATION FROM THE FILE PEERINFO DSS
      DO 150 I=1,NSCAT
      READ(4,120) MEASUR
  120 FORMAT(7X,I2)
  150 XMEAS(I) = MEASUR
C WRITE THE MICRO STATEMENTS TO THE FILE PEERMCRO DSS
C    1) WRITE STUDENT-CATEGORY NAME
      IFILE = 5
      DO 300 I=1,NSCAT
      WRITE(IFILE,210) (SCNAME(J,I),J=1,3)
  210 FORMAT('MICRO &NAME&',3A4,'&')
C    2) VARIABLES DEFINING THE PERCENTILES OR MEASURE TO BE USED.
      WRITE(IFILE,260) XMEAS(I)
  260 FORMAT('MICRO &PERCENTILE&',F6.2,'&')
  300 CONTINUE
      STOP
      END
```

```
C
C CONVERTS THE PERCENTAGE-OF-AVERAGE-TOTAL-COST GOALS TO MPS FORMAT
C
      DIMENSION COEF(10,10), TARGET(10,10)
      DIMENSION VNP(2),DV(2)
      DATA VNP/1.,-1./, DV/'N','P'/
C READ IN THE PROBLEM STRUCTURE DATA FROM FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRGMG
      READ(3,10) N1ABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
      READ(3,15) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
C
C READ IN COEFFICIENTS OF THE DECISION VARIABLES FROM FILE PERCOST IFPS
C
C BYPASS TITLES
      DO 20 J=1,4
   20 READ(4,15) A
C READ IN DATA
      DO 30 I=1,NYEAR
   30 READ(4,35) (COEF(I,J), J=1,NSCAT)
   35 FORMAT(8X,6F11.4)
C
C READ IN THE GOAL TARGET VALUES FROM FILE PERCOST2 IFPS
C
C BYPASS TITLES
      DO 40 J=1,6
   40 READ(5,15) A
C READ IN DATA
      DO 50 I=1,NSCAT
   50 READ(5,55) (TARGET(I,J), J=1,NYEAR)
   55 FORMAT(13X,6(2X,F6.2))
C SETUP RHS IN MPS FORMAT
      NROW = NYEAR*NAGRG + 1
      NDV = NROW - 1
      NUM = NYEAR*NSCAT
      K = NROW
      DO 80 J=1,NYEAR
      DO 80 I=1,NSCAT
      K = K + 1
```

```
      IF(K .LE. 9) GOTO 70
      WRITE(6,60) K, TARGET(I,J)
   60 FORMAT(4X,'RHS',7X,'ROW100',I2,3X,F10.3)
      GOTO 80
   70 WRITE(6,75) K, TARGET(I,J)
   75 FORMAT(4X,'RHS',7X,'ROW1000',I1,3X,F10.3)
   80 CONTINUE
C COLUMN DATA FOR MPS - DECISION VARIABLES AND DEVIATION VARIABLES
      DO 150 I=1,NYEAR
      DO 150 J=1,NSCAT
      NROW = NROW + 1
      NDV = NDV + 1
      IF(NROW .LE. 9) GOTO 100
      WRITE(7,90) J,I,NROW, COEF(I,J)
   90 FORMAT(9X,'X',2I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 120
  100 WRITE(7,110) J,I,NROW, COEF(I,J)
  110 FORMAT(9X,'X',2I1,2X,'ROW1000',I1,3X,F10.3)
  120 DO 150 L=1,2
      IF(NDV .EQ. 9) GOTO 130
      IF(NDV .LT. 9) GOTO 140
      WRITE(8,125) DV(L),NDV,NROW,VNP(L)
  125 FORMAT(8X,A1,'0',I2,2X,'ROW100',I2,3X,F10.3)
      GOTO 150
  130 WRITE(8,135) DV(L),NDV,NROW,VNP(L)
  135 FORMAT(8X,A1,'00',I1,2X,'ROW100',I2,3X,F10.3)
      GOTO 150
  140 WRITE(8,145) DV(L),NDV,NROW,VNP(L)
  145 FORMAT(8X,A1,'00',I1,2X,'ROW1000',I1,3X,F10.3)
  150 CONTINUE
      STOP
      END
```

```
C
C GENERATES THE FILE PRIOR ITPS USED TO INPUT THE PRIORITY STRUCTURE.
C   CALLED WHEN THE ANY RELATIVE OR ABSOLUTE GOALS ARE ADDED OR
C   DELETED FROM THE MODEL.
C
      DIMENSION TITLE(10,8), ABBR(10,3), TNA(20,4), TNR(50,2)
      DIMENSION SIGN(4),S(1)
      DATA SIGN/'   ',' + ',' + ',' + '/, S/' - '/
C READ THE PROBLEM STRUCTURE FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRGMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,26) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (TITLE(I,J),J=1,8)
   16 FORMAT(5X,8A4)
   20 FORMAT('    ')
      DO 25 I=1,NSCAT
   25 READ(3,26) (ABBR(I,J),J=1,3)
   26 FORMAT(5X,3A4)
C WRITE HEADING TITLES - 11 LINES
      WRITE(4,30)
   30 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(4,31)
   31 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(4,32)
   32 FORMAT(20X,'GOAL PRIORITY STRUCTURE',/)
      WRITE(4,33)
   33 FORMAT(2X,'IF THE GOAL TARGET IS THE MINIMUM ACCEPTABLE VALUE,')
      WRITE(4,34)
   34 FORMAT(6X,'ENTER THE PRIORITY LEVEL IN THE 1ST COLUMN')
      WRITE(4,35)
   35 FORMAT(2X,'IF THE GOAL TARGET IS THE MAXIMUM ACCEPTABLE VALUE,')
      WRITE(4,36)
   36 FORMAT(6X,'ENTER THE PRIORITY LEVEL IN THE 2ND COLUMN',/)
      WRITE(4,37)
   37 FORMAT(40X,'PRIORITY LEVEL',5X,'SUBGOAL')
      WRITE(4,38)
   38 FORMAT(44X,'>=',3X,'<=',8X,'WEIGHTS')
```

```
C
C AGGREGATE GOALS - STATE REVENUE AND SUPPLEMENTAL REVENUE RQMTS
C
      WRITE(4,51)
   51 FORMAT(/,5X,'STATE REVENUE REQUIREMENT              00   00     01
     1.000  01.000')
      WRITE(4,52)
   52 FORMAT(/,5X,'SUPPLEMENTAL REVENUE RQMT. - YEARLY    00   00     01
     1.000  01.000')
C
C INDIVIDUAL GOALS - PERCENTAGE OF AVERAGE TOTAL COST, CHANGE IN TUITION
C    CHARGES OVER TIME, PEER INSTITUTION, GENERAL INDIVIDUAL GOALS
C
      DO 75 J=1,NINDG
      GO TO (60,62,64,66), J
   60 WRITE(4,61)
   61 FORMAT(/,5X,'PERCENTAGE OF AVERAGE TOTAL COST GOALS')
      GO TO 69
   62 WRITE(4,63)
   63 FORMAT(/,5X,'CHANGE OVER TIME GOALS ')
      GO TO 69
   64 WRITE(4,65)
   65 FORMAT(/,5X,'PEER INSTITUTIONS GOALS ')
      GO TO 69
   66 WRITE(4,67)
   67 FORMAT(/,5X,'GENERAL INDIVIDUAL GOALS ')
      GO TO 69
   69 DO 70 I=1,NSCAT
   70 WRITE(4,71) (TITLE(I,K),K=1,8)
   71 FORMAT(8X,8A4,'    00   00     01.000  01.000')
   75 CONTINUE
C
C ABSOLUTE GOALS: SINGLE-CATEGORY, PAIRWISE, AND FOUR-CATEGORY
C
      GOTO 205
   80 WRITE(4,82)
   82 FORMAT(/,5X,'SINGLE-CATEGORY ABSOLUTE GOALS')
      IF(N1ABG .EQ. 0) GOTO 95
      DO 95 J=1,N1ABG
      READ(3,85) INA(J,1)
   85 FORMAT(5X,I2)
      J1 = INA(J,1)
      WRITE(4,90) (TITLE(J1,K),K=1,8)
   90 FORMAT(8X,8A4,'    00   00     01.000  01.000')
   95 CONTINUE
      WRITE(4,99)
   99 FORMAT(/,5X,'PAIRWISE ABSOLUTE GOALS')
      IF(N2ABG .EQ. 0) GOTO 115
      DO 115 J=1,N2ABG
      READ(3,100) INA(J,1), INA(J,2)
  100 FORMAT(5X,2I2)
      J1 = INA(J,1)
      J2 = INA(J,2)
      WRITE(4,110) (ABBR(J1,K),K=1,3), (ABBR(J2,K),K=1,3)
  110 FORMAT(8X,3A4,' VS. ',3A4,7X,'00   00     01.000  01.000')
```

```
  115 CONTINUE
      WRITE(4,120)
  120 FORMAT(/,5X,'FOUR-CATEGORY ABSOLUTE GOALS')
      IF(N4ABG .EQ. 0) GOTO 200
      DO 200 J=1,N4ABG
      READ(3,130) INA(J,1),INA(J,2),INA(J,3),INA(J,4)
  130 FORMAT(5X,4I3)
      J1 = INA(J,1)
      IF(J1 .LT. 0) SIGN(1)=S(1)
      J2 = INA(J,2)
      IF(J2 .LT. 0) SIGN(2)=S(1)
      J3 = INA(J,3)
      IF(J3 .LT. 0) SIGN(3)=S(1)
      J4 = INA(J,4)
      IF(J4 .LT. 0) SIGN(4)=S(1)
      K1 = IABS(J1)
      K2 = IABS(J2)
      K3 = IABS(J3)
      K4 = IABS(J4)
      WRITE(4,140) SIGN(1),(ABBR(K1,K),K=1,3),SIGN(2),
     1 (ABBR(K2,K),K=1,3)
  140 FORMAT(5X,A3,3A4,A3,3A4)
      WRITE(4,150) SIGN(3),(ABBR(K3,K),K=1,3),SIGN(4),
     1 (ABBR(K4,K),K=1,3)
  150 FORMAT(11X,A3,3A4,A3,3A4,3X,'00   00     01.000  01.000')
  200 CONTINUE
      GOTO 249
C
C RELATIVE COMPARISON GOALS
C
  205 WRITE(4,210)
  210 FORMAT(/,5X,'RELATIVE GOALS')
      IF(NRCMG .EQ. 0) GOTO 80
      DO 225 J=1,NRCMG
      READ(3,215) INA(J,1), INA(J,2)
  215 FORMAT(5X,2I2)
      J1 = INA(J,1)
      J2 = INA(J,2)
      WRITE(4,220) (ABBR(J1,K),K=1,3), (ABBR(J2,K),K=1,3)
  220 FORMAT(8X,3A4,' VS. ',3A4,7X,'00   00     01.000  01.000')
  225 CONTINUE
      GOTO 80
  249 DO 250 J=1,NOVRG
  250 WRITE(4,260)
  260 FORMAT(/,5X,'SUPPLEMENTAL REVENUE RQMT. - OVERALL   00   00     01
     1.000  01.000')
      STOP
      END
```

```
C
C CHECKS THE PRIORITY STRUCTURE, THAT WAS ESTABLISHED IN FILE
C   PRIOR TEPS, TO SEE THAT THE PRIORITY LEVELS HAVE BEEN NUMBERED
C   SEQUENTIALLY; I.E. NO PRIORITY LEVELS WERE LEFT OUT.
      DIMENSION IPRSTR(200), ICHK(100)
      DATA ICHK/10*0/
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
   10 FORMAT(8X,I1)
   11 FORMAT(7X,I2)
   12 FORMAT(8X,I4)
C READ THE PRIORITY STRUCTURE FROM THE FILE PRIOROPT DSS
      NPL = 0
      NGOAL = NAGRG + NINDG*NSCAT + N1ABG + N2ABG + N4ABG + NRCMG +NOVRG
C INPUT THE PRIORITY STRUCTURE INTO A VECTOR
      DO 50 J=1,NGOAL
      J1 = 2*(J-1) + 1
      J2 = J1 + 1
   50 READ(4,55) IPRSTR(J1), IPRSTR(J2)
   55 FORMAT(I2,1X,I2)
C DETERMINE THE MAXIMUM PRIORITY LEVEL RECORDED
      NGOAL2 = 2*NGOAL
      DO 100 J=1,NGOAL2
      IF(IPRSTR(J) .LE. NPL) GOTO 100
      NPL = IPRSTR(J)
  100 CONTINUE
C ORDER ALL OF THE PRIORITY LEVELS THAT WERE RECORDED
      DO 150 I=1,NPL
      DO 150 J=1,NGOAL2
      IF(IPRSTR(J) .NE. I) GOTO 150
      ICHK(I) = 1
  150 CONTINUE
C CHECK TO SEE IF ALL PRIORITY LEVELS FROM 1 TO NPL HAVE BEEN INCLUDED
C     IN THE PRIORITY STRUCTURE
      IFLAG = 0
      DO 200 I=1,NPL
      IF(ICHK(I) .EQ. 1) GOTO 200
      IFLAG=1
  200 CONTINUE
      IF(IFLAG .EQ. 0) GOTO 300
      WRITE(5,250)
  250 FORMAT('ERROR')
  300 CONTINUE
      STOP
      END
```

```
C
C CREATES THE 'ROWS' CARDS (MPS FORMAT) USED BY LINDO TO SOLVE THE
C    PROBLEM.  BASED ON THE PROBLEM STRUCTURE.
C
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP.DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NTNDG
      READ(3,10) NRCMG
      READ(3,10) N1ABS
      READ(3,10) N2ABS
      READ(3,10) N4ABS
      READ(3,10) NOVRG
   10 FORMAT(8X,I1)
   11 FORMAT(7X,I2)
   12 FORMAT(8X,I4)
   15 FORMAT(A4)
C SETUP ROWS IN MPS FORMAT
      WRITE(12,50)
   50 FORMAT('NAME    TUITION PROBLEM DATA',/,'ROWS',/,'  N ROW10001')
      NACMG = N1ABS + N2ABS + N4ABS
      NROW = NYEAR*(NAGRG + NTNDG*NSCAT + NRCMG + NACMG) + NOVRG + 1
      DO 100 J=2,NROW
      IF(J .GT. 9) GOTO 70
      WRITE(12,65) J
   65 FORMAT(2X,'E',1X,'ROW1000',I1)
      GOTO 100
   70 IF(J .GT. 99) GOTO 80
      WRITE(12,75) J
   75 FORMAT(2X,'E',1X,'ROW100',I2)
      GOTO 100
   80 WRITE(12,85) J
   85 FORMAT(2X,'E',1X,'ROW10',I3)
  100 CONTINUE
      STOP
      END
```

```
C
C CREATES AN IFPS READABLE FILE, SETUP IFPS, FROM THE FILE SETUP DSS,
C DEFINES THE STRUCTURE OF THE PROBLEM FOR IFPS MODELS AND REPORTS.
C
      INTEGER SYEAR, EYEAR, BYEAR, BASE
      DIMENSION SCNAME(3,6), ABBR(10,3)
      DATA SCNAME/'RES ','UG  ','    ','NRES',' UG ','    ','RES ',
     1'GR  ','    ','NRES',' GR ','    ','RES ','GR O','FC  ','NRES',
     2' GR ','OFC '/
C
C BYEAR = BASE YEAR = EARLIEST YEAR OF HISTORICAL DATA, COLUMN 1 IN
C         THE IFPS DATA FILES AND MODELS.
C SYEAR = STARTING YEAR OF THE ANALYSIS OR PROJECTION.
C NYEAR = NUMBER OF YEARS IN THE ANALYSIS OR PROJECTION.
C EYEAR = ENDING YEAR, LAST YEAR PROJECTED.
C BASE  = CURRENT YEAR, YEAR 0, YEAR PRIOR TO THE FIRST PROJECTED YEAR.
C ISPAN = THE SPAN OF THE MODEL.  NUMBER OF YEARS OF HISTORY + NUMBER
C         OF YEARS OF PROJECTION.
C
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
      READ(3,26) ENROLL
      READ(3,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
      DO 15 I=1,NSCAT
   15 READ(3,16) (ABBR(I,J),J=1,3)
   16 FORMAT(5X,3A4)
      DO 25 I=1,NSCAT
   25 READ(3,26) (ABBR(I,J),J=1,3)
   26 FORMAT(5X,3A4)
C
C THIS ROUTINE SETS UP AN IFPS DATA FILE THAT ...
C   1) STORES THE COLUMNS STATEMENT THAT IS TO BE INCLUDED IN MOST OF
C      THE IFPS REPORTS (VIA MICROS).  BASED ON THE START YEAR AND
C      THE NUMBER OF YEARS IN THE PLANNING HORIZON.
      EYEAR = SYEAR + NYEAR - 1
      WRITE(4,30) SYEAR,EYEAR
   30 FORMAT('MICRO &COLUMNS&',I4,'-',I4,'&')
C
C   2) STORES THE COLUMNS STATEMENT FOR USE BY THE STUDENT CHARGES
```

```
C       REPORTS; INCLUDES YEAR 0 AND THE PROJECTED YEARS.
        NCOL1 = SYEAR - 1
        NCOL2 = NCOL1 + NYEAR
        BASE = SYEAR - BYEAR
        WRITE(4,35) NCOL1, NCOL2
   35 FORMAT('MICRO &TUITION REPORT&',I4,'-',I4,'&')
C
C    3) STORES THE COLUMNS STATEMENT USED BY ALL OF THE IFPS MODELS.
        NCOL2 = BYEAR + ISPAN - 1
        WRITE(4,40) BYEAR, NCOL2
   40 FORMAT('MICRO &MODEL COLUMNS&',I4,'-',I4,'&')
C
C    4) STORES THE NAMES OF THE STUDENT CATEORIES (VIA MICROS)
        NCOL1 = SYEAR - BYEAR + 1
        NCOL2 = EYEAR - BYEAR + 1
        WRITE(4,50) NCOL1,NCOL2
   50 FORMAT('MICRO &DATAFILECOL&',I1,'-',I1,'&')
        DO 60 I=1,NSCAT
   60 WRITE(4,65) I, (SCNAME(J,I),J=1,3)
   65 FORMAT('MICRO &NAME',I1,'&',3A4,'&')
C
C    5) DEFINES THE VARIABLE BASE; BASE IS THE COLUMN NUMBER OF YEAR 0
        WRITE(4,75) BASE
   75 FORMAT('BASE = ',I2)
        STOP
        END
```

```
C
C GENERATES THE FILE SUPLMREV.TPS FOR INPUTTING THE SUPPLEMENTAL
C   REVENUE GOAL TARGETS.
C CALLED WHENEVER THE PROBLEM STRUCTURE IS CHANGED.
C
      INTEGER SYEAR
      DIMENSION TITLE(10,8), ABBR(10,3)
C READ THE PROBLEM STRUCTURE DATA FROM THE FILE SETUP.DSS
      READ(3,12) BYEAR
      READ(3,12) SYEAR
      READ(3,10) NYEAR
      READ(3,10) NSCAT
      READ(3,10) NAGRG
      READ(3,10) NINDG
      READ(3,10) NRCMG
      READ(3,10) N1ABG
      READ(3,10) N2ABG
      READ(3,10) N4ABG
      READ(3,10) NOVRG
   10 FORMAT(8X,I1)
   11 FORMAT(7X,I2)
   12 FORMAT(8X,I4)
  110 FORMAT(6X,I4,6X,'000000000.')
C WRITE THE HEADING TITLES
      WRITE(4,150)
  150 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(4,155)
  155 FORMAT(5X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(4,160)
  160 FORMAT(13X,'SUPPLEMENTAL REVENUE GOAL TARGET VALUES',/)
      WRITE(4,255)
  255 FORMAT(3X,'BY YEAR (IN DOLLARS)')
C WRITE THE INPUT SECTION WITH DEFAULT VALUES
      DO 300 I=1,NYEAR
      K = SYEAR + I - 1
  300 WRITE(4,110) K
      WRITE(4,350)
  350 FORMAT(3X,'OVERALL (IN DOLLARS)')
      WRITE(4,400)
  400 FORMAT(6X,'TOTAL',5X,'000000000.')
      STOP
      END
```

```
C
C SUBROUTINE THAT INTERFACES WITH THE LINEAR PROGRAMMING SYSTEM LINDO
C   TO SOLVE THE TUITION AND FEE ALLOCATION MODEL AS A GOAL PROGRAMMING
C   PROBLEM.
C
C FUNCTIONS:
C   1) INPUT THE PROBLEM TO BE SOLVED
C   2) TRANSLATE THE PRIORITY STRUCTURE PROVIDED BY THE USER INTO AN
C      ACHIEVEMENT FUNCTION THAT IS USED IN THE SOLUTION ROUTINE.
C   3) SOLVE THE MODEL AS A SLGP PROBLEM
C   4) GENERATE OUTPUT REPORTS USED TO ASSESS THE SOLUTION
C
C
      SUBROUTINE USER (IUNIT)
      INTEGER PLEVEL, PRIOR, BYEAR, SYEAR, BASE, YEAR
      DIMENSION NDVPPL(10), IVAR(10,50), AFCOEF(10,50), SNAME(10,3)
      DIMENSION XIND(400,10), IPRSTR(200), WGT(200), TUIT(10,10)
      DIMENSION IYEAR(10), BASEYR(10), SCNAME(3,10), DEVVAR(200,7)
      DIMENSION IPR(200), ACH(3), ACHFCN(50), NITER(50), TRACE(200,10)
C     DATA SCNAME/'RES ','UG  ','    ','NRES',' UG ','    ','RES ',
C    1'GR  ','    ','NRES',' GR ','    ','RES ','GR O','FC  ','NRES',
C    2' GR ','OFC '/
      DATA ACH/'YES ',' NO ',' -  '/
      COMMON LEVEL,ZBAR,EPSLON,MJRIT,MDLIT,LPLIM,MAXIT,ONE
      COMMON JBASIC(240),BHAT(240),AHAT(240),ALPHA(240)
      COMMON IOFREQ,EPCOL,EPRHS,DELTA,ONEDEL,DETERM,NMUL,MAXMUL
      COMMON /VARN/VARNAM(500)
      COMMON/VUBLP/M,N,MPSNAM,INEQLT(240),LASTK,B(240),A(3000)
     *       ,ISROW(3000),JCOLFR(500),LST2,NCOL
      DOUBLE PRECISION DV, VARNAM
      DATA XIND/4000*0./, NDVPPL/10*0/, NITER/50*0/
C
C  ************************************************************
C  *                                                          *
C  * INPUT DATA: 1) PROBLEM STRUCTURE, 2) BASE YEAR TUITION CHARGES,*
C  *      AND 3) PRIORITY STRUCTURE                           *
C  *                                                          *
C  ************************************************************
C READ IN THE PROBLEM STRUCTURE FROM THE FILE SETUP DSS
      READ(10,12) BYEAR
      READ(10,12) SYEAR
      READ(10,10) NYEAR
C     WRITE(IUNIT,*) SYEAR
      READ(10,10) NSCAT
      READ(10,10) NAGRG
      READ(10,10) NINDG
      READ(10,10) NRCMG
      READ(10,10) N1ABG
      READ(10,10) N2ABG
      READ(10,10) N4ABG
      READ(10,10) NOVRG
      READ(10,13) ENROLL
```

```
      READ(10,11) ISPAN
   10 FORMAT(8X,I1)
   11 FORMAT(8X,I2)
   12 FORMAT(8X,I4)
   13 FORMAT(A4)
      DO 20 I=1,7
      DO 20 J=1,7
   20 TUIT(I,J) = 0.0
      DO 50 J=1,2
      DO 50 I=1,NSCAT
   50 READ(10,55) (SNAME(I,K),K=1,3)
   55 FORMAT(6X,3A4)
C READ IN THE BASE-YEAR TUITION CHARGES FROM THE FILE BASEYR DSS.
      DO 80 I=1,NSCAT
   80 READ(11,85) BASEYR(I)
   85 FORMAT(13X,F10.2)
C READ IN THE PRIORITY STRUCTURE AND WEIGHTS FROM FILE PRIOROPT DSS
      NACMG = N1ABG + N2ABG + N4ABG
      NGOAL = NAGRG + NINDG*NSCAT + NACMG + NRCMG + NOVRG
      NSCAT2 = 2*NSCAT
      DO 120 J=1,NGOAL
      J1=2*(J-1) + 1
      J2=J1 + 1
  100 READ(12,101) IPRSTR(J1), IPRSTR(J2), WGT(J1), WGT(J2)
  101 FORMAT(I2,1X,I2,1X,F6.3,1X,F6.3)
      IPR(J1) = IPRSTR(J1)
      IPR(J2) = IPRSTR(J2)
  120 CONTINUE
C     DO 200 J=1,NGOAL
C     J1=2*(J-1) + 1
C     J2=J1 + 1
C 200 WRITE(IUNIT,*) IPRSTR(J1), IPRSTR(J2)
C
C  ********************************************************************
C  *                                                                  *
C  *                                                                  *
C  ********************************************************************
C                        TUITION PROBLEM
C  ********************************************************************
C  *                                                                  *
C  *                                                                  *
C  ********************************************************************
C  ********************************************************************
C  *            PRIORITY LEVEL FOR EACH GOAL IS ESTABLISHED IN        *
C  *             VECTOR IPRSTR(I) WHERE I IS THE GOAL NUMBER.         *
C  * THE VALUES IN IPRSTR(I) WERE ESTABLISHED INTERACTIVELY AND ARE   *
C  *                  STORED IN THE FILE PRIOR IFPS                   *
C  ********************************************************************
C
C  ********************************************************************
C  *    DETERMINE THE NUMBER OF PRIORITY LEVELS                       *
C  ********************************************************************
  400 NGLSTS = 2*NGOAL
      NPL = 0
      DO 405 J1= 1, NGLSTS
```

```
      IF (IPRSTR(J1) .LE. NPL) GO TO 405
      NPL = IPRSTR(J1)
  405 CONTINUE
C
C SKIP PRINTING THE PRIORITY STRUCTURE
C
      GO TO 449
C
C  ****************************************************************
C  *  NAGRG = NUMBER OF AGGREGATE GOALS                           *
C  *  NINDG = NUMBER OF INDIVIDUAL GOALS                          *
C  *  NRCMG = NUMBER OF RELATIVE COMPARATIVE GOALS                *
C  *  NACMG = NUMBER OF ABSOLUTE COMPARATIVE GOALS                *
C  *  NOVRG = NUMBER OF OVERALL GOALS                             *
C  *  NSCAT = NUMBER OF STUDENT CATEGORIES                        *
C  *  NYEAR = NUMBER OF YEARS IN THE PLANNING HORIZON             *
C  ****************************************************************
C
C  ****************************************************************
C  **  PRINT OUT THE PRIORITY STRUCTURE                          *
C  ****************************************************************
C
  406 WRITE(14,407) NPL
  407 FORMAT(/,25X,'*** ',I2,' PRIORITY LEVELS ***')
      WRITE(14,410)
  410 FORMAT(/,18X,'MINIMIZE PJ OR NJ',5X,'INTER-PRIORITY LEVEL')
      WRITE(14,411)
  411 FORMAT(/,18X,'AT PRIORITY LEVEL',9X,'WEIGHTING')
      WRITE(14,412)
  412 FORMAT(/,22X,'NJ',6X,'PJ',11X,'NJ',6X,'PJ')
      DO 414 J = 1, NAGRG
      J1 = 2*(J-1) + 1
      J2 = J1 + 1
  414 WRITE(14,415) J, IPRSTR(J1), IPRSTR(J2), WGT(J1), WGT(J2)
  415 FORMAT(/,'AGGREGATE GOAL ',I2,4X,I2,6X,I2,11X,F4.2,6X,F4.2)
      DO 435 J = 1, NINDG
      WRITE(14,420) J
  420 FORMAT(/,'INDIVIDUAL GOAL ',I2)
      J1 = 2*NAGRG + 2*(J-1)*NSCAT + 1
      J2 = 2*NAGRG + 2*J*NSCAT
      K2 = 1
      DO 430 K = J1, J2, 2
      K1 = K + 1
      WRITE(14,425) (SNAME(K2,M),M=1,3), IPRSTR(K), IPRSTR(K1),
     2WGT(K), WGT(K1)
  425 FORMAT(/,2X,3A4,8X,I2,5X,I2,11X,F4.2,6X,F4.2)
      K2 = K2 + 1
  430 CONTINUE
  435 CONTINUE
      DO 440 J=1,NRCMG
      J1 = 2*NAGRG + 2*NSCAT*NINDG + 2*(J - 1) + 1
      K1 = J1 + 1
  440 WRITE(14,441) J, IPRSTR(J1), IPRSTR(K1), WGT(J1), WGT(K1)
  441 FORMAT(/,'REL. COMP. GOAL ',I2,4X,I2,5X,I2,11X,F4.2,6X,F4.2)
      DO 445 J=1,NACMG
```

```
      J1 = 2*NAGRG + 2*NSCAT*NINDG + 2*NRCMG + 2*(J - 1) + 1
      K1 = J1 + 1
  445 WRITE(14,446) J, IPRSTR(J1), IPRSTR(K1), WGT(J1), WGT(K1)
  446 FORMAT(/,'ABS. COMP. GOAL ',I2,4X,I2,5X,I2,11X,F4.2,6X,F4.2)
      DO 447 J=1,NOVRG
      J1 = 2*(NAGRG + NSCAT*NINDG + NACMG + NRCMG + J - 1) + 1
      K1 = J1 + 1
  447 WRITE(14,448) J, IPRSTR(J1), IPRSTR(K1), WGT(J1), WGT(K1)
  448 FORMAT(/,'OVERALL GOAL ',I2,7X,I2,5X,I2,11X,F4.2,6X,F4.2)
C
C  ****************************************************************
C  *   DETERMINE WHICH GOALS ARE INCLUDED AT EACH PRIORITY LEVEL    *
C  ****************************************************************
C
C  **  IPL IS THE CURRENT PRIORITY LEVEL, NPL IS THE TOTAL NUMBER OF
C        PRIORITY LEVELS.
  449 IPL = 0
  450 IPL = IPL + 1
      IF (IPL .GT. NPL) GO TO 5000
C  **  HAVE INCLUDED ALL GOAL SETS WHEN TRANSFER TO L5000
  500 LOC = 1
C  **  SEARCH STRUCTURE VECTOR
  510 IF(IPRSTR(LOC) .EQ. IPL) GO TO 600
C  **  INCLUDE GOAL SET LOC AT PRIORITY LEVEL IPL
      LOC = LOC + 1
      IF (LOC .GT. NGLSTS) GO TO 450
C  **  TRANSFER TO L450 MEANS ALL GOAL SETS AT PRIORITY LEVEL IPL
C        HAVE BEEN ESTABLISHED, ELSE KEEP SEARCHING THE STRUCTURE
C        VECTOR FOR THE NEXT GOAL SET TO BE INCLUDED AT PRIORITY
C        LEVEL IPL.
      GO TO 510
C
C  ****************************************************************
C  *   PRIOR = NUMBER OF PRIOR GOAL EQUATIONS IN FORMULATION      *
C  *   PLEVEL = PRIORITY LEVEL                                    *
C  *   NGEQN = NUMBER OF GOALS IN THE GOAL SET                    *
C  *                                                              *
C  ****************************************************************
C
  600 PLEVEL = IPL
      IPRSTR(LOC) = 0
      NAGRG2 = 2*NAGRG
      WEIGHT = WGT(LOC)
      IF (LOC .GT. NAGRG2) GO TO 610
C  **  AN AGGREAGTE GOAL
      GO TO 1100
  610 ICHK = 2*NAGRG + 2*NINDG*NSCAT
C  **  AN INDIVIDUAL CATEGORY GOAL
      IF (LOC .GT. ICHK) GO TO 620
      ICHK1 = 2*NAGRG + 2*NSCAT
      IF (LOC .GT. ICHK1) GO TO 612
C  **  PERCENT OF AVERAGE COST GOAL
      IG = 1
      GO TO 1300
  612 ICHK1 = 2*NAGRG + 2*2*NSCAT
```

```
      IF (LOC .GT. ICHK1) GO TO 614
C  **   PERCENT CHANGE BETWEEN YEARS GOAL
      IG = 2
      GO TO 1300
  614 ICHK1 = 2*NAGRG + 3*2*NSCAT
      IF(LOC .GT. ICHK1) GO TO 616
C  **   PEER INSTITUTION GOAL
      IG = 3
      GO TO 1300
  616 ICHK1 = 2*NAGRG + 4*2*NSCAT
      IF (LOC .GT. ICHK1) GO TO 6900
C  **   GENERAL INDIVIDUAL GOAL
      IG = 4
      GO TO 1300
  620 ICHKC1 = 2*NAGRG + 2*NINDG*NSCAT + 2*NRCMG
C  **  A COMPARATIVE GOAL
      ICHKC2 = ICHKC1 + 2*NACMG
      IF(LOC .LE. ICHKC1) GO TO 1400
C  **  A RELATVIE COMPARATIVE GOAL
      IF (LOC .GT. ICHKC2) GO TO 630
      GO TO 1600
C  **  AN ABSOLUTE COMPARATIVE GOAL
  630 ICHKG = ICHKC2 + 2*NOVRG
      IF (LOC .GT. ICHKG) GO TO 6900
      GO TO 1800
C  **  AN OVERALL GOAL
C
C  ***********************************************************************
C  *********  AGGREGATE GOAL SETS ****************************************
C  *     1. STATE REVENUE REQUIREMENTS                                   *
C  *     2. SUPPLEMENTAL REVENUE REQUIREMENTS - YEARLY                   *
C  *  ALL YEARS, FOR EACH CATEGORY, ARE AT THE SAME PRIORITY LEVEL       *
C  *     AND AT SAME WEIGHTING                                           *
C  *                  1 GOAL PER YEAR                                    *
C  ***********************************************************************
C
 1100 PRIOR = 0
      NGLYR = 1
      NGEQN = NYEAR*NGLYR
      NDVPPL(PLEVEL) = NDVPPL(PLEVEL) + NGEQN
      LCODEV = MOD(LOC,2)
      IF(LCODEV .EQ. 0) GO TO 1200
      DO 1150 J=1,NYEAR
      K = NYEAR*(LOC-1) + 2*(J-1) + 1
C     WRITE(14,1149) K, PLEVEL
C1149 FORMAT(/,'K= ',I4,3X,'PLEVEL= ',I2)
 1150 XIND(K,PLEVEL) = WEIGHT
      GO TO 500
 1200 DO 1250 J=1,NYEAR
      K = NYEAR*(LOC-2) + 2*J
C     WRITE(14,1249) K, PLEVEL
C1249 FORMAT(/,'K= ',I4,3X,'PLEVEL= ',I2)
 1250 XIND(K,PLEVEL) = WEIGHT
      GO TO 500
C
```

```
C  ***********************************************************************
C  *********  INDIVIDUAL GOAL SETS  ******************************
C  *       1. TUITION CHARGES AS A PERCENTAGE OF TOTAL COST              *
C  *       2. CHANGE IN TUITION CHARGES OVER TIME                        *
C  *       3. PEER INSTITUTION GOALS                                     *
C  *       4. GENERAL INDIVIDUAL GOALS                                   *
C  *                                                                     *
C  *  ALL YEARS, FOR EACH CATEGORY, ARE AT THE SAME PRIORITY LEVEL       *
C  *     AND AT SAME WEIGHTING                                           *
C  *           1 GOAL PER DECISION VARIABLE                              *
C  * FOR EXAMPLE, SETTING TUITION BASED ON A SPECIFIED PERCENT           *
C  *   AVERAGE TOTAL COST USING A 3-YEAR PLANNING HORIZON WITH           *
C  *   6 STUDENT CATEGORIES AND 2 AGGREGATE GOALS, THEN:
C  *           FOR X11, MIN (N7, N13, N19) OR (P7, P13, P19)             *
C  *             DEC. VAR. NO. (13,25,37) OR (14,26,38)                  *
C  *           FOR X12, MIN (N8, N14, N20) OR (P8, P14, P20)             *
C  *              DEC. VAR. NO. (15,27,39) OR (16,28;40)                 *
C  *                    ETC.                                             *
C  ***********************************************************************
C
1300   PRIOR = NYEAR*(NAGRG + (IG-1)*NSCAT)
       NGLYR = 1
       NGEQN = NYEAR*NGLYR
       NDVPPL(PLEVEL) = NDVPPL(PLEVEL) + NGEQN
       DO 1350 J=1,NYEAR
       K = 2*PRIOR + (LOC - 2*NAGRG - 2*(IG-1)*NSCAT) + 2*(J-1)*NSCAT
C      WRITE(14,1349) K, PLEVEL
C1349  FORMAT(/,'K= ',14,3X,'PLEVEL= ',12)
 1350  XIND(K,PLEVEL) = WEIGHT
       GO TO 500
C
C  ***********************************************************************
C  *********  RELATIVE COMPARISON GOALS    ***************************
C  * ALL YEARS, FOR EACH CATEGORY, ARE AT THE SAME PRIORITY LEVEL       *
C  *     AND AT SAME WEIGHTING                                           *
C  *           1 GOAL PER DECISION VARIABLE                              *
C  ***********************************************************************
C
1400   PRIOR = NYEAR*(NAGRG + NSCAT*NINDG)
       NGLYR = 1
       NGEQN = NYEAR*NGLYR
       NDVPPL(PLEVEL) = NDVPPL(PLEVEL) + NGEQN
       KK = LOC - 2*NAGRG - 2*NINDG*NSCAT
       KODEV = MOD(KK,2)
       IF(KODEV .EQ. 0) GO TO 1500
       DO 1450 J=1,NYEAR
       K = 2*PRIOR + NYEAR*(KK-1) + 2*(J-1) + 1
C      WRITE(14,1449) K, PLEVEL
C1449  FORMAT(/,'K= ',14,3X,'PLEVEL= ',12)
 1450  XIND(K,PLEVEL) = WEIGHT
       GO TO 500
 1500  DO 1550 J=1,NYEAR
       K = 2*PRIOR + NYEAR*(KK-2) + 2*J
C      WRITE(14,1559) K, PLEVEL
C1559  FORMAT(/,'K= ',14,3X,'PLEVEL= ',12)
```

```
 1550 XIND(K,PLEVEL) = WEIGHT
      GO TO 500
C
C  ***************************************************************************
C  *********  ABSOLUTE COMPARISON GOALS   ****************************
C  *      1. SINGLE-CATEGORY                                                 *
C  *      2. PAIRWISE COMPARISONS                                            *
C  *      3. FOUR-CATEGORY                                                   *
C  * ALL YEARS, FOR EACH CATEGORY, ARE AT THE SAME PRIORITY LEVEL     *
C  *      AND AT SAME WEIGHTING                                              *
C  *           1 GOAL PER DECISION VARIABLE                                  *
C  ***************************************************************************
C
1600  PRIOR = NYEAR*(NAGRG + NSCAT*NINDG + NRCMG)
      NGLYR = 1
      NGEQN = NYEAR*NGLYR
      NDVPPL(PLEVEL) = NDVPPL(PLEVEL) + NGEQN
      KK = LOC - 2*NAGRG - 2*NINDG*NSCAT - 2*NRCMG
      KODEV = MOD(KK,2)
      IF(KODEV .EQ. 0) GO TO 1700
      DO 1650 J=1,NYEAR
      K = 2*PRIOR + NYEAR*(KK-1) + 2*(J-1) + 1
C     WRITE(14,1649) K, PLEVEL
C1649 FORMAT(/,'K= ',I4,3X,'PLEVEL= ',I2)
 1650 XIND(K,PLEVEL) = WEIGHT
      GO TO 500
 1700 DO 1750 J=1,NYEAR
      K = 2*PRIOR + NYEAR*(KK - 2) + 2*J
C     WRITE(14,1749) K, PLEVEL
C1749 FORMAT(/,'K= ',I4,3X,'PLEVEL= ',I2)
 1750 XIND(K,PLEVEL) = WEIGHT
      GO TO 500
C
C  ***************************************************************************
C  ********************  OVERALL  GOALS  ****************************
C  *            1 GOAL PER FORMULATION                                       *
C  ***************************************************************************
C
1800  PRIOR = NYEAR*(NAGRG + NSCAT*NINDG + NACMG + NRCMG)
      NGEQN = 1
      NDVPPL(PLEVEL) = NDVPPL(PLEVEL) + NGEQN
      KK = LOC - 2*(NAGRG + NINDG*NSCAT + NACMG + NRCMG)
      K = 2*PRIOR + KK
C     WRITE(14,1849) K, PLEVEL
C1849 FORMAT(/,'K= ',I4,3X,'PLEVEL= ',I2)
      XIND(K,PLEVEL) = WEIGHT
      GO TO 500
C  *                                                                         *
C  ***************************************************************************
C  *      ***** SET UP ACHIEVEMENT FUNCTION *****                            *
C  ***************************************************************************
C  *                                                                         *
 5000 NDV = 0
      DO 5050 I = 1, NPL
 5050 NDV = NDV + NDVPPL(I)
```

```
      DO 5100 I=1,NPL
      K = 0
      DO 5100 J = 1, 200
      IF(XIND(J,I) .EQ. 0.) GO TO 5100
      K = K + 1
      IVAR(I,K) = J
      AFCOEF(I,K) = XIND(J,I)
      XIND(J,I) = 0.
 5100 CONTINUE
C
C  ********************************************************************
C  *                                                                  *
C  *    SOLVE GOAL PROGRAMMING PROBLEM SEQUENTIALLY USING LP ROUTINE  *
C  *                                                                  *
C  *      PRIOR = TOTAL NUMBER OF DEVIATION VARIABLES                 *
C  ********************************************************************
C
      PRIOR = 2*NYEAR*(NAGRG + NINDG*NSCAT + NACMG + NRCMG) + 2*NOVRG
 5500 DO 5800 I=1,NPL
      K=NDVPPL(I)
      DO 5700 J=1,K
      NUM=IVAR(I,J)
      VAL=AFCOEF(I,J)
 5700 CALL INSERT(1,NUM,VAL,1)
      CALL LOOK(1,1)
      CALL GO(0,KONDN)
      CALL REPROW(1,PRIMAL,DUAL)
      ACHFCN(I) = PRIMAL
      NITER(I) = KONDN
C CREATE AN ARRAY, TRACE(L,M), OF THE DECISION VARIABLE'S VALUE AT
C  EACH PRIORITY LEVEL (AT EACH SOLUTION SEQUENCE)
      DO 5725 NY=1,NYEAR
      DO 5725 NS=1,NSCAT
      IX = PRIOR + (NY-1)*NSCAT + NS
      CALL REPVAR(IX,PRIMAL,DUAL)
      L = (NY - 1)*NPL + I
      TRACE(L,NS) = PRIMAL
 5725 CONTINUE
C SET NEW UPPER BOUND, FOR ALL DEVIATION VARIABLES ADDED AT THIS
C  PRIORITY LEVEL, TO ITS PRIMAL SOLUTION VALUE.
      DO 5750 J=1,K
      NUM=IVAR(I,J)
      CALL REPVAR(NUM,XPRIM,DUAL)
 5750 CALL SETSUB(NUM,XPRIM)
 5800 CONTINUE
C
C  ********************************************************************
C  *                                                                  *
C  *         PRINT FINAL RESULTS OF OPTIMIZATION                      *
C  *                                                                  *
C  *         PRINT DECISION VARIABLE VALUES FOR EACH YEAR             *
C  *                   TO THE FILE TUIT IFPS A                        *
C  ********************************************************************
C
      MM = 15
```

```
      WRITE(MM,5910)
 5910 FORMAT(8X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY')
      WRITE(MM,5915)
 5915 FORMAT(7X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS')
      WRITE(MM,5917)
 5917 FORMAT(5X,'RECOMMENDED TUITION CHARGES, IN DOLLARS PER CREDIT HOUR
     *')
      JYEAR = NYEAR + 1
      DO 5920 I=1,JYEAR
      K = I - 2
 5920 IYEAR(I) = SYEAR + K
      WRITE(MM,5930) (IYEAR(I), I=1,JYEAR)
 5930 FORMAT(/,13X,5(2X,I4,2X))
      DO 5940 J=1,NSCAT
 5940 TUIT(J,1) = BASEYR(J)
      DO 5950 J=1,NSCAT
      DO 5950 I=2,JYEAR
      NNUM = PRIOR + NSCAT*(I-2) + J
      CALL REPVAR(NNUM, XPRIM, DUAL)
 5950 TUIT(J,I) = XPRIM
      DO 5990 J=1,NSCAT
      WRITE(MM,5980) (SNAME(J,K),K=1,3), (TUIT(J,L),L=1,JYEAR)
 5980 FORMAT(/,3A4,7(2X,F6.2))
 5990 CONTINUE
C  ********************************************************************
C  *         SETUP AN ARRAY, DEVVAR(I,J), CONTAINING IN EACH ROW      *
C  *           THE FOLLOWING INFORMATION ON EACH GOAL:                *
C  *                                                                  *
C  *      1. WHETHER OR NOT THE GOAL WAS ACHIEVED (COL. 7)            *
C  *      2. PRIORITY LEVEL FOR EACH DEVIATION VARIABLE (COL 1, COL.2)*
C  *      3. DEVIATION VARIABLE VALUES: NJ AND PJ (COL. 3 AND COL. 4) *
C  *      4. THE TARGET GOAL VALUE (RHS) (COL. 5)                     *
C  *      5. ACTUAL GOAL VALUE = RHS -NJ - PJ (COL. 6)                *
C  ********************************************************************
C     NAGRGDV = 2*NYEAR*NAGRG
C     NINDGDV = NAGRGDV + 2*NYEAR*NINDG*NSCAT
C     NACMGDV = NINDGDV + 2*NYEAR*NACMG
C ***** AGGREGATE GOALS
      DO 6200 M=1,NAGRG
      DO 6200 I=1,NYEAR
C GOAL TARGET VALUE
      K = NYEAR*(M-1) + I
      KK = K + 1
      DEVVAR(K,5) = B(KK)
C VALUE OF THE DEVIATION VARIABLES, NJ AND PJ
      J = 2*NYEAR*(M - 1) + 2*(I - 1) + 1
      J1 = J + 1
      CALL REPVAR(J, XPRM1, DUAL1)
      CALL REPVAR(J1, XPRM2, DUAL2)
      DEVVAR(K,3) = XPRM1
      DEVVAR(K,4) = XPRM2
C ACTUAL GOAL VALUE
      DEVVAR(K,6) = DEVVAR(K,5) - DEVVAR(K,3) + DEVVAR(K,4)
C PRIORITY LEVEL FOR EACH DEVIATION VARIABLE
```

```
      M1 = 2*M - 1
      M2 = M1 + 1
      DEVVAR(K,1) = IPR(M1)
      DEVVAR(K,2) = IPR(M2)
C DETERMINE IF GOAL IS ACHIEVED OR NOT
      DEVVAR(K,7) = ACH(1)
      IF(DEVVAR(K,1) .EQ. 0 .AND. DEVVAR(K,2) .EQ. 0)
     1DEVVAR(K,7) = ACH(3)
      IF(DEVVAR(K,1) .GT. 0 .AND. DEVVAR(K,3) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
      IF(DEVVAR(K,2) .GT. 0 .AND. DEVVAR(K,4) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
 6200 CONTINUE
C ***** INDIVIDUAL GOALS
      DO 6250 M1=1,NINDG
      DO 6250 M2=1,NYEAR
      DO 6250 M3=1,NSCAT
C GOAL TARGET VALUE
      K = NYEAR*NAGRG + NYEAR*NSCAT*(M1-1) + NSCAT*(M2-1) + M3
      KK = K + 1
      DEVVAR(K,5) = B(KK)
C VALUE OF THE DEVIATION VARIABLES NJ AND PJ
      J=2*NYEAR*NAGRG +2*NYEAR*NSCAT*(M1-1) +2*NSCAT*(M2-1) +2*(M3-1) +1
      J1 = J + 1
      CALL REPVAR(J,XPRM1,DUAL1)
      CALL REPVAR(J1,XPRM2,DUAL2)
      DEVVAR(K,3) = XPRM1
      DEVVAR(K,4) = XPRM2
C ACTUAL GOAL VALUE
      DEVVAR(K,6) = DEVVAR(K,5) - DEVVAR(K,3) + DEVVAR(K,4)
C PRIORITY LEVEL FOR EACH DEVIATION VARIABLE
      M = 2*NAGRG + 2*(M1 - 1)*NSCAT + 2*M3 - 1
      MM = M + 1
      DEVVAR(K,1) = IPR(M)
      DEVVAR(K,2) = IPR(MM)
C DETERMINE IF GOAL IS ACHIEVED OR NOT
      DEVVAR(K,7) = ACH(1)
      IF(DEVVAR(K,1) .EQ. 0 .AND. DEVVAR(K,2) .EQ. 0)
     1DEVVAR(K,7) = ACH(3)
      IF(DEVVAR(K,1) .GT. 0 .AND. DEVVAR(K,3) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
      IF(DEVVAR(K,2) .GT. 0 .AND. DEVVAR(K,4) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
 6250 CONTINUE
C ***** RELATIVE COMPARISON GOALS
      DO 6300 M1=1,NRCMG
      DO 6300 I=1,NYEAR
C GOAL TARGET VALUE
      K = NYEAR*(NAGRG + NSCAT*NINDG + M1 - 1) + I
      KK = K + 1
      DEVVAR(K,5) = B(KK)
C VALUE OF THE DEVIATION VARIABLES NJ AND PJ
      J = 2*NYEAR*(NAGRG +NSCAT*NINDG + M1 - 1) + 2*(I - 1) + 1
      J1 = J + 1
      CALL REPVAR(J, XPRM1, DUAL1)
```

```
      CALL REPVAR(J1, XPRM2, DUAL2)
      DEVVAR(K,3) = XPRM1
      DEVVAR(K,4) = XPRM2
C ACTUAL GOAL VALUE
      DEVVAR(K,6) = DEVVAR(K,5) - DEVVAR(K,3) + DEVVAR(K,4)
C PRIORITY LEVEL FOR EACH DEVIATION VARIABLE
      M = 2*NAGRG + 2*NSCAT*NINDG + 2*(M1 - 1) + 1
      MM = M + 1
      DEVVAR(K,1) = IPR(M)
      DEVVAR(K,2) = IPR(MM)
C DETERMINE IF GOAL IS ACHIEVED OR NOT
      DEVVAR(K,7) = ACH(1)
      IF(DEVVAR(K,1) .EQ. 0 .AND. DEVVAR(K,2) .EQ. 0)
     1DEVVAR(K,7) = ACH(3)
      IF(DEVVAR(K,1) .GT. 0 .AND. DEVVAR(K,3) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
      IF(DEVVAR(K,2) .GT. 0 .AND. DEVVAR(K,4) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
 6300 CONTINUE
C ***** ABSOLUTE COMPARISON GOALS
      DO 6350 M1=1,NACMG
      DO 6350 I=1,NYEAR
C GOAL TARGET VALUE
      K = NYEAR*(NAGRG + NSCAT*NINDG + NRCMG + M1 - 1) + I
      KK = K + 1
      DEVVAR(K,5) = B(KK)
C VALUE OF THE DEVIATION VARIABLE NJ AND PJ
      J = 2*NYEAR*(NAGRG +NSCAT*NINDG + NRCMG + M1 - 1) + 2*(I - 1) + 1
      J1 = J + 1
      CALL REPVAR(J, XPRM1, DUAL1)
      CALL REPVAR(J1, XPRM2, DUAL2)
      DEVVAR(K,3) = XPRM1
      DEVVAR(K,4) = XPRM2
C ACTUAL GOAL VALUE
      DEVVAR(K,6) = DEVVAR(K,5) - DEVVAR(K,3) - DEVVAR(K,4)
C PRIORITY LEVEL FOR EACH DEVIATION VARIABLE
      M = 2*NAGRG + 2*NSCAT*NINDG + 2*NRCMG + 2*(M1 - 1) + 1
      MM = M + 1
      DEVVAR(K,1) = IPR(M)
      DEVVAR(K,2) = IPR(MM)
C DETERMINE IF GOAL IS ACHIEVED OR NOT
      DEVVAR(K,7) = ACH(1)
      IF(DEVVAR(K,1) .EQ. 0 .AND. DEVVAR(K,2) .EQ. 0)
     1DEVVAR(K,7) = ACH(3)
      IF(DEVVAR(K,1) .GT. 0 .AND. DEVVAR(K,3) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
      IF(DEVVAR(K,2) .GT. 0 .AND. DEVVAR(K,4) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
 6350 CONTINUE
C ***** OVERALL GOALS
      DO 6400 I=1,NOVRG
C GOAL TARGET VALUE
      K = NYEAR*(NAGRG + NSCAT*NINDG + NACMG + NRCMG) + I
      KK = K + 1
      DEVVAR(K,5) = B(KK)
```

```
C VALUE OF THE DEVIATION VARIABLES NJ AND PJ
      J = 2*NYEAR*(NAGRG + NSCAT*NINDG + NACMG + NRCMG) + 2*(I - 1) + 1
      J1 = J + 1
      CALL REPVAR(J, XPRM1, DUAL1)
      CALL REPVAR(J1, XPRM2, DUAL2)
      DEVVAR(K,3) = XPRM1
      DEVVAR(K,4) = XPRM2
C ACTUAL GOAL VALUE
      DEVVAR(K,6) = DEVVAR(K,5) - DEVVAR(K,3) + DEVVAR(K,4)
C PRIORITY LEVEL FOR EACH DEVIATION VARIABLE
      M = 2*NAGRG + 2*NSCAT*NINDG + 2*NRCMG + 2*NACMG + 2*(I - 1) + 1
      MM = M + 1
      DEVVAR(K,1) = IPR(M)
      DEVVAR(K,2) = IPR(MM)
C DETERMINE IF GOAL IS ACHIEVED OR NOT
      DEVVAR(K,7) = ACH(1)
      IF(DEVVAR(K,1) .EQ. 0 .AND. DEVVAR(K,2) .EQ. 0)
     1DEVVAR(K,7) = ACH(3)
      IF(DEVVAR(K,1) .GT. 0 .AND. DEVVAR(K,3) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
      IF(DEVVAR(K,2) .GT. 0 .AND. DEVVAR(K,4) .NE. 0)
     1DEVVAR(K,7) = ACH(2)
 6400 CONTINUE
C
C  ***********************************************************************
C  * CONVERT THE ARRAY DEVVAR(I,J) INTO A REPORT ON GOAL ACHIEVEMENTS *
C  ***********************************************************************
C
C HEADING TITLES
      WRITE(14,5910)
      WRITE(14,6515)
 6515 FORMAT(7X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANAL
     *YSIS',/)
      WRITE(14,6520)
 6520 FORMAT(25X,'SUMMARY OF GOAL ACHIEVEMENTS',/)
C COLUMN HEADINGS
      WRITE(14,6530)
 6530 FORMAT(18X,'PRIORITY')
      WRITE(14,6535)
 6535 FORMAT('GOAL',14X,'LEVEL OF',6X,'VALUE OF THE',9X,'GOAL',8X,
     1'GOAL')
      WRITE(14,6540)
 6540 FORMAT('ACHVD',4X,'GOAL',5X,'DEV VAR',3X,'DEVIATION VARIABLES',
     14X,'TARGET',7X,'LEVEL')
      WRITE(14,6545)
 6545 FORMAT('  ?  ',1X,'DEFINITION',4X,'N   P',8X,'N       P',11X,
     1'VALUE',5X,'ATTAINED')
      KOUNT = 0
C AGGREGATE GOALS
      DO 6560 I=1,NAGRG
      IF (I .EQ. 1) WRITE(14,6651)
      IF (I .EQ. 2) WRITE(14,6652)
      DO 6560 J=1,NYEAR
      YEAR = SYEAR + J - 1
      KOUNT = KOUNT + 1
```

```
      WRITE(14,6555) DEVVAR(KOUNT,7), YEAR, (DEVVAR(KOUNT,K), K=1,6)
 6555 FORMAT(/,A4,'YEAR: ',I4,3X,2(1X,F3.0),3X,2(1X,F7.2),4X,2(3X,F8.2))
 6560 CONTINUE
C INDIVIDUAL GOALS
      DO 6580 I=1,NINDG
      IF (I .EQ. 1) WRITE(14,6661)
      IF (I .EQ. 2) WRITE(14,6662)
      IF (I .EQ. 3) WRITE(14,6663)
      IF (I .EQ. 4) WRITE(14,6664)
      DO 6580 J=1,NYEAR
      YEAR = SYEAR + J - 1
      WRITE(14,6570) YEAR
 6570 FORMAT(/,5X,'YEAR: ',I4)
      DO 6580 K=1,NSCAT
      KOUNT = KOUNT + 1
      WRITE(14,6575) DEVVAR(KOUNT,7),(SNAME(K,L),L=1,3),
     1 (DEVVAR(KOUNT,M),M=1,6)
 6575 FORMAT(/,A4,2X,3A4,2(1X,F3.0),3X,2(1X,F7.2),4X,2(3X,F8.2))
 6580 CONTINUE
C RELATIVE COMPARISON GOALS
      DO 6590 I=1,NRCMG
      WRITE(14,6585) I
 6585 FORMAT(/,'RELATIVE COMPARISON GOAL',I3)
      DO 6590 J=1,NYEAR
      YEAR = SYEAR + J - 1
      KOUNT = KOUNT + 1
      WRITE(14,6555) DEVVAR(KOUNT,7), YEAR, (DEVVAR(KOUNT,K), K=1,6)
 6590 CONTINUE
C ABSOLUTE COMPARISON GOALS
      DO 6610 I=1,NACMG
      WRITE(14,6605) I
 6605 FORMAT(/,'ABSOLUTE COMPARISON GOAL',I3)
      DO 6610 J=1,NYEAR
      YEAR = SYEAR + J - 1
      KOUNT = KOUNT + 1
      WRITE(14,6555) DEVVAR(KOUNT,7), YEAR, (DEVVAR(KOUNT,K), K=1,6)
 6610 CONTINUE
C OVERALL GOALS
      DO 6620 I=1,NOVRG
      IF (I .EQ. 1) WRITE(14,6691)
      KOUNT = KOUNT + 1
      WRITE(14,6616) DEVVAR(KOUNT,7), (DEVVAR(KOUNT,K), K=1,6)
 6616 FORMAT(/,A4,13X,2(1X,F3.0),3X,2(1X,F7.2),4X,2(3X,F8.2))
 6620 CONTINUE
C
C PRINT ACHIEVEMENT FUNCTION VALUE AT EACH PRIORITY LEVEL
C
C     WRITE(14,6625)
C6625 FORMAT(////,'ACHIEVEMENT FUNCTION VALUE AT EACH PRIORITY LEVEL')
C     WRITE(14,6626)
C6626 FORMAT('-------------------------------------------------',/)
C     WRITE(14,6627)
C6627 FORMAT(13X,'ACHIEVEMENT   NUMBER')
C     WRITE(14,6628)
C6628 FORMAT('PRIORITY      FUNCTION       OF')
```

```
C     WRITE(14,6629)
C6629 FORMAT('  LEVEL         VALUE      ITERATIONS')
C     DO 6630 I=1,NPL
C6630 WRITE(14,6635) I,ACHFCN(I),NITER(I)
C6635 FORMAT(2X,I3,5X,F15.7,3X,I4)
 6651 FORMAT(/,'STATE REVENUE REQUIREMENT')
 6652 FORMAT(/,'SUPPLEMENTAL REVENUE REQUIREMENT-YEARLY')
 6661 FORMAT(/,'PERCENTAGE OF AVERAGE TOTAL COST')
 6662 FORMAT(/,'CHANGE IN TUITION CHARGES OVER TIME')
 6663 FORMAT(/,'RELATIONSHIP TO PEER INSTITUTIONS')
 6664 FORMAT(/,'GENERAL INDIVIDUAL GOALS')
 6691 FORMAT(/,'SUPPLEMENTAL REVENUE REQUIREMENT-OVERALL')
C
C PRINT THE ARRAY TRACE(L,M)
C
      WRITE(17,5910)
      WRITE(17,6515)
      WRITE(17,6710)
 6710 FORMAT(/,'SOLUTION VALUES FOR THE DECISION VARIABLES AT EACH PRIOR
     1ITY LEVEL')
      WRITE(17,6715)
 6715 FORMAT(/,4X,'VALUES AT EACH STEP IN THE SEQUENTIAL LINEAR PROGRAMM
     1ING')
      WRITE(17,6716)
 6716 FORMAT(/,10X,'SOLUTION TO THE GOAL PROGRAMMING PROBLEM')
      WRITE(17,6720)
 6720 FORMAT(//,1X,'PRIOR',3X,3('RES',6X,'NRES',5X))
      WRITE(17,6725)
 6725 FORMAT(1X,'LEVEL',4X,'UG',7X,'UG',7X,'GR',7X,'GR',5X,'GR-OFC',
     13X,'GR-OFC',/)
      DO 6750 I=1,NYEAR
      WRITE(17,6730) I
 6730 FORMAT('YEAR',I2)
      DO 6750 J=1,NPL
      L = I*J
      WRITE(17,6740) J,(TRACE(L,M),M=1,NSCAT)
 6740 FORMAT(2X,I2,8F9.2)
 6750 CONTINUE
C ***************************************************************************
C                                                                            *
C CREATE A FILE OF THE RECOMMENDED TUITION CHARGES, FOR USE BY IFPS,         *
C   TO A GENERATE MANAGEMENT REPORT.                                         *
C                                                                            *
C ***************************************************************************
      WRITE(16,6810)
 6810 FORMAT('*',7X,'VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
     1')
      WRITE(16,6820)
 6820 FORMAT('*',6X,'DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY
     *ANALYSIS')
      WRITE(16,6830)
 6830 FORMAT('*',4X,'RECOMMENDED TUITION CHARGES, IN DOLLARS PER CREDIT
     1 HOUR')
      BASE = IYEAR(1) - BYEAR + 1
      NNYEAR = NYEAR + 1
```

```
      NZYEAR = SYEAR - BYEAR - 1
      DO 6850 I=1,NSCAT
 6850 WRITE(16,6855) I, NZYEAR,(TUIT(I,K),K=1,7)
 6855 FORMAT('ST CAT ',I1,' = 0 FOR',I2,7(',',F6.2))
      GO TO 7000
C  ***************************************************************
C  *        ERROR MESSAGES                                       *
C  ***************************************************************
 6900 WRITE(14,6910)
 6910 FORMAT(/, 'YOU GOOFED SOMEWHERE, TRY AGAIN')
 7000 RETURN
      END
      SUBROUTINE ALLEN
      WRITE(14,10)
   10 FORMAT(/,'GOOD-BYE')
      RETURN
      END
```

Appendix D

IFPS-RELATED PROGRAM LISTINGS

Programs that link IFPS and the DSS
D.1.1 Linkage Program IFPSDSS
D.1.2 Linkage Program QUIT

IFPS Command Files
D.2.1 IFPS Command File COSTCOM
D.2.2 IFPS Command File DDFEECOM
D.2.3 IFPS Command File ENRLCRED
D.2.4 IFPS Command File PEERCOM
D.2.5 IFPS Command File TUITCOM

IFPS Models
D.3.1 IFPS Model COST
D.3.2 IFPS Model DD1
D.3.3 IFPS Model DD2
D.3.4 IFPS Model ENROLL
D.3.5 IFPS Model FEES
D.3.6 IFPS Model PEER
D.3.7 IFPS Model TUITION

IFPS Report Generators
D.4.1 IFPS Report Generator COSTINFO
D.4.2 IFPS Report Generator CREDIT
D.4.3 IFPS Report Generator CREDMPS
D.4.4 IFPS Report Generator DDFEEMPS
D.4.5 IFPS Report Generator DORMDING
D.4.6 IFPS Report Generator ENROLL
D.4.7 IFPS Report Generator FEES
D.4.8 IFPS Report Generator INDEXCHG
D.4.9 IFPS Report Generator PEERMPS
D.4.10 IFPS Report Generator PERCOST
D.4.11 IFPS Report Generator ROUND
D.4.12 IFPS Report Generator STATEREV
D.4.13 IFPS Report Generator TOTCOST1
D.4.14 IFPS Report Generator TOTCOST2
D.4.15 IFPS Report Generator TOTCOST3
D.4.16 IFPS Report Generator TOTCOST4

```
*************************************************************************
*                                                                       *
* MODIFICATION OF EXECUCOM'S IFPS EXEC                                  *
*                                                                       *
*************************************************************************
*
* THIS IS THE STANDARD IFPS EXEC MODIFIED TO DIRECTLY INTERACT WITH
*     IFPS; IT AUTOMATICALLY CALLS IFPS, DEFINES THE MODEL & REPORTS FILE
*     NAME, AND EXECUTES A COMMAND FILE.
* EIGHT PARAMETERS ARE PASSED FROM THE DSS EXECS:
*     1, 2, AND 3 ARE DUMMIES
*     4 = NAME OF THE MODEL & REPORTS FILE
*     5 = NAME OF THE COMMAND FILE
*     6 = COMMAND FILE PARAMETER 1
*     7 = COMMAND FILE PARAMETER 2
*     8 = COMMAND FILE PARAMETER 3
*
* THIS IS COMPATIBLE WITH IFPS RELEASE 9.0 V2 WITH IFPS/LINK 1.0 V2
* FOR USE WITH AN IBM 4341 GROUP 2 PROCESSOR.
*
***** STANDARD IFPS EXEC FROM HERE ..
*
*      IFPS EXEC
&CONTROL OFF NOMSG
SET LDRTBLS 10
CP TERM LINES 133
* MAKE SURE STACK IS CLEAR
&LOOP 1 &READFLAG EQ CONSOLE
&READ VARS
*&VADDR = 299
*&MODE = B
    FREEADDR
&READ VARS &VADDR
EXEC FREEFM STACK LIFO
&READ VARS &MODE
&NOCYL = 2
&LABEL = TMPDSK
* THIS TRANSFER HAS BEEN ADDED TO AVOID THE LOGICAL TEST. TO
*     HAVE THE EXEC IGNORE THE FIRST 3 PARAMETERS PASSED FROM THE DSS
*     EXECS.  ALSO LET THIS EXEC DYNAMICALLY ASSIGN THE MODE AND
*     VADDR OF THE TDISK.
&GOTO -A
&IF &INDEX GE 1 &NOCYL = &1
&IF &INDEX GE 2 &MODE = &2
&IF &INDEX GE 3 &VADDR = &3
-A
&STACK !!!
* WARNING: MAKE SURE VADDR AND MODE DO NOT CONFLICT WITH YOUR
*          CURRENT DISK ASSIGNMENTS. IF SO, AN ENTIRE DISK CAN
*          BE REFORMATTED RESULTING IN LOSS OF DATA.
*RELEASE &MODE (DET)
*CP DEFINE T3370 &VADDR &NOCYL
```

```
EXEC TDISK &NOCYL &MODE &VADDR
*&STACK YES
*&STACK &LABEL
*FORMAT &VADDR &MODE (BLKSIZE 800)
*      STACK MODE SO IFPS WILL READ IT FIRST
&STACK &MODE DISABLE
*      TERMINAL CANNOT HANDLE MORE THAN 130 CHARACTERS ON INP
FI 5 TERM (LRECL 132 BLKSIZE 132 LOWCASE)
*      IFPS DOES NOT USE UNIT 6 FOR OUTPUT BUT IT MUST BE ASS
FI 6 TERM (LRECL 80 BLKSIZE 80)
CP TERMINAL MODE VM
*    RUN IFPS
*
***** .... TO HERE
*
* CHANGED    &STACK RT   TO     SET CMSTYPE RT
* ADDED &STACK MODEL&REPORT_FILENAME
* ADDED &STACK COMMANDS COMMAND_FILENAME
SET CMSTYPE RT
&STACK &4
&STACK COMMANDS &5 &6 &7 &8
IFPS
*    FREE TEMP DISK
RELEASE &MODE (DET)
&EXIT
**********
*     IFPS IS CALLED BY:
*     IFPS NOCYL MODE VADDR
*     NOCYL IS NUMBER OF CYLINDERS (DEFAULT 2)
*     MODE IS DISK MODE (DEFAULT B)
*     VADDR IS DISK ADDRESS (DEFAULT SET ABOVE)
*     PARAMETERS MAY BE OMITTED FROM THE RIGHT OF COURSE
**********
```

D.1.2 Linkage Program QUIT

```
&TRACE
&STACK QUIT
&EXIT
```

```
10 !COMMENT CALCULATE THE STATE REVENUE REQUIREMENT GOAL TARGET VALUES
20 !COMMENT CALCULATE THE COEFFICIENTS OF THE PERCENTAGE OF COST GOALS
30 !SUPPRESS
200 COMBINE ENROLL COST
210 NONE
220 USING ENRLWORK COST SETUP
225 GENREPORT STATEREV APPEND ONLY ONTO STATEREV
230 GENREPORT PERCOST ONLY ONTO PERCOST
235 GENREPORT COSTINFO ONLY ONTO COSTINFO
240 GENREPORT INDEXCHG ONLY ONTO INDEXCHG
700 CMS EXEC QUIT
1001 !EXIT
```

```
20 !COMMENT PROCESSING FEE INFORMATION USING IFPS
25 !COMMENT &1, PASSED FROM DSS EXEC, IS THE DORM & DINING OPTION NO.
30 !SUPPRESS
200 COMBINE ENROLL FEES DD&1
210 NONE
220 USING ENRLWORK COST FEEBASE CAPIFPS DDBASE&1 SETUP
230 STORENT DDFEEWK&1
240 L1200 THRU L3060
245 NONE
250 GENREPORT FEES ONLY ONTO DDFEEOUT
260 GENREPORT DORMDING APPEND ONLY ONTO DDFEEOUT
270 GENREPORT DDFEEMPS ONLY ONTO DDFEEMPS
700 CMS EXEC QUIT
1001 !EXIT
```

```
1 !COMMENT ALL PARAMETERS ARE PASSED FROM THE EDP, DSS EXEC
5 !COMMENT &1 THE NAME OF THE "PROJECTED" ENROLLMENT DATA FILE
6 !COMMENT &2 THE NAME OF THE "ACTUAL" DATA FILE
10 !SUPPRESS
20 CMS ERASE STATEREV IFPS A
21 CMS ERASE CREDOUT IFPS A
22 CMS ERASE ENROLOUT IFPS A
99 !SUPPRESS
175 MODEL ENROLL USING &1 &2 FACTOR SETUP
300 STORENT ENRLWORK
310 ALL
400 !COMMENT STATE REV RQMT COEFFICIENTS DATAFILE:
410 GENREPORT CREDMPS ONLY ONTO STATEREV
450 !COMMENT MANAGEMENT REPORT - CREDIT HOUR BACKUP DATA:
460 GENREPORT CREDIT ONLY ONTO CREDOUT
500 !COMMENT MANAGEMENT REPORT - ENROLLMENT BACKUP DATA:
510 GENREPORT ENROLL ONLY ONTO ENROLOUT
1000 CMS EXEC QUIT
1001 !EXIT
```

```
20 !COMMENT CALCULATING PEER INSTITUTION VALUES USING IFPS
25 !COMMENT &1 = &NSCAT PASSED FROM PEERINST EXEC
30 !SUPPRESS
50 CMS ERASE PEERCALC IFPS
60 CMS ERASE PEEROUT IFPS
70 CMS ERASE PEERCHG IFPS
75 !SUPPRESS
100 !COMMENT OPTION H2
120 !SETPARAM 2 0
150 !PARAMCOMPUTE 2 &2 + 1
200 MODEL PEER USING PEER&2 COST SETUP
210 STORENT PEERCALC
220 L3300
225 NONE
230 L3300
240 NONE
510 !IF &2 .LT. &1 THEN !GOTO 150
600 MERGE PEERCALC, SETUP, NONE
610 PEEROUT
620 DATAFILE PEEROUT
630 GENREPORT PEERMPS ONLY ONTO PEERCHG
700 CMS EXEC QUIT
1001 !EXIT
```

```
10 !COMMENT ALL PARAMETERS ARE PASED FROM THE CCP REPORTS EXEC
20 !COMMENT PROCESSING TUITION INFORMATION AFTER RUNNING GP MODEL
21 !COMMENT &1 = NAME OF THE "PROJECTED" ENROLLMENT DATA FILE
22 !COMMENT &2 = NAME OF THE "ACTUAL" ENROLLMENT DATA FILE
23 !COMMENT &3 = DORM DINING CALCULATION OPTION NUMBER
30 !SUPPRESS
35 CMS ERASE TOTCOST IFPS A
200 COMBINE ENROLL FEES DD&3 TUITION COST
210 NONE
220 USING COST FEEBASE CAPIFPS DDBASE&3 TUITION SETUP FACTOR &1 &2
310 GENREPORT TOTCOST1 APPEND ONLY ONTO TOTCOST
320 GENREPORT TOTCOST2 APPEND ONLY ONTO TOTCOST
330 GENREPORT TOTCOST3 APPEND ONLY ONTO TOTCOST
340 GENREPORT TOTCOST4 APPEND ONLY ONTO TOTCOST
350 GENREPORT ROUND APPEND ONLY ONTO TOTCOST
700 CMS EXEC QUIT
1001 !EXIT
```

```
FILE     MODEL    COST     12/03/8330:54:04
10 COLUMNS &MODEL COLUMNS&
5100     *
5100.1 * CALCULATE TUITION AND FEE RECOVERY RATE
5100.2 *
5101 CONSUMER PRICE INDEX = DATA
5102 CPI CHANGE = 0, L5101/(PREVIOUS L5101)
5103 PERCENT CHANGE IN CPI = 0, 100*(L5102 - 1)
5105 TUITION AND FEE RECOVERY RATE = DATA
5106 MAXIMUM RECOVERY RATE INCREASE = DATA
5107 MAX RATE = L5106/100
5108 MAXIMUM RECOVERY RATE DECREASE = DATA
5109 MIN RATE = -1*(L5108)/100
5110 UNDERGRADUATE IEP COST PER STUDENT = DATA
5120 GRADUATE IEP COST PER STUDENT = DATA
5150 * POLICY RATES
5160 RESIDENT     UNDERGRAD POLICY RATE = DATA
5170 NONRESIDENT UNDERGRAD POLICY RATE = DATA
5180 RESIDENT     GRADUATE  POLICY RATE = DATA
5190 NONRESIDENT GRADUATE  POLICY RATE = DATA
5200     * COST RATIOS (ASSUME LATEST IEP DATA IS USED UNLESS OTHERWISE
5200.1 *    INDICATED IN FILE COST IFPS
5210 AVG IEP COST PER STUDENT = (L5110*L871 + L5120*L872)/L870
5220 UNDERGRAD IEP COST RATIO = L5110/L5210
5230 GRADUATE  IEP COST RATIO = L5120/L5210
5300 * ENROLLMENT MIX
5310     * CALCULATE WEIGHTED AVERAGE OF GRADUATE STUDENT EMPLOYMENT RATES
5310.1 * ASSUME THERE IS A 2-YEAR LAG BETWEEN THE LATEST EMPLOYMENT DATA
5310.2 *   AND THE YEARS THAT THE TUITION CHARGES APPLY.
5310.3 * USE AN EQUALLY WEIGHTED 3-YEAR MOVING AVERAGE
5320 WTG AVG EMPLOY RATE = ((PREVIOUS 3 L410) + (PREVIOUS 4 L410) + '
5321    (PREVIOUS 5 L410))/3
5330 EMPLOYMENT RATE = IF L5320 .LT. L420 THEN L5320 ELSE L420
5350 * PROPORTION OF STUDENTS IN EACH CATEGORY, ADJUSTED FOR NON-RESIDENT
5351 *   GRADUATE EMPLOYMENT
5360 RES   UG ENROLL PROP = L810/L870
5370 NRES UG ENROLL PROP = L820/L870
5380 RES   GR ENROLL PROP = (L873 + L5330*L874)/L870
5390 NRES GR ENROLL PROP = (1 - L5330)*L874/L870
5400 * RECOVERY RATE
5410 R   UG PARTIAL RECOVERY RATE = L5160*L5220*L5360/100
5420 NR UG PARTIAL RECOVERY RATE = L5170*L5220*L5370/100
5430 R   GR PARTIAL RECOVERY RATE = L5180*L5230*L5380/100
5440 NR GR PARTIAL RECOVERY RATE = L5190*L5230*L5390/100
5445 SUM RATE = SUM(L5410,L5440)
5450 RATE = IF L5105 .EQ. 0 THEN L5445 ELSE L5105
5460 RATE CHANGE = L5450 - PREVIOUS L5450
5470 RECOVERY RATE = IF L5460 .LT. L5109 THEN '
5471    PREVIOUS L5450 + L5109 ELSE IF L5460 .GT. L5107 THEN '
5472    PREVIOUS L5450 + L5107  ELSE L5450
5600     *
5600.1 * CALCULATE THE STATE REVENUE REQUIREMENT GOAL TARGETS
```

```
5600.2 *
5610 ADJ INSTRUCT E!&G = DATA
5612 UNADJUSTED STATE REVENUE RQMT = L5610*L5470
5615 MISC FEES = DATA
5616 MISCELLANEOUS FEES = -L5615
5620 STATE REVENUE REQUIREMENT = L5612 - L5615
5630 ST REV RQMT = L5620
5650 ST REV RQMT !% CHANGE OVER PRIOR YEAR = 0,'
5655 IF PREVIOUS L5630 .EQ. 0 THEN 0 ELSE ((L5620/PREVIOUS L5620)-1)*100
5660 CHANGE IN OPERATING EXPENSES OVER PRIOR YEAR = 0,'
5665 IF PREVIOUS L5610 .EQ. 0 THEN 0 ELSE ((L5610/PREVIOUS L5610)-1)*100
5700   *
5700.1 * CALCULATE COEFFICIENTS FOR PERCENTAGE OF TOTAL COST GOALS
5700.2 *
5710 AVERAGE COST = 3*L5610/L881
5810 &NAME1& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L311/L5710
5820 &NAME2& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L311/L5710
5830 &NAME3& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L312/L5710
5840 &NAME4& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L312/L5710
5850 &NAME5& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L312/L5710
5860 &NAME6& PERCOST COEF = IF L5710 .EQ. 0 THEN 0 ELSE 300*L312/L5710
```

```
FILE    MODEL    DD1      10/31/8320:12:07
10 COLUMNS &MODEL COLUMNS&
2500    *
2500.1 * CALCULATE AVERAGE DORM AND DINING CHARGES FOR
2500.2 *    UNDERGRADUATES AND GRADUATES LIVING ON-CAMPUS
2500.7 *
2500.9 * BASED ON CHARGES INPUT BY DORMITORY
2501    *
2501.1 * DORMITORY CHARGES
2501.9 *
2505 *NOTE: ALL 1983-84 DORM RATES HAVE BEEN REDUCED BY $37/STUDENT/QRTR
2505.1 *TO ACCOUNT FOR THE DEBT SERVICE ON THE NEW DORM FACILITIES
2505.2 *(ASSUMING 9,000,000 FOR 20 YR. AT 9%)
2510 UPPER QUAD QRTLY CHARGE = DATA
2515 UQ QRTLY CHARGE = IF L2510 .NE. 0 THEN L2510 ELSE '
2516    (PREVIOUS L2515)*L2590
2520 LOWER QUAD QRTLY CHARGE = DATA
2525 LQ QRTLY CHARGE = IF L2520 .NE. 0 THEN L2520 ELSE '
2526    (PREVIOUS L2525)*L2590
2530 PRAIRIE QUAD QRTLY CHARGE = DATA
2535 PQ QRTLY CHARGE = IF L2530 .NE. 0 THEN L2530 ELSE '
2536    (PREVIOUS L2535)*L2590
2540 NEW UNDERGRAD QRTLY CHARGE = DATA
2545 NU QRTLY CHARGE = IF L2540 .NE. 0 THEN L2540 ELSE '
2546    (PREVIOUS L2545)*L2590
2550 SPEC PURPOSE QRTLY CHARGE = DATA
2555 SP QRTLY CHARGE = IF L2550 .NE. 0 THEN L2550 ELSE '
2556    (PREVIOUS L2555)*L2590
2560 GRAD SINGLE OCC QRTLY CHARGE = DATA
2565 GS QRTLY CHARGE = IF L2560 .NE. 0 THEN L2560 ELSE '
2566    (PREVIOUS L2565)*L2590
2570 GRAD DOUBLE OCC QRTLY CHARGE = DATA
2575 GD QRTLY CHARGE = IF L2570 .NE. 0 THEN L2570 ELSE '
2576    (PREVIOUS L2575)*L2590
2585 DORM CHARGES ESCALATOR = DATA
2590 DORM INDEX = IF L2585 .LT. 0 THEN L1202 ELSE L2585
2610 UPPER QUAD STUDENTS  = DATA
2620 LOWER QUAD STUDENTS  = DATA
2630 PRAIRIE QUAD STUDENTS  = DATA
2640 NEW UNDERGRAD STUDENTS = DATA
2650 SPEC PURPOSE STUDENTS  = DATA
2660 GRAD SINGLE OCC STUDENTS  = DATA
2670 GRAD DOUBLE OCC STUDENTS  = DATA
2700 QRTLY UG DORM REVENUE = L2515*L2610 + L2525*L2620 + L2535*L2630'
2710       +L2545*L2640 + L2555*L2650
2720 QRTLY GRAD DORM REVENUE = L2565*L2660 + L2575*L2670
2730 UG DORM STUDENTS = L2610 + L2620 + L2630 + L2640 + L2650
2740 GRAD DORM STUDENTS = L2660 + L2670
2750 TOTAL DORM STUDENTS = L2730 + L2740
2760 ANNUAL EQUIVALENT DORM STUDENTS = 3*L2750
2770 UG AVG BASE DORM FEE = ROUND(L2700/L2730)
2775 UG AVG ANNUAL DORM = 3*L2770
```

```
2780 GR AVG BASE DORM FEE = ROUND(L2720/L2740)
2785 GR AVG ANNUAL DORM = 3*L2780
2800   *
2800.1 * DINING CHARGES
2800.9 *
2810 UG 19 MEAL PLAN QRTLY CHARGE  = DATA
2815 UG QRTLY MEAL CHARGE = IF L2810 .NE. 0 THEN L2810 ELSE '
2816   (PREVIOUS L2815)*L2890
2820 ANNUAL UG MEAL = 3*ROUND(L2815)
2830 UG 19 MEAL PLAN STUDENTS  = L2730
2840 GR 15 MEAL PLAN QRTLY CHARGE = DATA
2845 GR QRTLY MEAL CHARGE = IF L2840 .NE. 0 THEN L2840 ELSE '
2846   (PREVIOUS L2845)*L2890
2850 ANNUAL GR MEAL = 3*ROUND(L2845)
2860 GRADUATE 15 MEAL PLAN STUDENTS  = L2740
2885 DINING CHARGES ESCALATOR = DATA
2890 DINING INDEX = IF L2885 .LT. 0 THEN L1202 ELSE L2885 .
2900   *
2900.1 * NEW DORM AND DINING CAPITAL PROJECTS
2900.9 *
2910 * DORM AND DINING PROJECT 1
2911 DD1 AMORT     = DATA
2912 DD1 UNAMORT   = DATA
2913 DD1 INT RATE  = DATA
2914 DD1 PR LIFE   = DATA
2915 DD1 START     = DATA
2916 DD1 PPY       = DATA
2917 AMORT(DD1 AMORT, DD1 UNAMORT, DD1 INT RATE, DD1 PR LIFE, DD1 START,'
2918     DD1 PPY, DD1 PAY,INT,PRIN,BAL)
2919 DD1 DEBT = 1000*(DD1 PAY)/L2760
2920 * DORM AND DINING PROJECT 2
2921 DD2 AMORT     = DATA
2922 DD2 UNAMORT   = DATA
2923 DD2 INT RATE  = DATA
2924 DD2 PR LIFE   = DATA
2925 DD2 START     = DATA
2926 DD2 PPY       = DATA
2927 AMORT(DD2 AMORT, DD2 UNAMORT, DD2 INT RATE, DD2 PR LIFE, DD2 START,'
2928     DD2 PPY, DD2 PAY,INT,PRIN,BAL)
2929 DD2 DEBT = 1000*(DD2 PAY)/L2760
2930 * DORM AND DINING PROJECT 3
2931 DD3 AMORT     = DATA
2932 DD3 UNAMORT   = DATA
2933 DD3 INT RATE  = DATA
2934 DD3 PR LIFE   = DATA
2935 DD3 START     = DATA
2936 DD3 PPY       = DATA
2937 AMORT(DD3 AMORT, DD3 UNAMORT, DD3 INT RATE, DD3 PR LIFE, DD3 START,'
2938     DD3 PPY, DD3 PAY,INT,PRIN,BAL)
2939 DD3 DEBT = 1000*(DD3 PAY)/L2760
2940 QRTLY DEBT SERVICE = L2919 + L2929 + L2939
2950 DD DEBT SERVICE = 3*L2940
3010 BASE UG ROOM AND BOARD = L2775 + L2820
3020 BASE GR ROOM AND BOARD = L2785 + L2850
3030 TOTAL UG ROOM AND BOARD = L3010 + L2950
3040 TOTAL GR ROOM AND BOARD = L3020 + L2950
3050 TOTAL UG FEES ROOM AND BOARD = L3030 + L2100
3060 TOTAL GR FEES ROOM AND BOARD = L3040 + L2100
```

```
FILE     MODEL    DD2      10/31/8320:12:07
10 COLUMNS &MODEL COLUMNS&
2400     *
2400.1 * CALCULATE AVERAGE DORM AND DINING CHARGES FOR UNDERGRADUATES
2400.2 *    AND GRADUATES LIVING ON-CAMPUS, BASED ON OVERALL OPERATING
2400.4 *    OPERATING EXPENSES.
2400.5 *
2400.6 * OVERALL BUDGET IS ALLOCATED TO THE INDIVIDUAL DORMITORY TYPES
2400.7 *
2405 DORM OPERATING EXPENSES 000 = DATA
2410 DORM BUDGET = IF L2405 .EQ. 0 THEN (PREVIOUS L2410)*L2590 '
2411      ELSE 1000*L2405
2420 EQUIV QRTLY STUDENTS = 3*( L2510*L2610 + L2520*L2620 + L2530*L2630 '
2425    + L2540*L2640 + L2550*L2650 + L2560*L2660 + L2570*L2670 )
2450 AVG QRTLY CHG PER STUDENT = L2410/L2420
2460 DINING OPERATING EXPENSES 000 = DATA
2470 DINING BUDGET = IF L2460 .EQ. 0 THEN (PREVIOUS L2470)*L2890 '
2471      ELSE 1000*L2460
2480 EQUIV QRTLY STUDENTS = 3*( L2810*L2830 + L2840*L2860 )
2490 AVG QRTLY CHG PER STUDENT = L2470/L2480
2501     *
2501.1 * DORMITORY CHARGES
2501.9 *
2502.2 *NOTE: ALL 1983-84 DORM RATES HAVE BEEN REDUCED BY $37/STUDENT/QTR
2502.3 *TO ACCOUNT FOR THE DEBT SERVICE ON THE NEW DORM FACILITIES
2502.4 *(ASSUMING 9,000,000 FOR 20 YR. AT 9%)
2510 UPPER QUAD RATIO = DATA
2515 UQ QRTLY CHARGE = L2510*L2450
2520 LOWER QUAD RATIO = DATA
2525 LQ QRTLY CHARGE = L2520*L2450
2530 PRAIRIE QUAD RATIO = DATA
2535 PQ QRTLY CHARGE = L2530*L2450
2540 NEW UNDERGRAD RATIO = DATA
2545 NU QRTLY CHARGE = L2540*L2450
2550 SPEC PURPOSE RATIO = DATA
2555 SP QRTLY CHARGE = L2550*L2450
2560 GRAD SINGLE OCC RATIO = DATA
2565 GS QRTLY CHARGE = L2560*L2450
2570 GRAD DOUBLE OCC RATIO = DATA
2575 GD QRTLY CHARGE = L2570*L2450
2585 DORM CHARGES ESCALATOR = DATA
2590 DORM INDEX = IF L2585 .LT. 0 THEN L1202 ELSE L2585
2610 UPPER QUAD STUDENTS  = DATA
2620 LOWER QUAD STUDENTS  = DATA
2630 PRAIRIE QUAD STUDENTS  = DATA
2640 NEW UNDERGRAD STUDENTS = DATA
2650 SPEC PURPOSE STUDENTS  = DATA
2660 GRAD SINGLE OCC STUDENTS  = DATA
2670 GRAD DOUBLE OCC STUDENTS  = DATA
2700 QRTLY UG DORM REVENUE = L2515*L2610 + L2525*L2620 + L2535*L2630'
2710      +L2545*L2640 + L2555*L2650
2720 QRTLY GRAD DORM REVENUE = L2565*L2660 + L2575*L2670
```

```
2730 UG DORM STUDENTS = L2610 + L2620 + L2630 + L2640 + L2650
2740 GRAD DORM STUDENTS = L2660 + L2670
2750 TOTAL DORM STUDENTS = L2730 + L2740
2760 ANNUAL EQUIVALENT DORM STUDENTS = 3*L2750
2770 UG AVG BASE DORM FEE = ROUND(L2700/L2730)
2775 UG AVG ANNUAL DORM = 3*L2770
2780 GR AVG BASE DORM FEE = ROUND(L2720/L2740)
2785 GR AVG ANNUAL DORM = 3*L2780
2800   *
2800.1 * DINING CHARGES
2800.9 *
2810 UG 19 MEAL PLAN RATIO  = DATA
2815 UG QRTLY MEAL CHARGE = L2810*L2490
2820 ANNUAL UG MEAL = 3*ROUND(L2815)
2830 UG 19 MEAL PLAN STUDENTS  = L2730
2840 GR 15 MEAL PLAN RATIO = DATA
2845 GR QRTLY MEAL CHARGE = L2840*L2490
2850 ANNUAL GR MEAL = 3*ROUND(L2845)
2860 GRADUATE 15 MEAL PLAN STUDENTS  = L2740
2885 DINING CHARGES ESCALATOR = DATA
2890 DINING INDEX = IF L2885 .LT. 0 THEN L1202 ELSE L2885
2900   *
2900.1 * NEW DORM AND DINING CAPITAL PROJECTS
2900.9 *
2910 * DORM AND DINING PROJECT 1
2911 DD1 AMORT     = DATA
2912 DD1 UNAMORT   = DATA
2913 DD1 INT RATE = DATA
2914 DD1 PR LIFE   = DATA
2915 DD1 START     = DATA
2916 DD1 PPY       = DATA
2917 AMORT(DD1 AMORT, DD1 UNAMORT, DD1 INT RATE, DD1 PR LIFE, DD1 START,'
2918    DD1 PPY, DD1 PAY,INT,PRIN,BAL)
2919 DD1 DEBT = 1000*(DD1 PAY)/L2760
2920 * DORM AND DINING PROJECT 2
2921 DD2 AMORT     = DATA
2922 DD2 UNAMORT   = DATA
2923 DD2 INT RATE = DATA
2924 DD2 PR LIFE   = DATA
2925 DD2 START     = DATA
2926 DD2 PPY       = DATA
2927 AMORT(DD2 AMORT, DD2 UNAMORT, DD2 INT RATE, DD2 PR LIFE, DD2 START,'
2928    DD2 PPY, DD2 PAY,INT,PRIN,BAL)
2929 DD2 DEBT = 1000*(DD2 PAY)/L2760
2930 * DORM AND DINING PROJECT 3
2931 DD3 AMORT     = DATA
2932 DD3 UNAMORT   = DATA
2933 DD3 INT RATE = DATA
2934 DD3 PR LIFE   = DATA
2935 DD3 START     = DATA
2936 DD3 PPY       = DATA
2937 AMORT(DD3 AMORT, DD3 UNAMORT, DD3 INT RATE, DD3 PR LIFE, DD3 START,'
2938    DD3 PPY, DD3 PAY,INT,PRIN,BAL)
2939 DD3 DEBT = 1000*(DD3 PAY)/L2760
2940 QRTLY DEBT SERVICE = L2919 + L2929 + L2939
```

```
2950 DD DEBT SERVICE = 3*L2940
3010 BASE UG ROOM AND BOARD = L2775 + L2820
3020 BASE GR ROOM AND BOARD = L2785 + L2850
3030 TOTAL UG ROOM AND BOARD = L3010 + L2950
3040 TOTAL GR ROOM AND BOARD = L3020 + L2950
3050 TOTAL UG FEES ROOM AND BOARD = L3030 + L2100
3060 TOTAL GR FEES ROOM AND BOARD = L3040 + L2100
```

```
FILE    MODEL    ENROLL  10/21/8320:15:34
10 COLUMNS &MODEL COLUMNS&
12 *  &DATE&
100    *
100.1 *  "ACTUAL" ENROLLMENT - USED TO CALCULATE EXPECTED CREDIT HOURS
100.2 *      AND  STUDENT FEES
100.9 *
110 **      REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
110.5 **           FULL-TIME (FT)
111 ACTUAL REG FT UG RES  = DATA
112 ACTUAL REG FT UG NRES = DATA
113 ACTUAL REG FT GR RES  = DATA
114 ACTUAL REG FT GR NRES = DATA
120 ***            PART-TIME (PT)
121 ACTUAL REG PT UG RES  = DATA
122 ACTUAL REG PT UG NRES = DATA
123 ACTUAL REG PT GR RES  = DATA
124 ACTUAL REG PT GR NRES = DATA
130 **      SUMMER SESSION (SUM) - HEADCOUNT
130.5 ***          FULL-TIME (FT)
131 ACTUAL SUM FT UG RES  = DATA
132 ACTUAL SUM FT UG NRES = DATA
133 ACTUAL SUM FT GR RES  = DATA
134 ACTUAL SUM FT GR NRES = DATA
140 ***            PART-TIME (PT)
141 ACTUAL SUM PT UG RES  = DATA
142 ACTUAL SUM PT UG NRES = DATA
143 ACTUAL SUM PT GR RES  = DATA
144 ACTUAL SUM PT GR NRES = DATA
150 **      OFF-CAMPUS (OFC) - FALL QUARTER HEADCOUNT
150.5 ***          FULL-TIME (FT)
151 ACTUAL OFC FT UG RES  = DATA
152 ACTUAL OFC FT UG NRES = DATA
153 ACTUAL OFC FT GR RES  = DATA
154 ACTUAL OFC FT GR NRES = DATA
160 ***            PART-TIME (PT)
161 ACTUAL OFC PT UG RES  = DATA
162 ACTUAL OFC PT UG NRES = DATA
163 ACTUAL OFC PT GR RES  = DATA
164 ACTUAL OFC PT GR NRES = DATA
170 * HEADCOUNT SUBTOTALS
171 ACTUAL REG UG = L111 + L112 + L121 + L122
172 ACTUAL REG GR = L113 + L114 + L123 + L124
173 ACTUAL SUM UG = L131 + L132 + L141 + L142
174 ACTUAL SUM GR = L133 + L134 + L143 + L144
176 ACTUAL OFC GR = L153 + L154 + L163 + L164
177 ACTUAL REG = L171 + L172
178 ACTUAL SUM = L173 + L174
200    *
200.1 * "PROJECTED" ENROLLMENT - USED TO CALCULATE STATE REVENUE REQMT
200.2 *
210 **  REGULAR SESSION
```

```
210.5 *** FULL-TIME (FT)
211 PROJ REG FT UG RES  = DATA
212 PROJ REG FT UG NRES = DATA
213 PROJ REG FT GR RES  = DATA
214 PROJ REG FT GR NRES = DATA
220 ***         PART-TIME (PT)
221 PROJ REG PT UG RES  = DATA
222 PROJ REG PT UG NRES = DATA
223 PROJ REG PT GR RES  = DATA
224 PROJ REG PT GR NRES = DATA
230 ***   SUMMER SESSION
230.5 ***         FULL-TIME (FT)
231 PROJ SUM FT UG RES  = DATA
232 PROJ SUM FT UG NRES = DATA
233 PROJ SUM FT GR RES  = DATA
234 PROJ SUM FT GR NRES = DATA
240 ***         PART-TIME (PT)
241 PROJ SUM PT UG RES  = DATA
242 PROJ SUM PT UG NRES = DATA
243 PROJ SUM PT GR RES  = DATA
244 PROJ SUM PT GR NRES = DATA
250 **     OFF-CAMPUS (OFC)
250.5 ***         FULL-TIME (FT)
251 PROJ OFC FT UG RES  = DATA
252 PROJ OFC FT UG NRES = DATA
253 PROJ OFC FT GR RES  = DATA
254 PROJ OFC FT GR NRES = DATA
260 ***         PART-TIME (PT)
261 PROJ OFC PT UG RES  = DATA
262 PROJ OFC PT UG NRES = DATA
263 PROJ OFC PT GR RES  = DATA
264 PROJ OFC PT GR NRES = DATA
270 * HEADCOUNT SUBTOTALS
271 PROJ REG UG = L211 + L212 + L221 + L222
272 PROJ REG GR = L213 + L214 + L223 + L224
273 PROJ SUM UG = L231 + L232 + L241 + L242
274 PROJ SUM GR = L233 + L234 + L243 + L244
276 PROJ OFC GR = L253 + L254 + L263 + L264
277 PROJ REG = L271 + L272
278 PROJ SUM = L273 + L274
300   *
300.1 * CREDIT HOUR DATA
300.2 *    1) NUMBER OF CREDIT HOURS TO BE CONSIDERED FULL-TIME FOR FEE
300.3 *       PURPOSES
300.4 *    2) AVERAGE NUMBER OF CREDIT HOURS FOR PART-TIME STUDENTS,
300.5 *          PER QUARTER, BY CATEGORY
300.6 *
310 ***         FULL-TIME (FT)
311 CREDIT HOUR FT UNDERGRAD = DATA
312 CREDIT HOUR FT GRADUATE  = DATA
315 *SUBROUTINE CREDHR, CRED
316 *CREDHR(CREDIT HOUR FT UNDERGRAD, CREDIT HOUR FT GRADUATE,'
318 *      CH REG PT UG, CH REG PT GR, AVG CH REG UG, AVG CH REG GR)
320 ***         PART-TIME (PT)
320.5 **  REGULAR SESSION
```

```
321 CREDIT HOUR REG PT UG RES  = DATA
322 CREDIT HOUR REG PT UG NRES = DATA
323 CREDIT HOUR REG PT GR RES  = DATA
324 CREDIT HOUR REG PT GR NRES = DATA
330 ***    SUMMER SESSION
341 CREDIT HOUR SUM PT UG RES  = DATA
342 CREDIT HOUR SUM PT UG NRES = DATA
343 CREDIT HOUR SUM PT GR RES  = DATA
344 CREDIT HOUR SUM PT GR NRES = DATA
350 **      OFF-CAMPUS (OFC)
361 CREDIT HOUR OFC PT UG RES  = DATA
362 CREDIT HOUR OFC PT UG NRES = DATA
363 CREDIT HOUR OFC PT GR RES  = DATA
364 CREDIT HOUR OFC PT GR NRES = DATA
380   *
380.1 * RETENTION FACTORS - CHANGE IN ENROLLMENT BETWEEN QUARTERS
380.2 *
390 RETENTION FACTOR!-REG SESSION = DATA
395 RETENTION FACTOR!-OFC = DATA
400   *
400.1 * GRADUATE STUDENT EMPLOYMENT RATES
400.2 *
410 NR GR EMPLOYMENT RATE = DATA
420 MAX NR GR EMPLOYMENT RATE = DATA
500   *
500.1 * AVERAGE CREDIT HOURS PER STUDENT PER QUARTER (FULL-TIME AND
500.2 *     PART-TIME COMBINED, BASED ON FALL-QUARTER ENROLLMENTS)
500.3 *
510 **  REGULAR SESSION
521 AVERAGE CREDIT HOUR REG UG = DATA
522 AVERAGE CREDIT HOUR REG GR = DATA
530 ***    SUMMER SESSION
541 AVERAGE CREDIT HOUR SUM UG = DATA
542 AVERAGE CREDIT HOUR SUM GR = DATA
550 **      OFF-CAMPUS (OFC)
561 AVERAGE CREDIT HOUR OFC UG = DATA
562 AVERAGE CREDIT HOUR OFC GR = DATA
600   *
600.1 * FEE WAIVER DATA
600.2 *
614 FEE WAIVER REG FT GR NRES = DATA
624 FEE WAIVER REG PT GR NRES = DATA
634 FEE WAIVER SUM FT GR NRES = DATA
644 FEE WAIVER SUM PT GR NRES = DATA
654 FEE WAIVER OFC FT GR NRES = DATA
664 FEE WAIVER OFC PT GR NRES = DATA
700   *
700.1 * EXPECTED ANNUAL CREDIT HOURS GENERATED - FOR TUITION
700.2 *     REVENUE PURPOSES (BASED ON "ACTUAL" ENROLLMENT)
700.3 *
710 CR HRS REG SUM UG  RES = (L390*(L111*L311 + L121*L321) + '
711         (L131*L311 + L141*L341))
720 CR HRS REG SUM UG NRES = (L390*(L112*L311 + L122*L322) + '
721         (L132*L311 + L142*L342))
730 CR HRS REG SUM GR  RES = (L390*(L113*L312 + L123*L323 '
```

```
731     + L114*L312*L614 + L124*L324*L624)  + (L133*L312 + L143*L343'
732         + L134*L312*L634 + L144*L344*L644))
740 CR HRS REG SUM GR NRES = (L390*(L114*L312*(1-L614) + L124*L324*(1-L624))'
741    + (L134*L312*(1-L634) + L144*L344*(1-L644)))
750 CR HRS OFC GR  RES = L395*(L153*L312 + L163*L363 '
751    + L154*L312*L654 + L164*L364*L664)
760 CR HRS OFC GR NRES = L395*(L154*L312*(1-L654) + L164*L364*(1-L664))
770 CR HRS TOTAL = L710 + L720 + L730 + L740 + L750 + L760
800    *
800.1 * FALL QUARTER FULL-TIME EQUIVALENT STUDENTS (FTE)
800.2 *    BASED ON "PROJECTED" ENROLLMENT
800.3 *
810 FALL QTR FTE  UG  RES = L521*(L211 + L221)/15
820 FALL QTR FTE  UG NRES = L521*(L212 + L222)/15
830 FALL QTR FTE  GR  RES = L522*(L213 + L223)/12
840 FALL QTR FTE  GR NRES = L522*(L214 + L224)/12
850 FALL QTR FTE  GR  RES OFC = L562*(L253 + L263)/12
860 FALL QTR FTE  GR NRES OFC = L562*(L254 + L264)/12
870 FALL QTR FTE  = SUM(L810,L860)
871 FALL QTR FTE  UG = L810 + L820
872 FALL QTR FTE  GR = SUM(L830,L860)
873 FALL QTR FTE  RES GRADUATES  = L830 + L850
874 FALL QTR FTE  NRES GRADUATES = L840 + L860
875 FALL QTR FTE  ON  CAMPUS = SUM(L810,L840)
876 FALL QTR FTE  OFF CAMPUS = L850 + L860
880    *
880.1 * ANNUAL FULL-TIME EQUIVALENT STUDENTS BASED ON "PROJECTED"
880.2 *   ENROLLMENT
880.3 *
881 ANNUAL TOTAL FTE = L390*L875 + L395*L876 + L885 + L886
885 UG SUM FTE =  L541*(L231 + L232 + L241 + L242)/15
886 GR SUM FTE =  L542*(L233 + L234 + L243 + L244)/12
890 FTE CH = 15*(L390*L871 + L885) + '
891      12*(L390*(L830 + L840) + L395*L876 + L886)
910 &NAME1& = L710
920 &NAME2& = L720
930 &NAME3& = L730
940 &NAME4& = L740
950 &NAME5& = L750
960 &NAME6& = L760
```

```
FILE    MODEL    FEES     10/31/8320:12:07
10 COLUMNS &MODEL COLUMNS&
1200    *
1200.1 * EQUIVALENT FULL-TIME STUDENTS FOR FEE PURPOSES :
1200.2 *     ATHLETIC, BUS, STUDENT CENTER, AND HEALTH SERVICE FEES
1200.9 *
1201 CONSUMER PRICE INDEX = DATA
1202 CPI CHANGE = IF (PREVIOUS L1201) .EQ. 0 THEN 1. ELSE '
1203         L1201/(PREVIOUS L1201)
1205 GENERATE L1210 THRU L1220 AT L1225 BY 5 FOR ATH, BUS, STC, HLT
1210 # EQUIV FT STUDENTS = L390*(SUM(L111,L114) + '
1215     ( # PROP PART TIME STUDENTS)*SUM(L121,L124)) + SUM(L131,L134) + '
1220         ( # PROP PART TIME STUDENTS)*SUM(L141,L144)
1300    *
1300.1 * ATHLETIC ASSOCIATION FEE CALCULATIONS
1300.9 *
1310 ATH PROP PART TIME STUDENTS = DATA
1320 QUARTERLY ATHLETIC FEE = DATA
1330 ATH OPERATING EXPENSES 000 = DATA
1335 ATH ESCALATOR = DATA
1340 ATH INDEX = IF L1335 .LT. 0 THEN L1202 ELSE L1335
1350 CALCULATED ATH FEE = 1000*L1330/L1225
1355 PROJECTED ATH FEE = L1340*(PREVIOUS L1360)
1360 ATH FEE = IF L1320 .NE. 0  THEN L1320'
1365   ELSE IF L1330 .NE. 0 THEN L1350'
1370       ELSE L1355
1380 BASE ATHLETIC ASSOCIATION FEE = 3*ROUND(L1360)
1410 * ATHLETIC ASSOCIATION PROJECT 1
1411 AA1 AMORT     = DATA
1412 AA1 UNAMORT   = DATA
1413 AA1 INT RATE  = DATA
1414 AA1 PR LIFE   = DATA
1415 AA1 START     = DATA
1416 AA1 PPY       = DATA
1417 AMORT(AA1 AMORT, AA1 UNAMORT, AA1 INT RATE, AA1 PR LIFE, AA1 START,'
1418     AA1 PPY, ATH PAY,INT,PRIN,BAL)
1420 * ATHLETIC ASSOCIATION PROJECT 2
1421 AA2 AMORT     = DATA
1422 AA2 UNAMORT   = DATA
1423 AA2 INT RATE  = DATA
1424 AA2 PR LIFE   = DATA
1425 AA2 START     = DATA
1426 AA2 PPY       = DATA
1427 AMORT(AA2 AMORT, AA2 UNAMORT, AA2 INT RATE, AA2 PR LIFE, AA2 START,'
1428     AA2 PPY, ATH PAY,INT,PRIN,BAL)
1430 * ATHLETIC ASSOCIATION PROJECT 3
1431 AA3 AMORT     = DATA
1432 AA3 UNAMORT   = DATA
1433 AA3 INT RATE  = DATA
1434 AA3 PR LIFE   = DATA
1435 AA3 START     = DATA
1436 AA3 PPY       = DATA
```

```
1437 AMORT(AA3 AMORT, AA3 UNAMORT, AA3 INT RATE, AA3 PR LIFE, AA3 START,'
1438     AA3 PPY, ATH PAY,INT,PRIN,BAL)
1445 QRTLY ATH DEBT SERVICE = 1000*(ATH PAY)/L1225
1450 DEBT SERVICE = 3*ROUND(L1445)
1480 ATHLETIC ASSOCIATION FEE = 3*ROUND(L1360 + L1445)
1500    *
1500.1 * BUS SYSTEM FEE CALCULATIONS
1500.9 *
1510 BUS PROP PART TIME STUDENTS = DATA
1520 QUARTERLY BUS SYSTEM FEE = DATA
1530 BUS OPERATING EXPENSES 000 = DATA
1535 BUS ESCALATOR = DATA
1540 BUS INDEX = IF L1535 .LT. 0 THEN L1202 ELSE L1535
1550 CALCULATED BUS FEE = 1000*L1530/L1240
1555 PROJECTED BUS FEE = L1540*(PREVIOUS L1560)
1560 BUS FEE = IF L1520 .NE. 0  THEN L1520'
1565    ELSE IF L1530 .NE. 0 THEN L1550'
1570       ELSE L1555
1580 BASE BUS SYSTEM FEE = 3*ROUND(L1560)
1610 * BUS SYSTEM PROJECT 1
1611 BU1 AMORT    = 0
1612 BU1 UNAMORT  = 0
1613 BU1 INT RATE = 0
1614 BU1 PR LIFE  = 0
1615 BU1 START    = 0
1616 BU1 PPY      = 0
1617 AMORT(BU1 AMORT, BU1 UNAMORT, BU1 INT RATE, BU1 PR LIFE, BU1 START,'
1618     BU1 PPY, BUS PAY,INT,PRIN,BAL)
1620 * BUS SYSTEM PROJECT 2
1621 BU2 AMORT    = 0
1622 BU2 UNAMORT  = 0
1623 BU2 INT RATE = 0
1624 BU2 PR LIFE  = 0
1625 BU2 START    = 0
1626 BU2 PPY      = 0
1627 AMORT(BU2 AMORT, BU2 UNAMORT, BU2 INT RATE, BU2 PR LIFE, BU2 START,'
1628     BU2 PPY, BUS PAY,INT,PRIN,BAL)
1630 * BUS SYSTEM PROJECT 3
1631 BU3 AMORT    = 0
1632 BU3 UNAMORT  = 0
1633 BU3 INT RATE = 0
1634 BU3 PR LIFE  = 0
1635 BU3 START    = 0
1636 BU3 PPY      = 0
1637 AMORT(BU3 AMORT, BU3 UNAMORT, BU3 INT RATE, BU3 PR LIFE, BU3 START,'
1638     BU3 PPY, BUS PAY,INT,PRIN,BAL)
1645 QRTLY BUS DEBT SERVICE = 1000*(BUS PAY)/L1240
1650 DEBT SERVICE = 3*ROUND(L1645)
1680 BUS SYSTEM FEE = 3*ROUND(L1560 + L1645)
1700    *
1700.1 * STUDENT CENTER FEE CALCULATIONS
1700.9 *
1710 STC PROP PART TIME STUDENTS = DATA
1720 QUARTERLY STUDENT CENTER FEE = DATA
1730 STC OPERATING EXPENSES 000 = DATA
```

```
1735 STC ESCALATOR = DATA
1740 STC INDEX = IF L1735 .LT. 0 THEN L1202 ELSE L1735
1750 CALCULATED STC FEE = 1000*L1730/L1255
1755 PROJECTED STC FEE = L1740*(PREVIOUS L1760)
1760 STC FEE = IF L1720 .NE. 0  THEN L1720'
1765   ELSE IF L1730 .NE. 0 THEN L1750'
1770       ELSE L1755
1780 BASE STUDENT CENTER FEE = 3*ROUND(L1760)
1810 * STUDENT CENTER PROJECT 1
1811 SC1 AMORT     = DATA
1812 SC1 UNAMORT   = DATA
1813 SC1 INT RATE  = DATA
1814 SC1 PR LIFE   = DATA
1815 SC1 PPY       = DATA
1815.5 SC1 NUM     = 0
1816 SC1 START     = DATA
1817 AMORT(SC1 AMORT, SC1 UNAMORT, SC1 INT RATE, SC1 PR LIFE, SC1 START,'
1818    SC1 PPY, STC PAY,INT,PRIN,BAL)
1820 * STUDENT CENTER PROJECT 2
1821 SC2 AMORT     = DATA
1822 SC2 UNAMORT   = DATA
1823 SC2 INT RATE  = DATA
1824 SC2 PR LIFE   = DATA
1825 SC2 START     = DATA
1826 SC2 PPY       = DATA
1827 AMORT(SC2 AMORT, SC2 UNAMORT, SC2 INT RATE, SC2 PR LIFE, SC2 START,'
1828    SC2 PPY, STC PAY,INT,PRIN,BAL)
1830 * STUDENT CENTER PROJECT 3
1831 SC3 AMORT     = DATA
1832 SC3 UNAMORT   = DATA
1833 SC3 INT RATE  = DATA
1834 SC3 PR LIFE   = DATA
1835 SC3 START     = DATA
1836 SC3 PPY       = DATA
1837 AMORT(SC3 AMORT, SC3 UNAMORT, SC3 INT RATE, SC3 PR LIFE, SC3 START,'
1838    SC3 PPY, STC PAY,INT,PRIN,BAL)
1845 QRTLY STC DEBT SERVICE = 1000*(STC PAY)/L1255
1850 DEBT SERVICE = 3*ROUND(L1845)
1880 STUDENT CENTER FEE = 3*ROUND(L1760 + L1845)
1900   *
1900.1 * HEALTH SERVICE FEE CALCULATIONS
1900.9 *
1910 HLT PROP PART TIME STUDENTS = DATA
1920 QUARTERLY HEALTH SERVICE FEE = DATA
1930 HLT OPERATING EXPENSES 000 = DATA
1935 HLT ESCALATOR = DATA
1940 HLT INDEX = IF L1935 .LT. 0 THEN L1202 ELSE L1935
1950 CALCULATED HLT FEE = 1000*L1930/L1270
1955 PROJECTED HLT FEE = L1940*(PREVIOUS L1960)
1960 HLT FEE = IF L1920 .NE. 0  THEN L1920'
1965   ELSE IF L1930 .NE. 0 THEN L1950'
1970       ELSE L1955
1980 BASE HEALTH SERVICE FEE = 3*ROUND(L1960)
2010 * HEALTH SERVICE PROJECT 1
2011 HS1 AMORT     = DATA
```

```
2012 HS1 UNAMORT  = DATA
2013 HS1 INT RATE = DATA
2014 HS1 PR LIFE  = DATA
2015 HS1 START    = DATA
2016 HS1 PPY      = DATA
2017 AMORT(HS1 AMORT, HS1 UNAMORT, HS1 INT RATE, HS1 PR LIFE, HS1 START,'
2018    HS1 PPY, HLT PAY,INT,PRIN,BAL)
2020 * HEALTH SERVICE PROJECT 2
2021 HS2 AMORT    = DATA
2022 HS2 UNAMORT  = DATA
2023 HS2 INT RATE = DATA
2024 HS2 PR LIFE  = DATA
2025 HS2 START    = DATA
2026 HS2 PPY      = DATA
2027 AMORT(HS2 AMORT, HS2 UNAMORT, HS2 INT RATE, HS2 PR LIFE, HS2 START,'
2028    HS2 PPY, HLT PAY,INT,PRIN,BAL)
2030 * HEALTH SERVICE PROJECT 3
2031 HS3 AMORT    = DATA
2032 HS3 UNAMORT  = DATA
2033 HS3 INT RATE = DATA
2034 HS3 PR LIFE  = DATA
2035 HS3 START    = DATA
2036 HS3 PPY      = DATA
2037 AMORT(HS3 AMORT, HS3 UNAMORT, HS3 INT RATE, HS3 PR LIFE, HS3 START,'
2038    HS3 PPY, HLT PAY,INT,PRIN,BAL)
2045 QRTLY HLT DEBT SERVICE = 1000*(HLT PAY)/L1270
2050 DEBT SERVICE = 3*ROUND(L2045)
2080 HEALTH SERVICE FEE = 3*ROUND(L1960 + L2045)
2100 TOTAL ANNUAL FEES = L1480 + L1680 + L1880 + L2080
2110 TOTAL BASE FEES = L1380 + L1580 + L1780 + L1980
2120 TOTAL DEBT SERVICE = L1450 + L1650 + L1850 + L2050
```

```
FILE    MODEL    PEER    10/19/8330:54:04
10 COLUMNS &MODEL COLUMNS&
3100 PERCENTILE = &PERCENTILE&
3105 X = L3240
3110 CHARGE 99  = DATA FOR * , TREND(X)
3120 CHARGE 95  = DATA FOR * , TREND(X)
3130 CHARGE 90  = DATA FOR * , TREND(X)
3140 CHARGE 75  = DATA FOR * , TREND(X)
3150 CHARGE 50  = DATA FOR * , TREND(X)
3160 CHARGE 25  = DATA FOR * , TREND(X)
3170 CHARGE 10  = DATA FOR * , TREND(X)
3180 CHARGE 05  = DATA FOR * , TREND(X)
3190 CHARGE 01  = DATA FOR * , TREND(X)
3200 MEAN       = DATA FOR * , TREND(X)
3210 *CONSUMER PRICE INDEX STORED IN FILE COST IFPS IS FROM 1978-83
3211 *CPI ADDS 1976, 1977 DATA FOR LAG CALCULATIONS
3220 *USING CPI IMPLIES A 2-YEAR LAG
3221 *USING CPI+1 IMPLIES A 1-YEAR LAG
3222 *USING CPI+2 IMPLIES NO LAG
3240 CPI = 172.6, 184.0, PREVIOUS 2 L3250
3250 CONSUMER PRICE INDEX = DATA
3260 CPI!+1 = 184.0, PREVIOUS 1 L3250
3270 CPI!+2 = L3250
3300 &NAME& = IF PERCENTILE .LT. 0 THEN MEAN '
3310   ELSE IF PERCENTILE .EQ. 0 THEN 0 '
3320      ELSE INTERPOLATION ON(PERCENTILE, 99,CHARGE 99, 95, CHARGE 95,'
3330      90, CHARGE 90, 75, CHARGE 75, 50, CHARGE 50, 25, CHARGE 25,'
3340      10, CHARGE 10, 05, CHARGE 05, 01, CHARGE 01)
```

```
FILE     MODEL    TUITION 10/21/8320:15:34
10 COLUMNS &COLUMNS&
6001.1 * TUITION FROM THE FILE TUITION IFPS
6001.2 *  ROUND UNDERGRADUATE CHARGES TO THE NEAREST $0.50 AND GRADUATE
6001.3 *      CHARGES TO THE NEAREST $1.00 PER CREDIT-HOUR.
6005 UG ATH FEE PER CH = L1480/30
6006 GR ATH FEE PER CH = L1480/27
6007 UG BUS FEE PER CH = L1680/30
6008 GR BUS FEE PER CH = L1680/27
6010 ST CAT 1 = DATA
6010.5 TUIT ST CAT 1 = L6010 + L6005 + L6007
6010.6 A = 0
6011 TRUNC ST CAT 1 = TRUNCATE(L6010.5)
6011.1 DUMMY = 0
6012 TRUNC 25 ST CAT 1 = L6011 + 0.25
6013 TRUNC 50 ST CAT 1 = L6011 + 0.50
6014 TRUNC 75 ST CAT 1 = L6011 + 0.75
6015 RND TUIT ST CAT 1 = IF L6010.5 .LT. L6012 THEN L6011'
6016    ELSE IF L6010.5 .GT. L6014 THEN ROUNDUP(L6010.5)'
6017       ELSE L6013
6018 EG ST CAT 1 = L6015 - L6005 - L6007
6020 ST CAT 2 = DATA
6020.5 TUIT ST CAT 2 = L6020 + L6005 + L6007
6021 TRUNC ST CAT 2 = TRUNCATE(L6020.5)
6022 TRUNC 25 ST CAT 2 = L6021 + 0.25
6023 TRUNC 50 ST CAT 2 = L6021 + 0.50
6024 TRUNC 75 ST CAT 2 = L6021 + 0.75
6025 RND TUIT ST CAT 2 = IF L6020.5 .LT. L6022 THEN L6021'
6026    ELSE IF L6020.5 .GT. L6024 THEN ROUNDUP(L6020.5)'
6027       ELSE L6023
6028 EG ST CAT 2 = L6025 - L6005 - L6007
6030 ST CAT 3 = DATA
6030.5 TUIT ST CAT 3 = L6030 + L6006 + L6008
6035 RND ST CAT 3 = ROUND(L6030.5)
6038 EG ST CAT 3 = L6035 - L6006 - L6008
6040 ST CAT 4 = DATA
6040.5 TUIT ST CAT 4 = L6040 + L6006 + L6008
6045 RND ST CAT 4 = ROUND(L6040.5)
6048 EG ST CAT 4 = L6045 - L6006 - L6008
6050 ST CAT 5 = DATA
6050.5 TUIT ST CAT 5 = L6050
6055 RND ST CAT 5 = ROUND(L6050.5)
6058 EG ST CAT 5 = L6055
6060 ST CAT 6 = DATA
6060.5 TUIT ST CAT 6 = L6060
6065 RND ST CAT 6 = ROUND(L6060.5)
6068 EG ST CAT 6 = L6065
6069 * ASSUME THAT ATHLETIC AND BUS FEES ARE ONLY PAID BY FULL-TIME
6070 ATHLETIC REVENUE = SUM(L111,L114)*L390*L1480/3 + '
6071       SUM(L131,L134)*L1480/3
6075 BUS REVENUE = SUM(L111,L114)*L390*L1680/3 + '
6076       SUM(L131,L134)*L1680/3
```

```
6080 EXPECTED EG REVENUE = L710*L6018 + L720*L6028 + L730*L6038 '
6081      + L740*L6048 + L750*L6058 + L760*L6068
6085 * ATHLETIC AND BUS FEES PAID BY ON-CAMPUS PART-TIME STUDENTS
6085.1 * GOES TO GENERAL FUND
6086 EXTRA EG REV FROM PART TIME = '
6087 (L6005 + L6007)*(L321*L121 + L322*L122 + L341*L141 + L342*L142) +'
6088 (L6006 + L6008)*(L323*L123 + L324*L124 + L343*L143 + L344*L144)
6090 DIFFERENCE = L6080 - L5620 + L6086
6100 * CONVERT TO ANNUAL FULL-TIME CHARGES
6110 R UG TUITION = 3*L311*L6018
6120 NR UG TUITION = 3*L311*L6028
6130 R GR TUITION = 3*L312*L6038
6140 NR GR TUITION = 3*L312*L6048
6150 R GR OFF TUITION = 3*L312*L6058
6160 NR GR OFF TUITION = 3*L312*L6068
6205 * PERCENT OF AVERAGE COST CALCULATIONS
6210 R UG AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6110/L5710
6220 NR UG AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6120/L5710
6230 R GR AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6130/L5710
6240 NR GR AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6140/L5710
6250 R GR OFF AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6150/L5710
6260 NR GR OFF AVG COST = IF L5710 .EQ. 0 THEN 0 ELSE 100*L6160/L5710
6300 *TOTAL TUITION AND FEES
6310 R UG TUITION AND FEES = L6110 + L2100
6320 NR UG TUITION AND FEES = L6120 + L2100
6330 R GR TUITION AND FEES = L6130 + L2100
6340 NR GR TUITION AND FEES = L6140 + L2100
6350 R GR OFF TUITION AND FEES = L6150 + L2100
6360 NR GR OFF TUITION AND FEES = L6160 + L2100
6400 *TOTAL TUITION, FEES, ROOM AND BOARD
6410 R UG TUITION FEES ROOM AND BOARD = L6310 + L3030
6420 NR UG TUITION FEES ROOM AND BOARD = L6320 + L3030
6430 R GR TUITION FEES ROOM AND BOARD = IF L3040 .EQ. 0 THEN 0'
6431   ELSE L6330 + L3040
6440 NR GR TUITION FEES ROOM AND BOARD = IF L3040 .EQ. 0 THEN 0'
6441   ELSE L6340 + L3040
6450 R GR OFF TUITION FEES ROOM AND BOARD = IF L3040 .EQ. 0 THEN 0'
6451   ELSE L6350 + L3040
6460 NR GR OFF TUITION FEES ROOM AND BOARD = IF L3040 .EQ. 0 THEN 0'
6461   ELSE L6360 + L3040
6510 UG RES TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6511 IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE ((L6018/PREVIOUS L6018)-1)*100
6520 UG NRES TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6521 IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE ((L6028/PREVIOUS L6028)-1)*100
6530 GR RES TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6531 IF PREVIOUS L6030 .EQ. 0 THEN 0 ELSE ((L6038/PREVIOUS L6038)-1)*100
6540 GR NRES TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6541 IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE ((L6048/PREVIOUS L6048)-1)*100
6550 GR RES OFC TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6551 IF PREVIOUS L6050 .EQ. 0 THEN 0 ELSE ((L6058/PREVIOUS L6058)-1)*100
6560 GR NRES OFC TUIT !% INCREASE OVER PRIOR YEAR = 0,'
6561 IF PREVIOUS L6060 .EQ. 0 THEN 0 ELSE ((L6068/PREVIOUS L6068)-1)*100
7005 BASE = DATA
7010 *RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS
7010.5 UG RES TOT TUIT !& FEES !$ INCREASE OVER PRIOR YEAR = '
```

```
7010.6 0, IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE L6310 - PREVIOUS L6310
7011 UG RES TOT TUIT !& FEES !% INCREASE OVER PRIOR YEAR = 0,'
7012 IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE ((L6310/PREVIOUS L6310)-1)*100
7013 UG RES TOT TUIT !& FEES !% INCREASE OVER BASE YEAR='
7014 0, IF L6010 .EQ. 0 THEN 0 ELSE ((L6310/VMATRIX(L6310,BASE))-1)*100
7015 *RESIDENT UNDERGRADUATE LIVING ON-CAMPUS
7015.5 UG RES TOT TUIT !& FEES!, RM!& BRD !$ INCR OVER PRIOR YR ='
7015.6 0, IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE L6410 - PREVIOUS L6410
7016 UG RES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER PRIOR YR = 0,'
7017 IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE ((L6410/PREVIOUS L6410)-1)*100
7018 UG RES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER BASE YEAR ='
7019 0, IF PREVIOUS L6010 .EQ. 0 THEN 0 ELSE '
7019.5 ((L6410/VMATRIX(L6410,BASE))-1)*100
7020 *NON-RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS
7020.5 UG NRES TOT TUIT !& FEES !$ INCREASE OVER PRIOR YEAR = '
7020.6 0, IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE L6320 - PREVIOUS L6320
7021 UG NRES TOT TUIT !& FEES !% INCREASE OVER PRIOR YEAR= 0,'
7022 IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE ((L6320/PREVIOUS L6320)-1)*100
7023 UG NRES TOT TUIT !& FEES !% INCREASE OVER BASE YEAR='
7024 0, IF L6020 .EQ. 0 THEN 0 ELSE ((L6320/VMATRIX(L6320,BASE))-1)*100
7025 *NON-RESIDENT UNDERGRADUATE LIVING ON-CAMPUS
7025.5 UG NRES TOT TUIT !& FEES!, RM!& BRD !$ INCR OVER PRIOR YR = '
7025.6 0, IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE L6420 - PREVIOUS L6420
7026 UG NRES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER PRIOR YR = 0,'
7027 IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE ((L6420/PREVIOUS L6420)-1)*100
7028 UG NRES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER BASE YEAR ='
7029 0, IF PREVIOUS L6020 .EQ. 0 THEN 0 ELSE '
7029.5 ((L6420/VMATRIX(L6420,BASE))-1)*100
7030 *RESIDENT GRADUATE LIVING OFF-CAMPUS
7030.5 GR RES TOT TUIT !& FEES !$ INCREASE OVER PRIOR YEAR = '
7030.6 0, IF PREVIOUS L6030 .EQ. 0 THEN 0 ELSE L6330 - PREVIOUS L6330
7031 GR RES TOT TUIT !& FEES !% INCREASE OVER PRIOR YEAR= 0,'
7032 IF PREVIOUS L6030 .EQ. 0 THEN 0 ELSE ((L6330/PREVIOUS L6330)-1)*100
7033 GR RES TOT TUIT !& FEES !% INCREASE OVER BASE YEAR='
7034 0, IF L6030 .EQ. 0 THEN 0 ELSE ((L6330/VMATRIX(L6330,BASE))-1)*100
7035 *RESIDENT GRADUATE LIVING ON-CAMPUS
7035.5 GR RES TOT TUIT !& FEES!, RM!& BRD !$ INCR OVER PRIOR YR ='
7035.6 0, IF PREVIOUS L6030 .EQ. 0 THEN 0 ELSE L6430 - PREVIOUS L6430
7036 GR RES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER PRIOR YR = 0,'
7037 IF PREVIOUS L6430 .EQ. 0 THEN 0 ELSE ((L6430/PREVIOUS L6430)-1)*100
7038 GR RES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER BASE YEAR ='
7039 0, IF PREVIOUS L6030 .EQ. 0 THEN 0 ELSE IF VMATRIX(L6430,BASE)'
7039.5 .EQ. 0 THEN 0 ELSE ((L6430/VMATRIX(L6430,BASE))-1)*100
7040 *NON-RESIDENT GRADUATE LIVING OFF-CAMPUS
7040.5 GR NRES TOT TUIT !& FEES !$ INCREASE OVER PRIOR YEAR = '
7040.6 0, IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE L6340 - PREVIOUS L6340
7041 GR NRES TOT TUIT !& FEES !% INCREASE OVER PRIOR YEAR= 0,'
7042 IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE ((L6340/PREVIOUS L6340)-1)*100
7043 GR NRES TOT TUIT !& FEES !% INCREASE OVER BASE YEAR='
7044 0, IF L6040 .EQ. 0 THEN 0 ELSE ((L6340/VMATRIX(L6340,BASE))-1)*100
7045 *NON-RESIDENT GRADUATE LIVING ON-CAMPUS
7045.5 GR NRES TOT TUIT !& FEES!, RM!& BRD !$ INCR OVER PRIOR YR ='
7045.6 0, IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE L6440 - PREVIOUS L6440
7046 GR NRES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER PRIOR YR = 0,'
7047 IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE ((L6440/PREVIOUS L6440)-1)*100
7048 GR NRES TOT TUIT !& FEES!, RM !& BRD !% INCR OVER BASE YEAR ='
7049 0, IF PREVIOUS L6040 .EQ. 0 THEN 0 ELSE IF VMATRIX(L6440,BASE)'
7049.5 .EQ. 0 THEN 0 ELSE ((L6440/VMATRIX(L6440,BASE))-1)*100
```

```
FILE    REPORT  COSTINFO11/13/8323:03:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT.......................... 8<  DDD,DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
30 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
35 CENTER SUMMARY OF COST INFORMATION AND DATA                    &DATE&
40 SPACE
45 COLUMN TITLES
50 UNDERLINE
55 SPACE
60 /ADJUSTED INSTRUCTIONAL E&G/L5610
62 FORMAT.......................... 8<  DDDDDD.DDDD>
65 /TUITION AND FEE RECOVERY RATE (1)/L5470
67 RL20
70 UNDERLINE
71 L5612 $
72 L5616
73 UNDERLINE
75 L5620 $
76 FORMAT.......................... 8<  DDDDDDDD.DD>
77 /CHANGE IN RQMT. OVER PRIOR YEAR/L5650 %
80 SPACE 2
100 *COMPONENTS OF THE TUITION AND FEE RECOVERY RATE
105 *  1. STUDENT ENROLLMENT MIX (2)
108 RL62
110 /    RESIDENT UNDERGRADUATE/L5360
115 /    NON-RESIDENT UNDERGRADUATE/L5370
120 /    RESIDENT GRADUATE/L5380
125 /    NON-RESIDENT GRADUATE/L5390
130 SPACE
135 *    (2)BASED ON THE FOLLOWING WEIGHTED AVG. OF NON-RESIDENT GRADUATE
140 *     STUDENT EMPLOYMENT RATES
145 /     WEIGHTED AVG EMPLOYMENT RATE/L5330
150 SPACE
155 *  2. COST RATIOS
160 /    UNDERGRADUATE/L5220
165 /    GRADUATE/L5230
170 SPACE
175 *  3. POLICY RATES
180 /    RESIDENT UNDERGRADUATE/L5160
185 /    NON-RESIDENT UNDERGRADUATE/L5170
190 /    RESIDENT GRADUATE/L5180
195 /    NON-RESIDENT GRADUATE/L5190
200 SPACE
205 *  4. PARTIAL RECOVERY RATES (BY STUDENT CATEGORY)
210 /    RESIDENT UNDERGRADUATE/L5410
215 /    NON-RESIDENT UNDERGRADUATE/L5420
220 /    RESIDENT GRADUATE/L5430
225 /    NON-RESIDENT GRADUATE/L5440
230 UNDERLINE
```

```
235 /   RECOVERY RATE/L5445
250 NEWPAGE
255 *BASIS FOR PERCENTAGE OF AVERAGE COST GOAL:
265 FORMAT..................................... 8< DDDDDDDDDDD>
270 /  AVG. ANNUAL COST PER FULL-TIME STUDENT/L5710
275 SPACE
280 RL76
285 L5101
290 /CHANGE IN CPI/L5103 %
300 SPACE
310 * (1) MAXIMUM INCREASE .025/YEAR; MAXIMUM DECREASE .0125/YEAR
```

```
FILE    REPORT  CREDIT  12/13/8313:23:18
1 COLUMNS &COLUMNS&
10 PAGELENGTH 0,55,0
20 NEWPAGE
30 FORMAT..... 6<   DDD,DDD>
40 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
46 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
50 CENTER EXPECTED CREDIT HOURS BY STUDENT CATEGORY          &DATE&
60 SPACE 2
70 COLUMN TITLES
80 UNDERLINE
110 SPACE
120 *REGULAR AND SUMMER SESSIONS
130 UNDERLINE
140 /RESIDENT UNDERGRADUATE/L710
150 /NON-RESIDENT UNDERGRADUATE/L720
160 /RESIDENT GRADUATE/L730
170 /NON-RESIDENT GRADUATE/L740
180 SPACE
190 *OFF-CAMPUS
200 UNDERLINE
210 /RESIDENT GRADUATE/L750
220 /NON-RESIDENT GRADUATE/L760
230 SPACE
240 UNDERLINE
250 SPACE
260 /TOTAL EXPECTED CREDIT HOURS/L770
```

```
FILE    REPORT  CREDMPS 10/21/8313:23:18
1 COLUMNS &COLUMNS&
20 PAGELENGTH 0,10000,0
21 NOCOMMAS
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
26 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
27 CENTER EXPECTED CREDIT HOURS PER YEAR (000) BY STUDENT CATEGORY
30 SCALE 1000
35 FORMAT 6< 999999.999>
85 INVERT
90 L910 THRU L960
```

```
FILE    REPORT  DDFEEMPS10/21/8313:23:18
1 COLUMNS &COLUMNS&
20 PAGELENGTH 0,10000,0
21 NOCOMMAS
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
26 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
30 CENTER STUDENT FEE CHARGES BY YEAR                    &DATE&
35 FORMAT.....8<  9999.>
85 COLUMN TITLES
100 L1480
110 L1680
120 L1880
130 L2080
140 L3030
150 L3040
```

```
FILE    REPORT  DORMDING11/13/8323:03:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT.................................. 10<   DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
30 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
35 CENTER DORM AND DINING DATA                    &DATE&
40 SPACE 2
45 COLUMN TITLES
50 UNDERLINE
68 *UNDERGRADUATE STUDENTS
69 UNDERLINE
70 /UPPER QUAD/L2610
75 /LOWER QUAD/L2620
80 /PRAIRIE QUAD/L2630
85 /NEW UNDERGRADUATE DORM/L2640
90 /SPECIAL PURPOSE HOUSING/L2650
95 UNDERLINE
100 /TOTAL UNDERGRADUATE STUDENTS/L2730
105 SPACE
110 *GRADUATE STUDENTS
115 UNDERLINE
150 /GRADUATE SINGLE OCC/L2660
155 /GRADUATE DOUBLE OCC/L2670
160 UNDERLINE
170 /TOTAL GRADUATE STUDENTS/L2740
180 SPACE
190 /TOTAL ON CAMPUS/L2750
200 SPACE
205 FORMAT.................................. 10<   DDDDDDD>
210 *BASE DORMITORY CHARGES (QUARTERLY)
215 UNDERLINE
220 /UPPER QUAD/L2515 $
225 /LOWER QUAD/L2525
230 /PRAIRIE QUAD/L2535
235 /NEW UNDERGRADUATE DORM/L2545
240 /SPECIAL PURPOSE HOUSING/L2555
245 UNDERLINE
270 /UNDERGRADUATE AVERAGE/L2770 $
275 SPACE
285 /GRADUATE SINGLE OCC/L2565
290 /GRADUATE DOUBLE OCC/L2575
293 UNDERLINE
295 /GRADUATE AVERAGE/L2780 $
325 SPACE
330 *BASE DINING CHARGES (QUARTERLY)
335 UNDERLINE
340 /UNDERGRADUATE/L2815 $
350 /GRADUATE/L2845
400 SPACE 2
410 *NEW DEBT SERVICE (QUARTERLY)
```

```
420 UNDERLINE
430 /PROJECT 1/L2919 $
440 /PROJECT 2/L2929
450 /PROJECT 3/L2939
470 UNDERLINE
480 /TOTAL DEBT SERVICE/L2940 $
500 NEWPAGE
510 RL 25 THRU RL 35
515 SPACE
520 CENTER SUMMARY OF ANNUAL CHARGES
525 SPACE
530 COLUMN TITLES
535 UNDERLINE
540 SPACE
550 *UNDERGRADUATE
555 UNDERLINE
560 /BASE DORMITORY/L2775
565 /BASE DINING/L2820
570 UNDERLINE
575 /BASE DORM & DINING/L3010
580 SPACE
585 /DEBT SERVICE/L2950
590 SPACE 2
595 /TOTAL DORM & DINING/L3030
597 UNDERLINE =
600 SPACE 3
650 *GRADUATE
655 UNDERLINE
660 /BASE DORMITORY/L2785
665 /BASE DINING/L2850
670 UNDERLINE
675 /BASE DORM & DINING/L3020
680 SPACE
685 /DEBT SERVICE/L2950
690 SPACE 2
695 /TOTAL DORM & DINING/L3040
697 UNDERLINE =
```

```
FILE    REPORT  ENROLL  06/06/8323:03:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT..................................... 10<    DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
26 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
27 CENTER ENROLLMENT DATA                                  &DATE&
28 SPACE 2
30 CENTER "ACTUAL" ENROLLMENT DATA (HEADCOUNT)
32 UNDERLINE
35 CENTER USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES
70 SPACE 2
80 COLUMN TITLES
90 UNDERLINE
100 *REGULAR SESSION - FALL QUARTER ENROLLMENT
102 UNDERLINE
110 *UNDERGRADUATE
120 /FULL-TIME RESIDENT/L111
130 /FULL-TIME NON-RESIDENT/L112
140 /PART-TIME RESIDENT/L121
150 /PART-TIME NON-RESIDENT/L122
152 UNDERLINE
154 /TOTAL UNDERGRADUATE/L171
160 SPACE
210 *GRADUATE
220 /FULL-TIME RESIDENT/L113
230 /FULL-TIME NON-RESIDENT/L114
240 /PART-TIME RESIDENT/L123
250 /PART-TIME NON-RESIDENT/L124
252 UNDERLINE
254 /TOTAL GRADUATE/L172
255 SPACE
256 /TOTAL REGULAR SESSION/L177
260 SPACE
300 *SUMMER SESSION
301 UNDERLINE
310 *UNDERGRADUATE
320 /FULL-TIME RESIDENT/L131
330 /FULL-TIME NON-RESIDENT/L132
340 /PART-TIME RESIDENT/L141
350 /PART-TIME NON-RESIDENT/L142
352 UNDERLINE
354 /TOTAL UNDERGRADUATE/L173
360 SPACE
370 *GRADUATE
375 /FULL-TIME RESIDENT/L133
376 /FULL-TIME NON-RESIDENT/L134
377 /PART-TIME RESIDENT/L143
378 /PART-TIME NON-RESIDENT/L144
382 UNDERLINE
384 /TOTAL GRADUATE/L174
```

```
385 SPACE
386 /TOTAL SUMMER SESSION/L178
390 NEWPAGE
395 RL 25 THRU RL 90
400 *OFF-CAMPUS
402 UNDERLINE
410 *GRADUATE
420 /FULL-TIME RESIDENT/L153
430 /FULL-TIME NON-RESIDENT/L154
440 /PART-TIME RESIDENT/L163
450 /PART-TIME NON-RESIDENT/L164
452 UNDERLINE
454 /TOTAL OFF-CAMPUS/L176
500 NEWPAGE
510 RL 25 THRU RL 28
520 CENTER "PROJECTED" ENROLLMENT DATA (HEADCOUNT)
530 UNDERLINE
540 CENTER USED TO CALCULATE THE STATE REVENUE REQUIREMENT
550 RL 70 THRU RL 110
620 /FULL-TIME RESIDENT/L211
630 /FULL-TIME NON-RESIDENT/L212
640 /PART-TIME RESIDENT/L221
650 /PART-TIME NON-RESIDENT/L222
652 UNDERLINE
654 /TOTAL UNDERGRADUATE/L271
660 SPACE
710 *GRADUATE
720 /FULL-TIME RESIDENT/L213
730 /FULL-TIME NON-RESIDENT/L214
740 /PART-TIME RESIDENT/L223
750 /PART-TIME NON-RESIDENT/L224
752 UNDERLINE
754 /TOTAL GRADUATE/L272
755 SPACE
756 /TOTAL REGULAR SESSION/L277
760 SPACE
800 *SUMMER SESSION
801 UNDERLINE
810 *UNDERGRADUATE
820 /FULL-TIME RESIDENT/L231
830 /FULL-TIME NON-RESIDENT/L232
840 /PART-TIME RESIDENT/L241
850 /PART-TIME NON-RESIDENT/L242
852 UNDERLINE
854 /TOTAL UNDERGRADUATE/L273
860 SPACE
870 *GRADUATE
875 /FULL-TIME RESIDENT/L233
876 /FULL-TIME NON-RESIDENT/L234
877 /PART-TIME RESIDENT/L243
878 /PART-TIME NON-RESIDENT/L244
882 UNDERLINE
884 /TOTAL GRADUATE/L274
885 SPACE
886 /TOTAL SUMMER SESSION/L278
```

```
890 NEWPAGE
895 RL 25 THRU RL 90
900 *OFF-CAMPUS
902 UNDERLINE
910 *GRADUATE
920 /FULL-TIME RESIDENT/L253
930 /FULL-TIME NON-RESIDENT/L254
940 /PART-TIME RESIDENT/L263
950 /PART-TIME NON-RESIDENT/L264
952 UNDERLINE
954 /TOTAL OFF-CAMPUS/L276
1500 NEWPAGE
1510 RL 25 THRU RL 28
1520 CENTER "PROJECTED" ENROLLMENT (FTE, BASED ON FALL ENROLLMENT)
1530 UNDERLINE
1535 CENTER USED TO CALCULATE STATE REVENUE REQUIREMENT
1540 SPACE 2
1545 COLUMN TITLES
1550 UNDERLINE
1555 *UNDERGRADUATE
1560 UNDERLINE
1565 SPACE
1570 /RESIDENT/L810
1575 /NON-RESIDENT/L820
1580 SPACE
1585 /TOTAL UNDERGRADUATE/L871
1590 SPACE
1600 *GRADUATE
1605 UNDERLINE
1610 SPACE
1615 /RESIDENT/L830
1620 /NON-RESIDENT/L840
1625 SPACE
1630 /RESIDENT OFF-CAMPUS/L850
1635 /NON-RESIDENT OFF-CAMPUS/L860
1640 SPACE
1645 /TOTAL GRADUATE/L872
1650 SPACE 2
1655 /GRAND TOTAL/L870
1660 UNDERLINE =
1665 SPACE 2
1670 /ON CAMPUS/L875
1675 /OFF CAMPUS/L876
1677 SPACE 2
1680 /SUMMER UNDERGRAD/L885
1685 /SUMMER GRADUATE/L886
1690 SPACE 2
1695 /TOTAL ANNUAL FTE/L881
1696 /TOTAL ANNUAL FTE CH/L890
```

```
FILE    REPORT  FEES    11/13/8323:03:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT..................................... 10<   DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
30 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
35 CENTER STUDENT FEE DATA                          &DATE&
40 SPACE 2
45 COLUMN TITLES
50 UNDERLINE
55 SPACE
60 *ANNUAL EQUIVALENT STUDENTS
65 UNDERLINE
70 /ATHLETIC FEE/L1225
75 /BUS FEE/L1240
80 /STUDENT CENTER/L1255
85 /HEALTH SERVICE FEE/L1270
100 SPACE 2
110 *ANNUAL STUDENT FEES
115 UNDERLINE
120 SPACE
125 *ATHLETIC ASSOCIATION
126 /     BASE FEE/L1380
130 /     DEBT SERVICE/L1450
135 /     TOTAL FEE/L1480
140 SPACE
145 *BUS SYSTEM
146 /     BASE FEE/L1580
150 /     DEBT SERVICE/L1650
155 /     TOTAL FEE/L1680
160 SPACE
165 *STUDENT CENTER
166 /     BASE FEE/L1780
170 /     DEBT SERVICE/L1850
175 /     TOTAL FEE/L1880
180 SPACE
185 *HEALTH SERVICES
186 /     BASE FEE/L1980
190 /     DEBT SERVICE/L2050
195 /     TOTAL FEE/L2080
200 SPACE 3
230 L2100 $
240 UNDERLINE =
```

```
FILE     REPORT   INDEXCHG11/13/8323:05:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT.... 8<  DDDDD.DD>
25 *     VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
30 * DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
35 SPACE
40 * SUMMARY OF KEY ECONOMIC INDICATORES THAT ARE HELPFUL IN ESTABLISHING
45 *     THE PERCENTAGE-CHANGE-OVER-TIME GOAL TARGET VALUES
50 SPACE
55 COLUMN TITLES
60 UNDERLINE
65 SPACE
70 * CHANGE IN THE INSTITUTION'S OPERATING EXPENSES OVER THE PREVIOUS
75 /      YEAR/L5660 %
77 SPACE
80 * CHANGE IN THE STATE REVENUE REQUIREMENT OVER THE PREVIOUS
85 /      YEAR/L5650 %
90 SPACE
95 * CHANGE IN THE CONSUMER PRICE INDEX OVER THE PREVIOUS
100 /      YEAR/L5103 %
```

```
FILE     REPORT   PEERMPS 10/31/8313:23:18
1 COLUMNS &DATAFILECOL&
30 PAGELENGTH 0,10000,0
31 NOCOMMAS
35 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
36 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
37 CENTER ESTIMATED PEER INSTITUTION CHARGES BY STUDENT CATEGORY   &DATE&
39 FORMAT.....6<    99999.>
40 COLUMN TITLES
100 &NAME1&
110 &NAME2&
130 &NAME3&
130 &NAME4&
140 &NAME5&
150 &NAME6&
```

```
FILE     REPORT   PERCOST   12/03/8313:23:18
1 COLUMNS &COLUMNS&
20 PAGELENGTH 0,10000,0
21 NOCOMMAS
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
26 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
27 CENTER COEFFICIENTS FOR THE PERCENTAGE OF COST GOALS              &DATE&
35 FORMAT 6< 99999.9999>
85 INVERT
90 L5810 THRU L5860
```

```
TITLE    REPORT  ROUND   11/13/8323:03:38
1 COLUMNS &COLUMNS&
5 PAGELENGTH 0,55,0
10 NEWPAGE
20 FORMAT..................... 6< DDDDDD.DD>
25 *     VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
30 * DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
35 SPACE
40 *         SUMMARY OF TUITION CHARGES PER CREDIT HOUR
45 SPACE 2
50 COLUMN TITLES
55 UNDERLINE
60 SPACE
65 * E&G TUITION CHARGES PER CREDIT-HOUR, CALCULATED BY GP MODEL
66 UNDERLINE
70 /RESIDENT UNDERGRAD/L6010
75 /NON-RESIDENT UNDERGRAD/L6020
80 /RESIDENT GRADUATE/L6030
85 /NON-RESIDENT GRADUATE/L6040
90 /RESIDENT GRADUATE-OFC/L6050
95 /NON-RESIDENT GRADUATE-OFC/L6060
100 SPACE 2
110 *ATHLETIC FEE PER CREDIT HOUR
115 /    UNDERGRADUATE/L6005
120 /    GRADUATE/L6006
130 *BUS FEE PER CREDIT HOUR
135 /    UNDERGRADUATE/L6007
140 /    GRADUATE/L6008
150 SPACE 2
155 * ROUNDED TOTAL CHARGES PER CREDIT-HOUR, E&G + ATH + BUS
160 UNDERLINE
165 /RESIDENT UNDERGRAD/L6015
170 /NON-RESIDENT UNDERGRAD/L6025
175 /RESIDENT GRADUATE/L6035
180 /NON-RESIDENT GRADUATE/L6045
185 /RESIDENT GRADUATE-OFC/L6055
190 /NON-RESIDENT GRADUATE-OFC/L6065
200 SPACE 2
205 * NET E&G TUITION CHARGES PER CREDIT-HOUR AFTER ADDITION OF
206 *    ATHLETIC AND BUS FEES AND AFTER ROUNDING (1)
210 UNDERLINE
215 /RESIDENT UNDERGRAD/L6018
220 /NON-RESIDENT UNDERGRAD/L6028
225 /RESIDENT GRADUATE/L6038
230 /NON-RESIDENT GRADUATE/L6048
235 /RESIDENT GRADUATE-OFC/L6058
236 /NON-RESIDENT GRADUATE-OFC/L6068
240 SPACE 5
241 * (1) UNDERGRADUATE CHARGES PER CREDIT-HOUR ARE ROUNDED TO THE
242 *      NEAREST $ 0.50 AND GRADUATE CHARGES ARE ROUNDED TO THE
243 *      NEAREST $ 1.00.  ROUNDING OCCURS AFTER THE ATHLETIC AND
244 *      BUS FEES ARE ADDED TO E&G
```

```
250 NEWPAGE
255 FORMAT..................... 6< DD,DDD,DDD>
256 CENTER SUMMARY OF EXPECTED ANNUAL REVENUE
257 UNDERLINE
258 COLUMN TITLES
259 UNDERLINE
260 /ATHLETIC FEE REVENUE/L6070 $
265 SPACE 2
266 /BUS FEE REVENUE/L6075 $
267 SPACE 4
270 /STATE REVENUE RQMNT/L5620 $
275 SPACE 2
280 /EXPECTED E&G REVENUE/L6080
290 /EXTRA E&G REV FROM PT/L6086
292 UNDERLINE
295 L6090 $
300 UNDERLINE =
```

```
FILE     REPORT   STATEREV 12/03/8313:23:18
1 COLUMNS &COLUMNS&
20 PAGELENGTH 0,10000,0
21 NOCOMMAS
27 CENTER STATE REVENUE REQUIREMENT GOAL TARGETS, BY YEAR          &DATE&
30 SCALE 1000
35 FORMAT 6< 999999.999>
85 INVERT
90 L9630
```

```
FILE    REPORT  TOTCOST1 12/12/83 14:13:55
1 COLUMNS &TUITION REPORT&
5 PAGELENGTH 0,55,0
10 NEWPAGE
15 FORMAT.............................. 10<      DDD.D>
20 FORMAT.............................. 10<    DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
28 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
30 CENTER SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
40 CENTER FULL-TIME ON-CAMPUS RESIDENT UNDERGRADUATE          &DATE&
45 UNDERLINE
50 SPACE
80 COLUMN TITLES
90 UNDERLINE
100 *TUITION AND FEES:
105 UNDERLINE
110 /TUITION/L6110 $
111 RL15
120 /PERCENT  INCREASE OVER PRIOR YEAR/L6510 %
121 /PERCENT OF AVERAGE COST/L6210 %
122 RL20
123 SPACE
125 *FEES:
130 NAMEINDENT 4
135 L1480
140 L1680
145 L1880
150 L2080
155 NAMEINDENT 0
160 UNDERLINE
170 /A. TOTAL TUITION AND FEES/L6310 $
180 SPACE 2
190 *DORM & DINING:
195 UNDERLINE
210 /BASE ROOM/L2775 $
220 /BASE BOARD/L2820 $
230 /    NEW FACILITIES/L2950
245 UNDERLINE
260 /B. TOTAL ROOM AND BOARD/L3030 $
300 SPACE 4
310 *RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS
311 UNDERLINE
315 /  TOTAL CHARGES (A)/L6310 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7010.5 $
325 RL15
325 /    PERCENT INCREASE OVER PRIOR YEAR/L7011 %
330 /    PERCENT INCREASE OVER BASE YEAR/L7013 %
335 RL20
350 SPACE 2
360 *RESIDENT UNDERGRADUATE LIVING ON-CAMPUS
361 UNDERLINE
362 RL70
365 /  TOTAL CHARGES (A + B)/L6410 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7015.5 $
373 RL15
375 /    PERCENT INCREASE OVER PRIOR YEAR/L7016 %
380 /    PERCENT INCREASE OVER BASE YEAR/L7018 %
385 RL70
```

```
FILE    REPORT  TOTCOST2 12/12/83 14:13:55
1 COLUMNS &TUITION REPORT&
5 PAGELENGTH 0,55,0
10 NEWPAGE
15 FORMAT.................................. 10<      DDD.D>
20 FORMAT.................................. 10<    DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
28 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
30 CENTER SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
40 CENTER FULL-TIME ON-CAMPUS NON-RESIDENT UNDERGRADUATE          &DATE&
45 UNDERLINE
50 SPACE
80 COLUMN TITLES
90 UNDERLINE
100 *TUITION AND FEES:
105 UNDERLINE
110 /TUITION/L6120 $
111 RL15
120 /PERCENT  INCREASE OVER PRIOR YEAR/L6520 %
121 /PERCENT OF AVERAGE COST/L6220 %
122 RL20
123 SPACE
125 *FEES:
130 NAMEINDENT 4
135 L1480
140 L1680
145 L1880
150 L2080
155 NAMEINDENT 0
160 UNDERLINE
170 /A. TOTAL TUITION AND FEES/L6320 $
180 SPACE 2
190 *DORM & DINING:
195 UNDERLINE
210 /BASE ROOM/L2775 $
220 /BASE BOARD/L2820 $
230 /   NEW FACILITIES/L2950
245 UNDERLINE
260 /B. TOTAL ROOM AND BOARD/L3030 $
300 SPACE 4
310 *NON-RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS
311 UNDERLINE
315 /  TOTAL CHARGES (A)/L6320 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7020.5 $
325 RL15
325 /    PERCENT INCREASE OVER PRIOR YEAR/L7021 %
330 /    PERCENT INCREASE OVER BASE YEAR/L7023 %
335 RL20
350 SPACE 2
360 *NON-RESIDENT UNDERGRADUATE LIVING ON-CAMPUS
361 UNDERLINE
362 RL70
365 /  TOTAL CHARGES (A + B)/L6420 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7025.5 $
373 RL15
375 /    PERCENT INCREASE OVER PRIOR YEAR/L7026 %
380 /    PERCENT INCREASE OVER BASE YEAR/L7028 %
385 RL70
```

```
FILE     REPORT  TOTCOST3 12/12/83 14:13:55
1 COLUMNS &TUITION REPORT&
5 PAGELENGTH 0,55,0
10 NEWPAGE
15 FORMAT..................................... 10<      DDD.D>
20 FORMAT..................................... 10<    DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
28 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
30 CENTER SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
40 CENTER FULL-TIME ON-CAMPUS RESIDENT GRADUATE          &DATE&
45 UNDERLINE
50 SPACE
80 COLUMN TITLES
90 UNDERLINE
100 *TUITION AND FEES:
105 UNDERLINE
110 /TUITION/L6130 $
111 RL15
120 /PERCENT  INCREASE OVER PRIOR YEAR/L6530 %
121 /PERCENT OF AVERAGE COST/L6230 %
122 RL20
123 SPACE
125 *FEES:
130 NAMEINDENT 4
135 L1480
140 L1680
145 L1880
150 L2080
155 NAMEINDENT 0
160 UNDERLINE
170 /A. TOTAL TUITION AND FEES/L6330 $
180 SPACE 2
190 *DORM & DINING:
195 UNDERLINE
210 /BASE ROOM/L2785 $
220 /BASE BOARD/L2850 $
230 /   NEW FACILITIES/L2950
245 UNDERLINE
260 /B. TOTAL ROOM AND BOARD/L3040 $
300 SPACE 4
310 *RESIDENT GRADUATE LIVING OFF-CAMPUS
311 UNDERLINE
315 /  TOTAL CHARGES (A)/L6330 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7030.5 $
375 RL15
325 /    PERCENT INCREASE OVER PRIOR YEAR/L7031 %
330 /    PERCENT INCREASE OVER BASE YEAR/L7033 %
335 RL20
350 SPACE 2
360 *RESIDENT GRADUATE LIVING ON-CAMPUS
361 UNDERLINE
362 RL70
365 /  TOTAL CHARGES (A + B)/L6430 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7035.5 $
373 RL15
375 /    PERCENT INCREASE OVER PRIOR YEAR/L7036 %
380 /    PERCENT INCREASE OVER BASE YEAR/L7038 %
385 RL70
1010 NEWPAGE
```

```
FILE     REPORT  TOTCOST4 12/12/83 14:13:55
1 COLUMNS &TUITION REPORT&
5 PAGELENGTH 0,55,0
10 NEWPAGE
15 FORMAT..................................  10<      DDD.D>
20 FORMAT..................................  10<    DDD,DDD>
25 CENTER VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
28 CENTER DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
30 CENTER SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
40 CENTER FULL-TIME ON-CAMPUS NON-RESIDENT GRADUATE          &DATE&
45 UNDERLINE
50 SPACE
80 COLUMN TITLES
90 UNDERLINE
100 *TUITION AND FEES:
105 UNDERLINE
110 /TUITION/L6140 $
111 RL15
120 /PERCENT  INCREASE OVER PRIOR YEAR/L6540 %
121 /PERCENT OF AVERAGE COST/L6240 %
122 RL20
123 SPACE
125 *FEES:
130 NAMEINDENT 4
135 L1480
140 L1680
145 L1880
150 L2080
155 NAMEINDENT 0
160 UNDERLINE
170 /A. TOTAL TUITION AND FEES/L6340 $
180 SPACE 2
190 *DORM & DINING:
195 UNDERLINE
210 /BASE ROOM/L2785 $
220 /BASE BOARD/L2850 $
230 /   NEW FACILITIES/L2950
245 UNDERLINE
260 /B. TOTAL ROOM AND BOARD/L3040 $
300 SPACE 4
310 *NON-RESIDENT GRADUATE LIVING OFF-CAMPUS
311 UNDERLINE
315 /  TOTAL CHARGES (A)/L6340 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7040.5 $
325 RL15
325 /    PERCENT INCREASE OVER PRIOR YEAR/L7041 %
330 /    PERCENT INCREASE OVER BASE YEAR/L7043 %
335 RL20
350 SPACE 2
360 *NON-RESIDENT GRADUATE LIVING ON-CAMPUS
361 UNDERLINE
362 RL70
365 /  TOTAL CHARGES (A + B)/L6440 $
370 /    DOLLAR INCREASE OVER PRIOR YEAR/L7045.5 $
373 RL15
375 /    PERCENT INCREASE OVER PRIOR YEAR/L7046 %
380 /    PERCENT INCREASE OVER BASE YEAR/L7048 %
385 RL70
1010 NEWPAGE
```

Appendix E

FILES NOT DISPLAYED IN THE TEXT

E.1 File CHGTIME IFPS
E.2 File COST IFPS
E.3 File DDBASE1 IFPS
E.4 File DDBASE2 IFPS
E.5 File ENROLOUT IFPS
E.6 File FACTOR IFPS
E.7 File FEEBASE IFPS
E.8 File FILE FT08F001
E.9 File GENIND IFPS
E.10 File PEER IFPS
E.11 File PEER_ IFPS
E.12 File PERCOST2 IFPS
E.13 File SETUP IFPS
E.14 File SUPLMREV IFPS
E.15 File TUITHIST DSS

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
TARGETED PERCENTAGE CHANGE IN TUITION OVER PREVIOUS YEAR

	1983	1984	1985			
RES_UG	18.00	21.00	10.00	00.00	00.00	00.00
NRES_UG	20.00	25.00	10.00	00.00	00.00	00.00
RES_GR	18.00	23.00	10.00	00.00	00.00	00.00
NRES_GR	18.00	23.00	10.00	00.00	00.00	00.00
RES_GR_OEC	22.00	25.00	10.00	00.00	00.00	00.00
NRES_GR_OEC	22.00	25.00	10.00	00.00	00.00	00.00

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*          MISCELLANEOUS COST DATA, HISTORICAL AND PROJECTIONS
*
* ADJ INSTRUCT E&G = ADJUSTED INSTRUCTIONAL E&G; IT IS USED IN:
*    1) STATE REVENUE REQUIREMENT GOALS
*    2) PERCENTAGE OF AVERAGE COST GOALS
*
*                          1978        1979        1980        1981
*                          1982        1983        1984        1985
*                          1986        1987        1988        1989
*                          1990        1991        1992        1993
ADJ INSTRUCT E&G  = 000000000, 000000000, 000000000, 000000000,'
                    095931818, 104515143, 119142901, 127722440,'
                    136877443, 146292266, 156336674, 167258919

*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
CONSUMER PRICE INDEX =              199.3,   223.4,   251.7,   279.3,'
                                    293.3,   301.8, PREVIOUS*1.05


*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
TUITION AND FEE RECOVERY RATE  =  0.2640, 0.2800, 0.2870, 0.3000,'
                                  0.3250, 0.3500, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000
*
* ANNUAL COST FOR A FULL-TIME STUDENT, FROM IFP. (USED IN THE TUITION
*    AND FEE RECOVERY RATE CALCULATION)
*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
UNDERGRADUATE IFP COST PER STUDENT = 00000., 00000., 01586.

*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
GRADUATE IFP COST PER STUDENT =     00000., 00000., 06974.

*
```

```
* POLICY RATES FOR USE IN THE TUITION AND FEE RECOVERY RATE CALCULATIONS
*   DEF: PERCENTAGE OF THE FULL COST OF INSTRUCTION THAT THE STATE
*     EXPECTS EACH STUDENT CATEGORY TO PAY.
*
*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
RESIDENT UNDERGRAD POLICY RATE =    000.0,  000.0,  000.0,  000.0,'
                                    000.0,  000.0,  025.0,  025.0

*
NONRESIDENT UNDERGRAD POLICY RATE =  000.0,  000.0,  000.0,  000.0,'
                                    000.0,  000.0,  075.0,  075.0

*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
RESIDENT GRADUATE  POLICY RATE =    000.0,  000.0,  000.0,  000.0,'
                                    000.0,  000.0,  025.0,  025.0

*
NONRESIDENT GRADUATE  POLICY RATE = 000.0,  000.0,  000.0,  000.0,'
                                    000.0,  000.0,  075.0,  075.0

*
* MAXIMUM ALLOWABLE YEAR-TO-YEAR PERCENTAGE INCREASE AND DECREASE IN
*    THE TUITION AND FEE RECOVERY RATE
*
MAXIMUM RECOVERY RATE INCREASE    = 100.00, 100.00, 100.00, 100.00,'
                                    100.00, 100.00, 002.50, 002.50

*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
MAXIMUM RECOVERY RATE DECREASE    = 100.00, 100.00, 100.00, 100.00,'
                                    100.00, 100.00, 001.25, 001.25

*
MISC FEES = 350000
```

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                    DORM AND DINING DATA
*
* NOTES: 1) IF A CERTAIN CHARGE IS INPUT AS 0, THEN FUTURE CHARGES
*           WILL BE PROJECTED USING THE APPROPRIATE ESCALATOR.
*        2) IF AN ESCALATOR IS INPUT AS A NEGATIVE VALUE, THEN THE
*           CPI WILL BE USED AS THE INDEX TO PROJECT FUTURE VALUES.
*
* DORMITORY CHARGES                1978   1979   1980   1981   1982   1983
*                                  1984   1985   1986   1987   1988   1989
*                                  1990   1991   1992   1993
UPPER QUAD QRTLY CHARGE        = 0000,  0000,  0000,  0000,  0138,  0141,'
                                 0000,  0000,  0000,  0000,  0000,  0000

UPPER QUAD STUDENTS            = 1950,  1950,  1950,  1950,  1950,  1950,'
                                 1950
*
*                                1978   1979   1980   1981   1982   1983
*                                1984   1985   1986   1987   1988   1989
*                                1990   1991   1992   1993
*
LOWER QUAD QRTLY CHARGE        = 0000,  0000,  0000,  0000,  0152,  0159,'
                                 0000,  0000,  0000,  0000,  0000,  0000

LOWER QUAD STUDENTS            = 4380,  4380,  4380,  4380,  4380,  4380,'
                                 4380
*
PRAIRIE QUAD QRTLY CHARGE      = 0000,  0000,  0000,  0000,  0156,  0159,'
                                 0000,  0000,  0000,  0000,  0000,  0000

PRAIRIE QUAD STUDENTS          = 1969,  1969,  1969,  1969,  1969,  1969,'
                                 1969
*
*                                1978   1979   1980   1981   1982   1983
*                                1984   1985   1986   1987   1988   1989
*                                1990   1991   1992   1993
*
NEW UNDERGRAD QRTLY CHARGE     = 0000,  0000,  0000,  0000,  0000,  0220,'
                                 0000,  0000,  0000,  0000,  0000,  0000

NEW UNDERGRAD STUDENTS         = 0000,  0000,  0000,  0000,  0000,  0342,'
                                 0342
*
SPEC PURPOSE QRTLY CHARGE      = 0000,  0000,  0000,  0000,  0000,  0223,'
                                 0000,  0000,  0000,  0000,  0000,  0000

SPEC PURPOSE STUDENTS          = 0000,  0000,  0000,  0000,  0000,  0108,'
                                 0108
*
*                                1978   1979   1980   1981   1982   1983
*                                1984   1985   1986   1987   1988   1989
```

```
*                                  1990   1991   1992   1993
GRAD SINGLE OCC QRTLY CHARGE =   0000,  0000,  0000,  0000,  0000,  0346,'
                                 0000,  0000,  0000,  0000,  0000,  0000

GRAD SINGLE OCC STUDENTS     =   0000,  0000,  0000,  0000,  0000,  0036,'
                                 0036
*
GRAD DOUBLE OCC QRTLY CHARGE =   0000,  0000,  0000,  0000,  0000,  0219,'
                                 0000,  0000,  0000,  0000,  0000,  0000

GRAD DOUBLE OCC STUDENTS     =   0000,  0000,  0000,  0000,  0000,  0120,'
                                 0120
*
DORM CHARGES ESCALATOR       = 1.000, 1.000, 1.000, 1.000, 1.000, 1.000,'
                               1.070, 1.070, 1.070, 1.070, 1.070, 1.070

*
* DINING CHARGES                  1978,  1979,  1980,  1981,  1982,  1983
*                                 1984,  1985,  1986,  1987,  1988,  1989
*                                 1990,  1991,  1992,  1993
*
UG 19 MEAL PLAN QRTLY CHARGE =   0000,  0000,  0000,  0000,  0345,  0345,'
                                 0000,  0000,  0000,  0000,  0000,  0000

*
GR 15 MEAL PLAN QRTLY CHARGE =   0000,  0000,  0000,  0000,  0000,  0395,'
                                 0000,  0000,  0000,  0000,  0000,  0000

*
DINING CHARGES ESCALATOR     = 1.000, 1.000, 1.000, 1.000, 1.000, 1.000,'
                               1.070, 1.070, 1.070, 1.070, 1.070, 1.070
```

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                    DORM AND DINING DATA
*
* NOTES: 1) IF AN OPERATING EXPENSE IS INPUT AS 0, THEN FUTURE CHARGES
*           WILL BE PROJECTED USING THE APPROPRIATE ESCALATOR.
*        2) IF AN ESCALATOR IS INPUT AS A NEGATIVE VALUE, THEN THE
*           CPI WILL BE USED AS THE INDEX TO PROJECT FUTURE VALUES.
*        3) THE RATIOS ARE USED TO ALLOCATE THE OPERATING EXPENSES
*           TO QUARTERLY CHARGES FOR EACH DORMITORY OR DINING GROUP,
*           I.E. IN 1983 TOWER AND PRAIRIE QUADS WERE THE BASE (RATIO
*           SPECIFIED AS 1.000), THE UPPER QUAD CHARGE WAS 11.3% BELOW
*           BASE (RATIO WAS .887), NEW UNDERGRADUATE WAS 38.7% ABOVE
*           THE BASE (RATIO WAS 1.387), ETC.
*
*
*                                  1978       1979       1980       1981
*                                  1982       1983       1984       1985
*   ACADEMIC YEAR                  1986       1987       1988       1989
*                                  1990       1991       1992       1993
*
DORM OPERATING EXPENSES 000   =  0000.0,   0000.0,   0000.0,   0000.0,'
                                  3726.1,   4267.5,   0000.0,   0000.0

DORM CHARGES ESCALATOR        =     1.000,     1.000,     1.000,     1.000,'
                                    1.000,     1.000,     1.070,    -1.000

*
DINING OPERATING EXPENSES 000 =   000.0,   0000.0,   0000.0,   0000.0,'
                                  8589.5,   9240.1,   0000.0,   0000.0

DINING CHARGES ESCALATOR      =     1.000,     1.000,     1.000,     1.000,'
                                    1.000,     1.000,     1.070,    -1.000

*                                1978     1979     1980     1981     1982
*                                1983     1984     1985     1986     1987
*   ACADEMIC YEAR                1988     1989     1990     1991     1992
*                                1993
*
UPPER QUAD RATIO              =  0.000,   0.000,   0.000,   0.000,   0.907,'
                                 0.887

UPPER QUAD STUDENTS           =   1950,    1950,    1950,    1950,    1950,'
                                  1950
*
LOWER QUAD RATIO              =  0.000,   0.000,   0.000,   0.000,   1.000,'
                                 1.000

LOWER QUAD STUDENTS           =   4380,    4380,    4380,    4380,    4380,'
                                  4380
*                                1978     1979     1980     1981     1982
*                                1983     1984     1985     1986     1987
```

```
*    ACADEMIC YEAR                1988    1989    1990    1991    1992
*                                 1993
*
PRAIRIE QUAD RATIO          =    0.000,  0.000,  0.000,  0.000,  1.026,'
                                 1.000
PRAIRIE QUAD STUDENTS       =    1969,   1969,   1969,   1969,   1969,'
                                 1969
*
NEW UNDERGRAD RATIO         =    0.000,  0.000,  0.000,  0.000,  0.000,'
                                 1.387
NEW UNDERGRAD STUDENTS      =    0000,   0000,   0000,   0000,   0000,'
                                  342
*
*                                1978    1979    1980    1981    1982
*                                1983    1984    1985    1986    1987
*    ACADEMIC YEAR               1988    1989    1990    1991    1992
*                                1993
*
SPEC PURPOSE RATIO          =    0.000,  0.000,  0.000,  0.000,  0.000,'
                                 1.402
SPEC PURPOSE STUDENTS       =     108,    108,    108,    108,    108,'
                                  108
*
GRAD SINGLE OCC RATIO       =    0.000,  0.000,  0.000,  0.000,  0.000,'
                                 2.176
GRAD SINGLE OCC STUDENTS    =    0000,   0000,   0000,   0000,   0000,'
                                   36
*
*                                1978    1979    1980    1981    1982
*                                1983    1984    1985    1986    1987
*    ACADEMIC YEAR               1988    1989    1990    1991    1992
*                                1993
*
GRAD DOUBLE OCC RATIO       =    0.000,  0.000,  0.000,  0.000,  0.000,'
                                 1.377
GRAD DOUBLE OCC STUDENTS    =    0000,   0000,   0000,   0000,   0000,'
                                  120
*
UG 19 MEAL PLAN  RATIO      =    0.000,  0.000,  0.000,  0.000,  1.000,'
                                 1.000
*
GR 15 MEAL PLAN  RATIO      =    0.000,  0.000,  0.000,  0.000,  0.000,'
                                 1.150
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)
USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
REGULAR SESSION - FALL QUARTER ENROLLMENT			
UNDERGRADUATE			
FULL-TIME RESIDENT	13,617	13,502	13,447
FULL-TIME NON-RESIDENT	3,993	3,953	3,929
PART-TIME RESIDENT	527	524	519
PART-TIME NON-RESIDENT	73	73	73
TOTAL UNDERGRADUATE	18,210	18,052	17,968
GRADUATE			
FULL-TIME RESIDENT	864	945	982
FULL-TIME NON-RESIDENT	1,713	1,826	1,906
PART-TIME RESIDENT	462	489	511
PART-TIME NON-RESIDENT	279	294	305
TOTAL GRADUATE	3,318	3,554	3,704
TOTAL REGULAR SESSION	21,528	21,606	21,672
SUMMER SESSION			
UNDERGRADUATE			
FULL-TIME RESIDENT	1,997	1,977	1,948
FULL-TIME NON-RESIDENT	432	427	402
PART-TIME RESIDENT	2,473	2,451	2,421
PART-TIME NON-RESIDENT	572	566	559
TOTAL UNDERGRADUATE	5,474	5,421	5,330
GRADUATE			
FULL-TIME RESIDENT	427	447	462
FULL-TIME NON-RESIDENT	614	641	669
PART-TIME RESIDENT	507	530	553
PART-TIME NON-RESIDENT	389	407	425
TOTAL GRADUATE	1,937	2,025	2,109
TOTAL SUMMER SESSION	7,411	7,446	7,439

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)

USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
OFF-CAMPUS			
GRADUATE			
FULL-TIME RESIDENT	62	63	65
FULL-TIME NON-RESIDENT	14	14	14
PART-TIME RESIDENT	1,342	1,355	1,375
PART-TIME NON-RESIDENT	179	181	184
TOTAL OFF-CAMPUS	1,597	1,613	1,638

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"PROJECTED" ENROLLMENT DATA (HEADCOUNT)
USED TO CALCULATE THE STATE REVENUE REQUIREMENT

	1983	1984	1985
REGULAR SESSION - FALL QUARTER ENROLLMENT			
UNDERGRADUATE			
FULL-TIME RESIDENT	13,617	13,502	13,447
FULL-TIME NON-RESIDENT	3,993	3,953	3,929
PART-TIME RESIDENT	527	524	519
PART-TIME NON-RESIDENT	73	73	73
TOTAL UNDERGRADUATE	18,210	18,052	17,968
GRADUATE			
FULL-TIME RESIDENT	864	945	982
FULL-TIME NON-RESIDENT	1,713	1,826	1,906
PART-TIME RESIDENT	462	489	511
PART-TIME NON-RESIDENT	279	294	305
TOTAL GRADUATE	3,318	3,554	3,704
TOTAL REGULAR SESSION	21,528	21,606	21,672
SUMMER SESSION			
UNDERGRADUATE			
FULL-TIME RESIDENT	1,997	1,977	1,948
FULL-TIME NON-RESIDENT	432	427	402
PART-TIME RESIDENT	2,473	2,451	2,421
PART-TIME NON-RESIDENT	572	566	559
TOTAL UNDERGRADUATE	5,474	5,421	5,330
GRADUATE			
FULL-TIME RESIDENT	427	447	462
FULL-TIME NON-RESIDENT	614	641	669
PART-TIME RESIDENT	507	530	553
PART-TIME NON-RESIDENT	389	407	425
TOTAL GRADUATE	1,937	2,025	2,109
TOTAL SUMMER SESSION	7,411	7,446	7,439

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)

USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
OFF-CAMPUS			
GRADUATE			
FULL-TIME RESIDENT	62	63	65
FULL-TIME NON-RESIDENT	14	14	14
PART-TIME RESIDENT	1,342	1,355	1,375
PART-TIME NON-RESIDENT	179	181	184
TOTAL OFF-CAMPUS	1,597	1,613	1,638

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"PROJECTED" ENROLLMENT (FTE, BASED ON FALL ENROLLMENT)
USED TO CALCULATE STATE REVENUE REQUIREMENT

	1983	1984	1985
UNDERGRADUATE			
RESIDENT	14,229	14,110	14,050
NON-RESIDENT	4,090	4,050	4,026
TOTAL UNDERGRADUATE	18,319	18,160	18,076
GRADUATE			
RESIDENT	1,014	1,097	1,142
NON-RESIDENT	1,524	1,622	1,691
RESIDENT OFF-CAMPUS	534	539	547
NON-RESIDENT OFF-CAMPUS	73	74	75
TOTAL GRADUATE	3,145	3,332	3,456
GRAND TOTAL	21,464	21,492	21,532
ON CAMPUS	20,858	20,879	20,909
OFF CAMPUS	607	613	622
SUMMER UNDERGRAD	1,303	1,290	1,269
SUMMER GRADUATE	461	482	502
TOTAL ANNUAL FTE	64,303	64,393	64,510
TOTAL ANNUAL FTE CH	935,139	934,805	935,408

```
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*       DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*              MISCELLANEOUS ENROLLMENT FACTOR DATA
*
*
** NUMBER OF CREDIT HOURS TO BE CONSIDERED A FULL-TIME STUDENT
*
CREDIT HOUR FT UNDERGRAD = 10
CREDIT HOUR FT GRADUATE  =  9
*
** AVERAGE NUMBER OF CREDIT HOURS FOR PART-TIME STUDENTS, PER QUARTER
*   REGULAR SESSION
CREDIT HOUR REG PT UG RES  = 5.5
CREDIT HOUR REG PT UG NRES = 5.5
CREDIT HOUR REG PT GR RES  = 4.1
CREDIT HOUR REG PT GR NRES = 4.1
*   SUMMER SESSION
CREDIT HOUR SUM PT UG RES  = 6
CREDIT HOUR SUM PT UG NRES = 6
CREDIT HOUR SUM PT GR RES  = 5.4
CREDIT HOUR SUM PT GR NRES = 5.4
*   OFF-CAMPUS (OFC)
CREDIT HOUR OFC PT UG RES  = 0
CREDIT HOUR OFC PT UG NRES = 0
CREDIT HOUR OFC PT GR RES  = 3
CREDIT HOUR OFC PT GR NRES = 3
*
* AVERAGE CREDIT HOURS PER STUDENT PER QUARTER (FULL-TIME AND
*     PART-TIME COMBINED, BASED ON FALL-QUARTER ENROLLMENTS)
*
**   REGULAR SESSION
AVERAGE CREDIT HOUR REG UG = 15*1.006
AVERAGE CREDIT HOUR REG GR = 12*0.765
***    SUMMER SESSION
AVERAGE CREDIT HOUR SUM UG = 15*0.238
AVERAGE CREDIT HOUR SUM GR = 12*0.238
**      OFF-CAMPUS (OFC)
AVERAGE CREDIT HOUR OFC UG = 0
AVERAGE CREDIT HOUR OFC GR = 12*.38
** RETENTION FACTORS - FALL(1.00) + WINTER + SPRING
*
RETENTION FACTOR!-REG SESSION = 1.000 + .966 + .938
RETENTION FACTOR!-OFC = 1.000 + .873 + .806 + .565
*
*
** FRACTION OF STUDENTS RECEIVING TUITION WAIVERS
*
FEE WAIVER REG FT GR NRES = 0.75
FEE WAIVER REG PT GR NRES = 0.75
FEE WAIVER SUM FT GR NRES = 0.75
FEE WAIVER SUM PT GR NRES = 0.75
FEE WAIVER OFC FT GR NRES = 0.75
```

```
FEE WAIVER OFC PT GR NRES = 0.75
*
*
** GRADUATE STUDENT EMPLOYMENT RATES
*     PROPORTION OF NON-RESIDENT GRADUATE STUDENTS THAT ARE SIGNIFICANTLY
*      EMPLOYED IN THE INSTRUCTIONAL PROGRAM.
*      (USED IN THE TUITION AND FEE RECOVERY RATE CALCULATION)
*
*                              1978       1979       1980       1981
*                              1982       1983       1984       1985
*                              1986       1987       1988       1989
*                              1990       1991       1992       1993
NR GR EMPLOYMENT RATE =       .700,      .650,      .600,      .600,
                              .570
*
* MAXIMUM LEVEL OF GRADUATE STUDENT EMPLOYMENT RECOGNIZED BY THE
*    STATE'S TUITION AND FEE RECOVERY RATE FORMULA
MAX NR GR EMPLOYMENT RATE = 0.500
```

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                        STUDENT FEE DATA
*
* NOTES: 1) IF A CERTAIN FEE IS INPUT AS 0, THEN FUTURE FEE CHARGES
*           WILL BE BASED ON THE OPERATING EXPENSES.
*        2) IF AN OPERATING EXPENSE IS INPUT AS 0, THEN FUTURE OPERATING
*           EXPENSES WILL BE PROJECTED USING THE APPROPRIATE ESCALATOR.
*        3) IF AN ESCALATOR IS INPUT AS A NEGATIVE VALUE, THEN THE
*           CPI WILL BE USED AS THE INDEX TO PROJECT FUTURE VALUES.
*
*
*
*
*
*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
QUARTERLY ATHLETIC FEE             = 000.00, 000.00, 018.00, 020.00,'
                                     022.00, 024.00, 026.00, 028.00,'
                                     030.00, 032.00, 034.00, 036.00

   ATH OPERATING EXPENSES 000      = 00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.

   ATH ESCALATOR                   = 1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000, 1.0000

   ATH PROP PART TIME STUDENTS     = 0.1000, 0.1000, 0.1000, 0.1000,'
                                     0.1000, 0.1000, 0.1000, 0.1000,'
                                     0.1000, 0.1000, 0.1000, 0.1000

*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
QUARTERLY BUS SYSTEM FEE           = 000.00, 000.00, 000.00, 000.00,'
                                     004.00, 004.00, 005.00, 000.00,'
                                     000.00, 000.00, 000.00, 000.00

   BUS OPERATING EXPENSES 000      = 00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.

   BUS ESCALATOR                   = 1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000,-1.0000,'
                                    -1.0000,-1.0000,-1.0000,-1.0000
```

```
   BUS PROP PART TIME STUDENTS  = 0.0000, 0.0000, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000

*
*                                  1978    1979    1980    1981
*                                  1982    1983    1984    1985
*                                  1986    1987    1988    1989
*                                  1990    1991    1992    1993
QUARTERLY STUDENT CENTER FEE    = 000.00, 000.00, 015.00, 018.00,'
                                  000.00, 000.00, 000.00, 000.00,'
                                  000.00, 000.00, 000.00, 000.00

   STC OPERATING EXPENSES 000   = 01140., 01200., 01275., 01350.,'
                                  00000., 00000., 00000., 00000.,'
                                  00000., 00000., 00000., 00000.

   STC ESCALATOR                = 1.0000, 1.0000, 1.0000, 1.0000,'
                                  1.1000, 1.0700, 1.0700, 1.0700,'
                                  1.0700, 1.0700, 1.0700, 1.0700

   STC PROP PART TIME STUDENTS  = 0.5000, 0.5000, 0.5000, 0.5000,'
                                  0.5000, 0.5000, 0.5000, 0.5000,'
                                  0.5000, 0.5000, 0.5000, 0.5000

*
*                                  1978    1979    1980    1981
*                                  1982    1983    1984    1985
*                                  1986    1987    1988    1989
*                                  1990    1991    1992    1993
QUARTERLY HEALTH SERVICE FEE    = 000.00, 000.00, 014.00, 020.00,'
                                  022.00, 023.00, 024.00, 026.00,'
                                  028.00, 030.00, 031.00, 031.00

   HLT OPERATING EXPENSES 000   = 00000., 00000., 00000., 00000.,'
                                  00000., 00000., 00000., 00000.,'
                                  00000., 00000., 00000., 00000.

   HLT ESCALATOR                = 1.0000, 1.0000, 1.0000, 1.0000,'
                                  1.0000, 1.0000, 1.0000, 1.0000,'
                                  1.0000, 1.0000, 1.0000, 1.0000

   HLT PROP PART TIME STUDENTS  = 0.0000, 0.0000, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000,'
                                  0.0000, 0.0000, 0.0000, 0.0000
```

```
NAME    TUITION PROBLEM DATA
ROWS
   N  ROW10001
   E  ROW10002
   E  ROW10003
   E  ROW10004
   E  ROW10005
   E  ROW10006
   E  ROW10007
   E  ROW10008
   E  ROW10009
   E  ROW10010
   E  ROW10011
   E  ROW10012
   E  ROW10013
   E  ROW10014
   E  ROW10015
   E  ROW10016
   E  ROW10017
   E  ROW10018
   E  ROW10019
   E  ROW10020
   E  ROW10021
   E  ROW10022
   E  ROW10023
   E  ROW10024
   E  ROW10025
   E  ROW10026
   E  ROW10027
   E  ROW10028
   E  ROW10029
   E  ROW10030
   E  ROW10031
   E  ROW10032
   E  ROW10033
   E  ROW10034
   E  ROW10035
   E  ROW10036
   E  ROW10037
   E  ROW10038
   E  ROW10039
   E  ROW10040
   E  ROW10041
   E  ROW10042
   E  ROW10043
   E  ROW10044
   E  ROW10045
   E  ROW10046
   E  ROW10047
   E  ROW10048
   E  ROW10049
   E  ROW10050
```

```
 E  ROW10051
 E  ROW10052
 E  ROW10053
 E  ROW10054
 E  ROW10055
 E  ROW10056
 E  ROW10057
 E  ROW10058
 E  ROW10059
 E  ROW10060
 E  ROW10061
 E  ROW10062
 E  ROW10063
 E  ROW10064
 E  ROW10065
 E  ROW10066
 E  ROW10067
 E  ROW10068
 E  ROW10069
 E  ROW10070
 E  ROW10071
 E  ROW10072
 E  ROW10073
 E  ROW10074
 E  ROW10075
 E  ROW10076
 E  ROW10077
 E  ROW10078
 E  ROW10079
 E  ROW10080
 E  ROW10081
 E  ROW10082
 E  ROW10083
 E  ROW10084
 E  ROW10085
 E  ROW10086
 E  ROW10087
 E  ROW10088
 E  ROW10089
 E  ROW10090
 E  ROW10091
 E  ROW10092
 E  ROW10093
 E  ROW10094
 E  ROW10095
 E  ROW10096
 E  ROW10097
 E  ROW10098
COLUMNS
        N001   ROW10002          1.000
        P001   ROW10002         -1.000
        P001   ROW10005          1.000
        P001   ROW10098          1.000
        N002   ROW10003          1.000
        P002   ROW10003         -1.000
```

P002	ROW10006	1.000
P002	ROW10098	1.000
N003	ROW10004	1.000
P003	ROW10004	-1.000
P003	ROW10007	1.000
P003	ROW10098	1.000
N004	ROW10005	1.000
P004	ROW10005	-1.000
N005	ROW10006	1.000
P005	ROW10006	-1.000
N006	ROW10007	1.000
P006	ROW10007	-1.000
N007	ROW10008	1.000
P007	ROW10008	-1.000
N008	ROW10009	1.000
P008	ROW10009	-1.000
N009	ROW10010	1.000
P009	ROW10010	-1.000
N010	ROW10011	1.000
P010	ROW10011	-1.000
N011	ROW10012	1.000
P011	ROW10012	-1.000
N012	ROW10013	1.000
P012	ROW10013	-1.000
N013	ROW10014	1.000
P013	ROW10014	-1.000
N014	ROW10015	1.000
P014	ROW10015	-1.000
N015	ROW10016	1.000
P015	ROW10016	-1.000
N016	ROW10017	1.000
P016	ROW10017	-1.000
N017	ROW10018	1.000
P017	ROW10018	-1.000
N018	ROW10019	1.000
P018	ROW10019	-1.000
N019	ROW10020	1.000
P019	ROW10020	-1.000
N020	ROW10021	1.000
P020	ROW10021	-1.000
N021	ROW10022	1.000
P021	ROW10022	-1.000
N022	ROW10023	1.000
P022	ROW10023	-1.000
N023	ROW10024	1.000
P023	ROW10024	-1.000
N024	ROW10025	1.000
P024	ROW10025	-1.000
N025	ROW10026	1.000
P025	ROW10026	-1.000
N026	ROW10027	1.000
P026	ROW10027	-1.000
N027	ROW10028	1.000
P027	ROW10028	-1.000
N028	ROW10029	1.000

P028	ROW10029	-1.000
N029	ROW10030	1.000
P029	ROW10030	-1.000
N030	ROW10031	1.000
P030	ROW10031	-1.000
N031	ROW10032	1.000
P031	ROW10032	-1.000
N032	ROW10033	1.000
P032	ROW10033	-1.000
N033	ROW10034	1.000
P033	ROW10034	-1.000
N034	ROW10035	1.000
P034	ROW10035	-1.000
N035	ROW10036	1.000
P035	ROW10036	-1.000
N036	ROW10037	1.000
P036	ROW10037	-1.000
N037	ROW10038	1.000
P037	ROW10038	-1.000
N038	ROW10039	1.000
P038	ROW10039	-1.000
N039	ROW10040	1.000
P039	ROW10040	-1.000
N040	ROW10041	1.000
P040	ROW10041	-1.000
N041	ROW10042	1.000
P041	ROW10042	-1.000
N042	ROW10043	1.000
P042	ROW10043	-1.000
N043	ROW10044	1.000
P043	ROW10044	-1.000
N044	ROW10045	1.000
P044	ROW10045	-1.000
N045	ROW10046	1.000
P045	ROW10046	-1.000
N046	ROW10047	1.000
P046	ROW10047	-1.000
N047	ROW10048	1.000
P047	ROW10048	-1.000
N048	ROW10049	1.000
P048	ROW10049	-1.000
N049	ROW10050	1.000
P049	ROW10050	-1.000
N050	ROW10051	1.000
P050	ROW10051	-1.000
N051	ROW10052	1.000
P051	ROW10052	-1.000
N052	ROW10053	1.000
P052	ROW10053	-1.000
N053	ROW10054	1.000
P053	ROW10054	-1.000
N054	ROW10055	1.000
P054	ROW10055	-1.000
N055	ROW10056	1.000
P055	ROW10056	-1.000

N056	ROW10057	1.000
P056	ROW10057	-1.000
N057	ROW10058	1.000
P057	ROW10058	-1.000
N058	ROW10059	1.000
P058	ROW10059	-1.000
N059	ROW10060	1.000
P059	ROW10060	-1.000
N060	ROW10061	1.000
P060	ROW10061	-1.000
N061	ROW10062	1.000
P061	ROW10062	-1.000
N062	ROW10063	1.000
P062	ROW10063	-1.000
N063	ROW10064	1.000
P063	ROW10064	-1.000
N064	ROW10065	1.000
P064	ROW10065	-1.000
N065	ROW10066	1.000
P065	ROW10066	-1.000
N066	ROW10067	1.000
P066	ROW10067	-1.000
N067	ROW10068	1.000
P067	ROW10068	-1.000
N068	ROW10069	1.000
P068	ROW10069	-1.000
N069	ROW10070	1.000
P069	ROW10070	-1.000
N070	ROW10071	1.000
P070	ROW10071	-1.000
N071	ROW10072	1.000
P071	ROW10072	-1.000
N072	ROW10073	1.000
P072	ROW10073	-1.000
N073	ROW10074	1.000
P073	ROW10074	-1.000
N074	ROW10075	1.000
P074	ROW10075	-1.000
N075	ROW10076	1.000
P075	ROW10076	-1.000
N076	ROW10077	1.000
P076	ROW10077	-1.000
N077	ROW10078	1.000
P077	ROW10078	-1.000
N078	ROW10079	1.000
P078	ROW10079	-1.000
N079	ROW10080	1.000
P079	ROW10080	-1.000
N080	ROW10081	1.000
P080	ROW10081	-1.000
N081	ROW10082	1.000
P081	ROW10082	-1.000
N082	ROW10083	1.000
P082	ROW10083	-1.000
N083	ROW10084	1.000

P083	ROW10084	-1.000
N084	ROW10085	1.000
P084	ROW10085	-1.000
N085	ROW10086	1.000
P085	ROW10086	-1.000
N086	ROW10087	1.000
P086	ROW10087	-1.000
N087	ROW10088	1.000
P087	ROW10088	-1.000
N088	ROW10089	1.000
P088	ROW10089	-1.000
N089	ROW10090	1.000
P089	ROW10090	-1.000
N090	ROW10091	1.000
P090	ROW10091	-1.000
N091	ROW10092	1.000
P091	ROW10092	-1.000
N092	ROW10093	1.000
P092	ROW10093	-1.000
N093	ROW10094	1.000
P093	ROW10094	-1.000
N094	ROW10095	1.000
P094	ROW10095	-1.000
N095	ROW10096	1.000
P095	ROW10096	-1.000
N096	ROW10097	1.000
P096	ROW10097	-1.000
N097	ROW10098	1.000
P097	ROW10098	-1.000
X11	ROW10002	438.663
X11	ROW10008	0.615
X11	ROW10026	1.000
X11	ROW10032	-1.210
X11	ROW10044	30.000
X11	ROW10062	1.000
X11	ROW10080	-2.000
X11	ROW10083	-1.200
X21	ROW10002	124.875
X21	ROW10009	0.615
X21	ROW10027	1.000
X21	ROW10033	-1.250
X21	ROW10045	30.000
X21	ROW10063	1.000
X21	ROW10080	1.000
X31	ROW10002	76.453
X31	ROW10010	0.554
X31	ROW10028	1.000
X31	ROW10034	-1.230
X31	ROW10046	27.000
X31	ROW10064	1.000
X31	ROW10083	1.000
X31	ROW10086	-1.000
X31	ROW10089	-1.000
X31	ROW10092	-1.000
X31	ROW10095	-1.000

X41	ROW10002	13.930
X41	ROW10011	0.554
X41	ROW10029	1.000
X41	ROW10035	-1.230
X41	ROW10047	27.000
X41	ROW10065	1.000
X41	ROW10086	1.000
X41	ROW10089	1.000
X41	ROW10095	1.000
X51	ROW10002	16.484
X51	ROW10012	0.554
X51	ROW10030	1.000
X51	ROW10036	-1.250
X51	ROW10048	3.000
X51	ROW10066	1.000
X51	ROW10092	1.000
X51	ROW10095	1.000
X61	ROW10002	0.538
X61	ROW10013	0.554
X61	ROW10031	1.000
X61	ROW10037	-1.250
X61	ROW10049	3.000
X61	ROW10067	1.000
X61	ROW10095	-1.000
X12	ROW10003	434.943
X12	ROW10014	0.540
X12	ROW10032	1.000
X12	ROW10038	-1.100
X12	ROW10050	30.000
X12	ROW10068	1.000
X12	ROW10081	-2.000
X12	ROW10084	-1.200
X22	ROW10003	123.627
X22	ROW10015	0.540
X22	ROW10033	1.000
X22	ROW10039	-1.100
X22	ROW10051	30.000
X22	ROW10069	1.000
X22	ROW10081	1.000
X32	ROW10003	81.799
X32	ROW10016	0.486
X32	ROW10034	1.000
X32	ROW10040	-1.100
X32	ROW10052	27.000
X32	ROW10070	1.000
X32	ROW10084	1.000
X32	ROW10087	-1.000
X32	ROW10090	-1.000
X32	ROW10093	-1.000
X32	ROW10096	-1.000
X42	ROW10003	14.798
X42	ROW10017	0.486
X42	ROW10035	1.000
X42	ROW10041	-1.100
X42	ROW10053	27.000

X42	ROW10071	1.000
X42	ROW10087	1.000
X42	ROW10090	1.000
X42	ROW10096	1.000
X52	ROW10003	16.654
X52	ROW10018	0.486
X52	ROW10036	1.000
X52	ROW10042	-1.100
X52	ROW10054	3.000
X52	ROW10072	1.000
X52	ROW10093	1.000
X52	ROW10096	1.000
X62	ROW10003	0.543
X62	ROW10019	0.486
X62	ROW10037	1.000
X62	ROW10043	-1.100
X62	ROW10055	3.000
X62	ROW10073	1.000
X62	ROW10096	-1.000
X13	ROW10004	432.796
X13	ROW10020	0.505
X13	ROW10038	1.000
X13	ROW10056	30.000
X13	ROW10074	1.000
X13	ROW10082	-2.000
X13	ROW10085	-1.200
X23	ROW10004	122.638
X23	ROW10021	0.505
X23	ROW10039	1.000
X23	ROW10057	30.000
X23	ROW10075	1.000
X23	ROW10082	1.000
X33	ROW10004	85.216
X33	ROW10022	0.455
X33	ROW10040	1.000
X33	ROW10058	27.000
X33	ROW10076	1.000
X33	ROW10085	1.000
X33	ROW10088	-1.000
X33	ROW10091	-1.000
X33	ROW10094	-1.000
X33	ROW10097	-1.000
X43	ROW10004	15.441
X43	ROW10023	0.455
X43	ROW10041	1.000
X43	ROW10059	27.000
X43	ROW10077	1.000
X43	ROW10088	1.000
X43	ROW10091	1.000
X43	ROW10097	1.000
X53	ROW10004	16.929
X53	ROW10024	0.455
X53	ROW10042	1.000
X53	ROW10060	3.000
X53	ROW10078	1.000

```
    X53       ROW10094          1.000
    X53       ROW10097          1.000
    X63       ROW10004          0.550
    X63       ROW10025          0.455
    X63       ROW10043          1.000
    X63       ROW10061          3.000
    X63       ROW10079          1.000
    X63       ROW10097         -1.000
RHS
    RHS       ROW10002      36230.301
    RHS       ROW10003      43559.004
    RHS       ROW10004      46762.992
    RHS       ROW10005        400.000
    RHS       ROW10006        500.000
    RHS       ROW10007        600.000
    RHS       ROW10008         25.000
    RHS       ROW10009         50.000
    RHS       ROW10010         30.000
    RHS       ROW10011         35.000
    RHS       ROW10012         40.000
    RHS       ROW10013         45.000
    RHS       ROW10014         28.000
    RHS       ROW10015         60.000
    RHS       ROW10016         33.000
    RHS       ROW10017         48.000
    RHS       ROW10018         40.000
    RHS       ROW10019         45.000
    RHS       ROW10020         30.000
    RHS       ROW10021         65.000
    RHS       ROW10022         35.000
    RHS       ROW10023         40.000
    RHS       ROW10024         40.000
    RHS       ROW10025         45.000
    RHS       ROW10026         42.126
    RHS       ROW10027         92.640
    RHS       ROW10028         52.970
    RHS       ROW10029         58.870
    RHS       ROW10030         65.880
    RHS       ROW10031         71.980
    RHS       ROW10032          0.0
    RHS       ROW10033          0.0
    RHS       ROW10034          0.0
    RHS       ROW10035          0.0
    RHS       ROW10036          0.0
    RHS       ROW10037          0.0
    RHS       ROW10038          0.0
    RHS       ROW10039          0.0
    RHS       ROW10040          0.0
    RHS       ROW10041          0.0
    RHS       ROW10042          0.0
    RHS       ROW10043          0.0
    RHS       ROW10044       1795.000
    RHS       ROW10045       2740.000
    RHS       ROW10046       1224.000
    RHS       ROW10047       2140.000
```

```
    RHS       ROW10048      200.000
    RHS       ROW10049      280.000
    RHS       ROW10050     1790.000
    RHS       ROW10051     2703.000
    RHS       ROW10052     1227.000
    RHS       ROW10053     2166.000
    RHS       ROW10054      225.000
    RHS       ROW10055      310.000
    RHS       ROW10056     1787.000
    RHS       ROW10057     2719.000
    RHS       ROW10058     1253.000
    RHS       ROW10059     2209.000
    RHS       ROW10060      250.000
    RHS       ROW10061      340.000
    RHS       ROW10062        0.0
    RHS       ROW10063        0.0
    RHS       ROW10064        0.0
    RHS       ROW10065        0.0
    RHS       ROW10066        0.0
    RHS       ROW10067        0.0
    RHS       ROW10068        0.0
    RHS       ROW10069        0.0
    RHS       ROW10070        0.0
    RHS       ROW10071        0.0
    RHS       ROW10072        0.0
    RHS       ROW10073        0.0
    RHS       ROW10074        0.0
    RHS       ROW10075        0.0
    RHS       ROW10076        0.0
    RHS       ROW10077        0.0
    RHS       ROW10078        0.0
    RHS       ROW10079        0.0
    RHS       ROW10080        0.0
    RHS       ROW10081        0.0
    RHS       ROW10082        0.0
    RHS       ROW10083        0.0
    RHS       ROW10084        0.0
    RHS       ROW10085        0.0
    RHS       ROW10086        5.000
    RHS       ROW10087        8.000
    RHS       ROW10088       10.000
    RHS       ROW10089       10.000
    RHS       ROW10090       16.000
    RHS       ROW10091       20.000
    RHS       ROW10092       10.000
    RHS       ROW10093       15.000
    RHS       ROW10094       20.000
    RHS       ROW10095        0.0
    RHS       ROW10096        0.0
    RHS       ROW10097        0.0
    RHS       ROW10098     2000.000
ENDATA
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
GOAL TARGETS FOR GENERAL INDIVIDUAL GOALS

	1983	1984	1985			
RES_UG	000.00	000.00	000.00	000.00	000.00	000.00
NRES_UG	000.00	000.00	000.00	000.00	000.00	000.00
RES_GR	000.00	000.00	000.00	000.00	000.00	000.00
NRES_GR	000.00	000.00	000.00	000.00	000.00	000.00
RES_GR_OFC	000.00	000.00	000.00	000.00	000.00	000.00
NRES_GR_OFC	000.00	000.00	000.00	000.00	000.00	000.00

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
PEER INSTITUTION GOAL TARGET VALUES

YEAR	1983	1984	1985
****	RESIDENT UNDERGRADUATE		
EXPECTED PEER CHARGES	3549.	3789.	3894.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	1538.	1702.	1786.
PEER INST GOAL TARGET	1795.	1790.	1787.
****	NON-RESIDENT UNDERGRADUATE		
EXPECTED PEER CHARGES	4494.	4702.	4826.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	1538.	1702.	1786.
PEER INST GOAL TARGET	2740.	2703.	2719.
****	RESIDENT GRADUATE		
EXPECTED PEER CHARGES	1440.	1524.	1574.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	1224.	1227.	1253.
****	NON-RESIDENT GRADUATE		
EXPECTED PEER CHARGES	2356.	2463.	2530.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	2140.	2166.	2209.
****	RESIDENT GRADUATE OFF-CAMPUS		
EXPECTED PEER CHARGES	200.	225.	250.
LESS FEE CHARGES	0.	0.	0.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	200.	225.	250.
****	NON-RESIDENT GRADUATE OFF-CAMPUS		
EXPECTED PEER CHARGES	280.	310.	340.
LESS FEE CHARGES	0.	0.	0.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	280.	310.	340.

```
MICRO &NAME&RES UG       &
MICRO &PERCENTILE& -1.00&
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                  PEER INSTITUTION DATA FOR:
*RESIDENT UNDERGRAD TUITION, FEES, RM & BRD AT LAND-GRANT INSTITUTIONS
*
*              1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988
CHARGE 99    = 4115,4466,4970,5700,6418,6990
CHARGE 95    = 3100,3393,3481,4186,4777,5137
CHARGE 90    = 2916,3130,3323,3879,4376,4756
CHARGE 75    = 2611,2743,3049,3413,3827,4052
CHARGE 50    = 2170,2299,2431,2827,3093,3315
CHARGE 25    = 1947,2092,2134,2513,2777,2966
CHARGE 10    = 1728,1922,2044,2353,2532,2743
CHARGE 05    = 1660,1861,1983,2247,2413,2507
CHARGE 01    = 1345,1775,1922,2050,2222,1647
MEAN         = 2265,2440,2593,3018,3319,3549
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
GOAL TARGETS FOR PERCENTAGE OF AVG. TOTAL COST GOALS

	1983	1984	1985			
RES_UG	25.00	28.00	30.00	00.00	00.00	00.00
NRES_UG	50.00	60.00	65.00	00.00	00.00	00.00
RES_GR	30.00	33.00	35.00	00.00	00.00	00.00
NRES_GR	35.00	48.00	40.00	00.00	00.00	00.00
RES_GR_OFC	40.00	40.00	40.00	00.00	00.00	00.00
NRES_GR_OFC	45.00	45.00	45.00	00.00	00.00	00.00

```
MICRO &COLUMNS&1983-1985&
MICRO &TUITION REPORT&1982-1985&
MICRO &MODEL COLUMNS&1978-1993&
MICRO &DATAFILECOL&6-8&
MICRO &NAME1&RES UG          &
MICRO &NAME2&NRES UG         &
MICRO &NAME3&RES GR          &
MICRO &NAME4&NRES GR         &
MICRO &NAME5&RES GR OFC      &
MICRO &NAME6&NRES GR OFC     &
BASE = 5
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUPPLEMENTAL REVENUE GOAL TARGET VALUES

BY YEAR (IN DOLLARS)	
1983	000400000.
1984	000500000.
1985	000600000.
OVERALL (IN DOLLARS)	
TOTAL	002000000.

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
HISTORY OF TUITION CHARGES AT VPI&SU

	1978	1979	1980	1981	1982	1983
RES_UG	00.00	00.00	00.00	00.00	35.70	46.00
NRES_UG	00.00	00.00	00.00	00.00	77.20	98.50
RES_GR	00.00	00.00	00.00	00.00	44.89	58.67
NRES_GR	00.00	00.00	00.00	00.00	49.89	64.67
RES_GR_OFC	00.00	00.00	43.00	47.00	54.00	69.00
NRES_GR_OFC	00.00	00.00	43.00	47.00	59.00	75.00

	1984	1985	1986	1987	1988	1989
RES_UG	00.00	00.00	00.00	00.00	00.00	00.00
NRES_UG	00.00	00.00	00.00	00.00	00.00	00.00
RES_GR	00.00	00.00	00.00	00.00	00.00	00.00
NRES_GR	00.00	00.00	00.00	00.00	00.00	00.00
RES_GR_OFC	00.00	00.00	00.00	00.00	00.00	00.00
NRES_GR_OFC	00.00	00.00	00.00	00.00	00.00	00.00

Appendix F

R^2 VALUES, BY STUDENT CATEGORY, FOR TUITION CHARGES VERSUS 0, 1, AND 2-YEAR LAG IN THE CPI

R^2 VALUES, BY STUDENT CATEGORY, FOR TUITION CHARGES VERSUS 0, 1, AND 2-YEAR LAG IN THE CPI

		Percentiles									
Student Category	Lag	99	95	90	75	50	25	10	5	1	Mean
Non-Resident Graduate	0	.8783	.8794	.9169	.8685	.9137	.8667	.7753	.8430	.8965	.9053
	1	.9586	.9551	.9709	.9618	.9808	.9484	.8682	.9311	.9068	.9785
	2	.9797	.9414	.9615	.9925	.9989	.9857	.9499	.9740	.8496	.9941
Non-Resident Undergraduate	0	.8791	.9333	.8875	.9144	.8903	.9006	.8434	.8844	.6570	.8982
	1	.9589	.9869	.9638	.9835	.9654	.9598	.9325	.9234	.6084	.9736
	2	.9789	.9945	.9876	.9882	.9970	.9645	.9863	.9428	.6160	.9965
Resident Undergraduate	0	.9076	.8743	.9165	.8870	.8937	.8777	.8639	.8371	.8427	.8893
	1	.9780	.9571	.9801	.9710	.9715	.9621	.9501	.9050	.7347	.9711
	2	.9995	.9763	.9828	.9971	.9948	.9857	.9801	.9562	.6323	.9955
Resident Undergraduate incl. Room and Board	0	.9380	.8935	.9066	.9403	.9229	.8984	.9532	.9805	.4058	.9345
	1	.9929	.9700	.9796	.9953	.9846	.9705	.9888	.9899	.2817	.9893
	2	.9934	.9850	.9946	.9870	.9880	.9829	.9811	.9508	.1571	.9868
Non-Resident Undergraduate incl. Room and Board	0	.9167	.9276	.8975	.9387	.8852	.8718	.9199	.9685	.7142	.9233
	1	.9837	.9851	.9720	.9921	.9603	.9504	.9810	.9668	.5746	.9849
	2	.9938	.9946	.9913	.9880	.9964	.9786	.9957	.9397	.4430	.9923
Resident Graduate	0	.9058	.8734	.8901	.9055	.8863	.8688	.9061	.6947	.9437	.8859
	1	.9776	.9569	.9666	.9752	.9678	.9453	.9617	.8134	.9435	.9670
	2	.9996	.9773	.9987	.9982	.9914	.9857	.9888	.9028	.9216	.9973

Appendix G

DESCRIPTION OF THE GOALS INCLUDED IN THE DSS

Goals That are Included in VPI & SU's
DSS for Tuition and Fee Policy Analysis

A. Aggregate Goals

1. State-level tuition revenue requirement in each year of the planning horizon.

2. Supplemental funds revenue requirement. Used to generate additional revenue above the state requirement.

   a. in each year
   b. total over the entire planning horizon

B. Individual Goals

1. Tuition charges as a percentage of average total cost per student. There is a separate goal for each student category in each year. For example, it may be desirable to set tuition charges such that a resident undergraduate pays no more than 30% of the average total cost, a non-resident undergraduate pays at least 50% of the average cost, etc. Another way of stating this is that a resident undergraduate should pay no more than 30% of his or her education (the state pays the other 70%), a non-resident undergraduate is expected to pay at least half of the cost of education (the state pays the other half), etc.

2. Percentage change in tuition charges over time. There is a separate goal for each student category in each year. This goal is used to control the rate of change in tuition charges separately for each student category. For example, if the overall revenue requirement is increasing faster than the CPI it may be possible to limit the increase

in resident student charges to the change in the CPI and let the non-resident students absorb the difference.

3. University position in relation to peer institutions. There is a separate goal for each student category in each year. In order to remain competitive in different student markets, the university often attempts to maintain a certain tuition position relative to similar schools within its peer group. For example, in setting tuition charges for in-state students, the university may be concerned with the average total charge for full-time students at other institutions within the state. For out-of-state undergraduates, the university may compare its charges to other Land-Grant colleges and universities, charges at neighboring state universities, etc. If most of the off-campus students are part-time, then it would be more appropriate to consider charges for a single course, rather than full-time charges, at institutions near the off-campus location(s).

C. Comparison Goals

1. Relative relationships between student categories. Tuition charges may be established by considering the ratio between any two student categories. For example, the university may set its tuition charge structure so that non-resident undergraduates pay at least twice the in-state rate, resident graduate students pay at least 25% more than resident undergraduates, etc. There is a separate goal for each year in the planning horizon.

2. Single-category absolute goal. It is sometimes desirable to directly specify a particular student category's charge, its lower limit, or its

upper limit. For example, the resident undergraduate charge may be limited to no more than $60 per credit-hour, the non-resident undergraduate charge to at least $100 per credit-hour, etc. There is a separate goal for each year in the planning horizon.

3. Absolute relationships between student categories. This goal is used to specify dollar per credit-hour differences between pairs of student categories. For example, the tuition charge structure may be set with the intent of maintaining at least a $10 per credit hour differential between on-campus and off-campus programs, maintaining the difference between resident and non-resident graduate students at between $10 and $15 per credit hour, etc. There is a separate goal for each year in the planning horizon.

4. Four-category absolute goal. This goal, while not especially intuitive, is used to model a situation where four student categories are simultaneously involved in a goal formulation. For example, this goal type is used to maintain the same difference in tuition charges between resident and non-resident graduate students in both the on-campus and off-campus programs (four student categories). There is a separate goal for each year in the planning horizon.

Appendix H

HARDCOPY VERSION OF THE COMPUTER SESSION USED TO DEMONSTRATE THE DSS

```
                ----  SYSTEM DEMONSTRATION  ----          5/7/84
              EXAMPLE SESSION WHERE NOTHING IS CHANGED

R;
dss

 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
 $                                                                      $
 $      VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY             $
 $ DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS          $
 $                                                                      $
 $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$


  .. IF YOU WANT TO GO DIRECTLY TO THE PRIORITY STRUCTURE AND NOT CHANGE
     ANY OF THE MODEL'S GOAL TARGETS OR BASIC DATA, >>> ENTER PRIORITY

  .. IF YOU WANT TO GO DIRECTLY TO THE GOAL TARGET VALUES AND NOT CHANGE
     ANY OF THE MODEL'S BASIC DATA, >>> ENTER TARGET

  .. IF YOU WANT TO STEP THROUGH THE PROCESS AND HAVE THE OPPORTUNITY TO
     VIEW OR CHANGE THE MODEL'S BASIC DATA, >>> ENTER STEP

s


                ****************************************
                *                                      *
 ************** DEFINITION OF THE  PLANNING HORIZON   ********************
                *                                      *
                ****************************************

.. THE NUMBER OF YEARS IN THE PLANNING HORIZON IS 3 (NYEAR)
..... (MAXIMUM NUMBER OF YEARS POSSIBLE IS 6 )

.. THE PLANNING HORIZON STARTS IN YEAR 1983 (SYEAR)

.. THE LAST YEAR OF THE PLANNING HORIZON MUST BE 1993 OR PRIOR

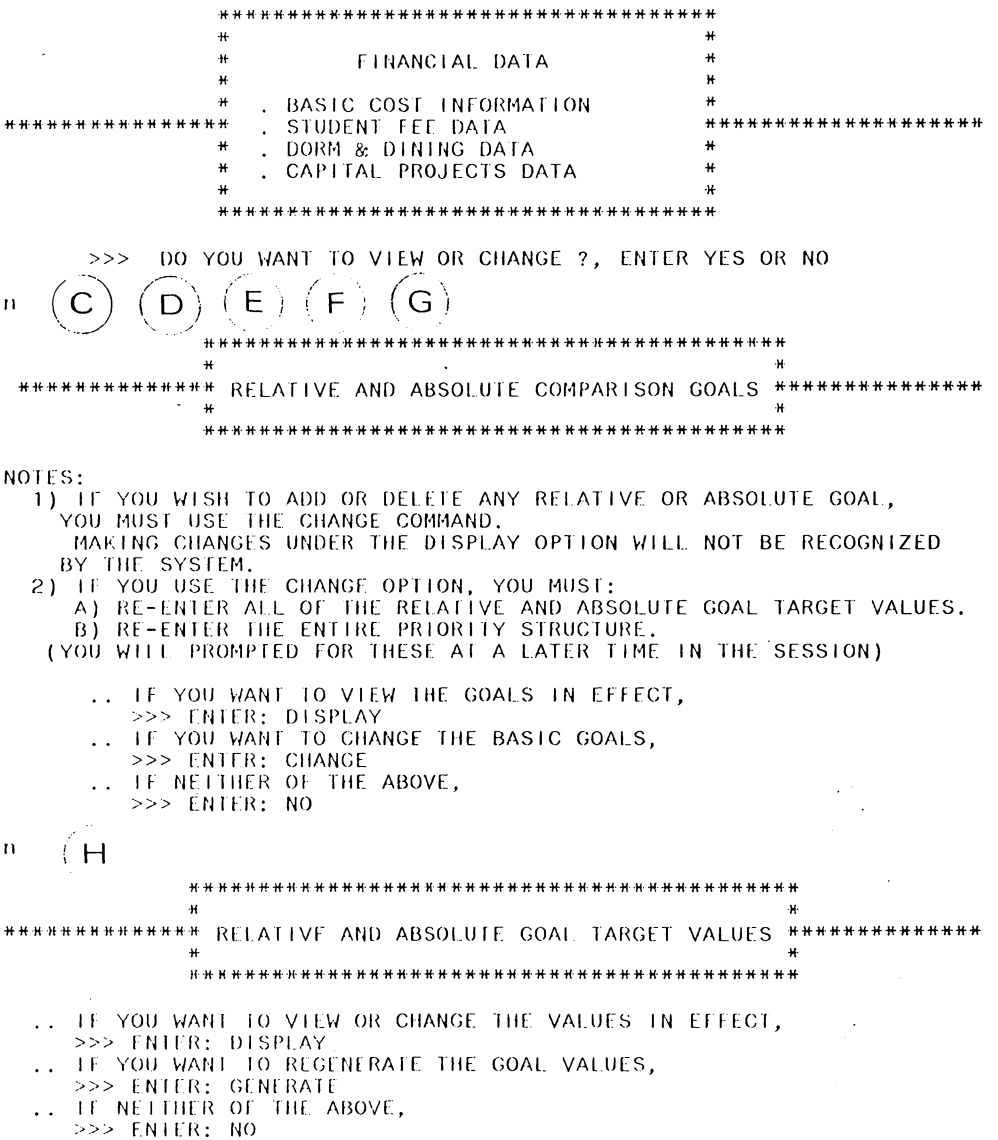
>>> DO YOU WANT TO CHANGE NYEAR OR SYEAR?, ENTER YES OR NO

n

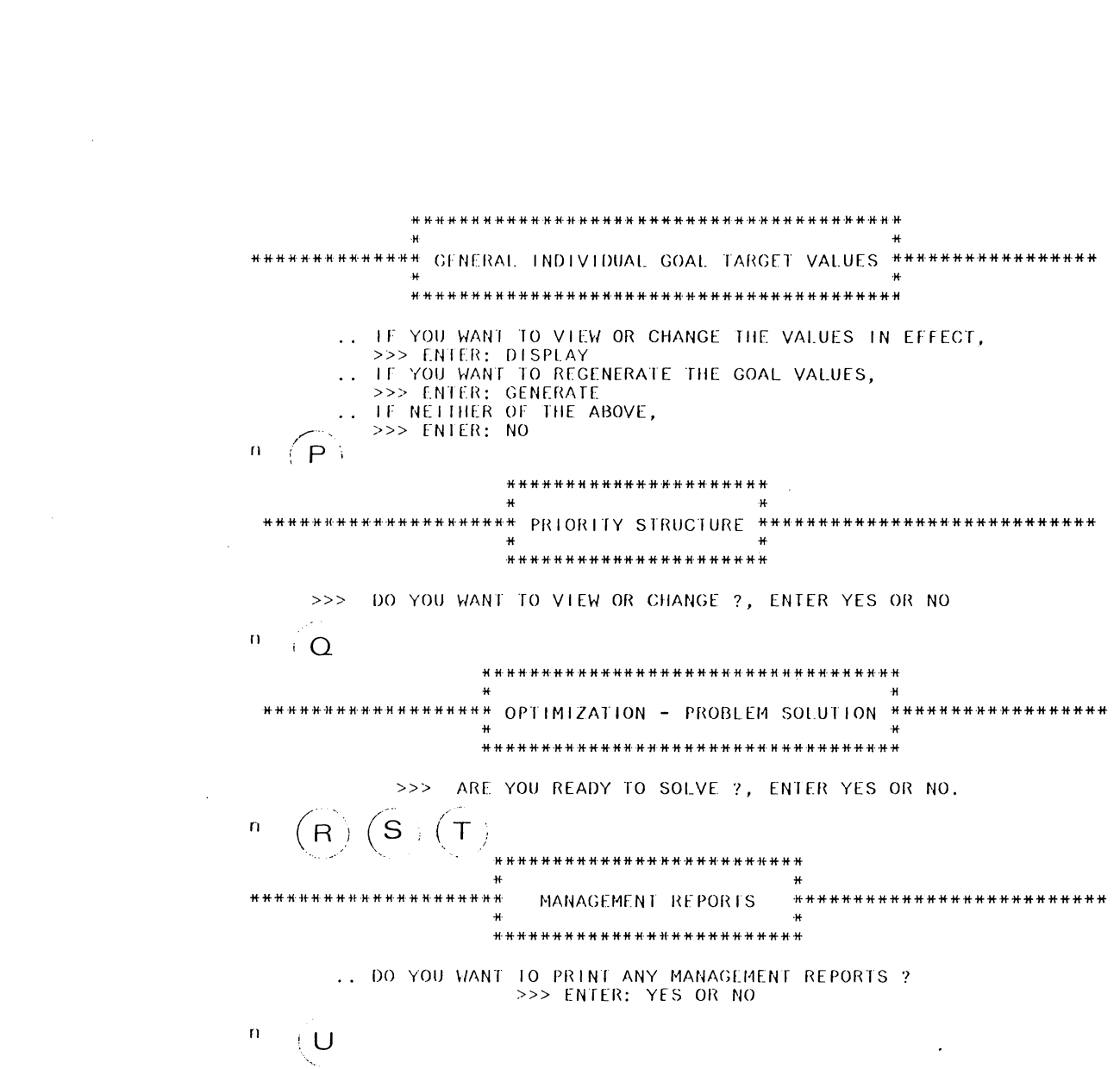
                              **********************
                              *                    *
 *****************************  ENROLLMENT DATA    ****************************
                              *                    *
                              **********************

    >>> DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO

n   (A)    (B
```


FINANCIAL DATA
. BASIC COST INFORMATION
. STUDENT FEE DATA
. DORM & DINING DATA
. CAPITAL PROJECTS DATA
>>> DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO
n
C
D
E
F
G
RELATIVE AND ABSOLUTE COMPARISON GOALS
NOTES:
1) IF YOU WISH TO ADD OR DELETE ANY RELATIVE OR ABSOLUTE GOAL,
YOU MUST USE THE CHANGE COMMAND.
MAKING CHANGES UNDER THE DISPLAY OPTION WILL NOT BE RECOGNIZED
BY THE SYSTEM.
2) IF YOU USE THE CHANGE OPTION, YOU MUST:
A) RE-ENTER ALL OF THE RELATIVE AND ABSOLUTE GOAL TARGET VALUES.
B) RE-ENTER THE ENTIRE PRIORITY STRUCTURE.
(YOU WILL PROMPTED FOR THESE AT A LATER TIME IN THE SESSION)
.. IF YOU WANT TO VIEW THE GOALS IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO CHANGE THE BASIC GOALS,
>>> ENTER: CHANGE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
n
H
RELATIVE AND ABSOLUTE GOAL TARGET VALUES
.. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO REGENERATE THE GOAL VALUES,
>>> ENTER: GENERATE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
n


********** SUPPLEMENTAL REVENUE REQUIREMENT GOAL TARGET VALUES **********
.. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO REGENERATE THE GOAL VALUES,
>>> ENTER: GENERATE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
n
J
********** PERCENTAGE OF AVERAGE TOTAL COST GOAL TARGET VALUES **********
.. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO REGENERATE THE GOAL VALUES,
>>> ENTER: GENERATE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
n
K
********** CHANGE-IN-TUITION-CHARGES-OVER-TIME GOAL TARGET VALUES **********
.. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO REGENERATE THE GOAL VALUES,
>>> ENTER: GENERATE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
n
L
M
********** PEER INSTITUTION GOALS **********
>>> DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO
n
N
O


GENERAL INDIVIDUAL GOAL TARGET VALUES
.. IF YOU WANT TO VIEW OR CHANGE THE VALUES IN EFFECT,
>>> ENTER: DISPLAY
.. IF YOU WANT TO REGENERATE THE GOAL VALUES,
>>> ENTER: GENERATE
.. IF NEITHER OF THE ABOVE,
>>> ENTER: NO
P
PRIORITY STRUCTURE
>>> DO YOU WANT TO VIEW OR CHANGE ?, ENTER YES OR NO
Q
OPTIMIZATION - PROBLEM SOLUTION
>>> ARE YOU READY TO SOLVE ?, ENTER YES OR NO.
R
S
T
MANAGEMENT REPORTS
.. DO YOU WANT TO PRINT ANY MANAGEMENT REPORTS ?
>>> ENTER: YES OR NO
U

```
                    ****************************
                    *                          *
 ********************  ?? ANOTHER ANALYSIS ??  ************************
                    *                          *
                    ****************************

    >>>  DO YOU WANT TO CONSIDER AN ALTERNATIVE OR A VARIATION
                ON THE PROBLEM ?,  ENTER YES OR NO.

n


$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$                                                                              $
$                END OF THE SESSION WITH THE VPI&SU                            $
$    DECISON SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS                $
$                                                                              $
$                             GOOD-BYE                                         $
$                                                                              $
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

R;
```

```
*         VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*     DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*            BASELINE FILE FOR "ACTUAL" ENROLLMENT VALUES
*    OTR DPB FORM 2-B      (AY 82-83 THRU 88-89)       DATED:  9/29/82
* INSTRUCTIONS:
*      1) DIRECTLY ENTER THE ENROLLMENT VALUES THAT YOU WANT TO USE
*         IN THE APPROPRIATE YEARS,
* OR   2) USE THE 'PREVIOUS MODIFIER' TO INCREASE OR DECREASE THE
*         ENROLLMENT BY A CONSTANT RATE, I.E. TO LET RESIDENT UNDERGRAD
*         ENROLLMENT INCREASE BY 1.5 % PER YEAR AFTER 1984, CHANGE:
*                              1978    1979    1980    1981    1982    1983
*    ACADEMIC YEARS            1984    1985    1986    1987    1988    1989
*ACTUAL REG FT UG RES  =    13637, 13637, 13637, 13637, 13637, 13617,'
*                           13502, 13447, 13453, 13457, 13548
*  TO:
*ACTUAL REG FT UG RES  =    13637, 13637, 13637, 13637, 13637, 13617,'
*                           13502, PREVIOUS * 1.015
***   REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
******   FULL-TIME (FT)
*                           1978    1979    1980    1981    1982    1983
*  ACADEMIC YEARS           1984    1985    1986    1987    1988    1989
*                           1990    1991    1992    1993
ACTUAL REG FT UG RES  =    13637, 13637, 13637, 13637, 13637, 13617,'
                           13502, 13447, 13453, 13457, 13548

ACTUAL REG FT UG NRES =     3975,   3975,   3975,   3975,   3975,   3993,'
                            3953,   3929,   3930,   3928,   3927

ACTUAL REG FT GR RES  =      805,    805,    805,    805,    805,    864,'
                             945,    982,   1027,   1036,   1042

ACTUAL REG FT GR NRES =     1602,   1602,   1602,   1602,   1602,   1713,'
                            1826,   1906,   1968,   1996,   1993

***   REGULAR SESSION (REG) - FALL QUARTER HEADCOUNT
******     PART-TIME (PT)
*                           1978    1979    1980    1981    1982    1983
* ACADEMIC YEARS            1984    1985    1986    1987    1988    1989
*                           1990    1991    1992    1993
ACTUAL REG PT UG RES  =      506,    506,    506,    506,    506,    527,'
                             524,    519,    519,    519,    519

ACTUAL REG PT UG NRES =       72,     72,     72,     72,     72,     73,'
                              73,     73,     73,     73,     73

ACTUAL REG PT GR RES  =      432,    432,    432,    432,    432,    462,'
                             489,    511,    532,    540,    538

ACTUAL REG PT GR NRES =      267,    267,    267,    267,    267,    279,'
                             294,    305,    319,    322,    321

***  SUMMER SESSION (SUM) - HEADCOUNT
```

(A)

```
******   FULL-TIME (FT)
*                          1978   1979   1980   1981   1982   1983
*    ACADEMIC YEARS        1984   1985   1986   1987   1988   1989
*                          1990   1991   1992   1993
ACTUAL SUM FT UG RES  =    1983,  1983,  1983,  1983,  1983,  1997,'
                           1977,  1948,  1948,  1948,  1948

ACTUAL SUM FT UG NRES =     429,   429,   429,   429,   429,   432,'
                            427,   402,   402,   402,   402

ACTUAL SUM FT GR RES  =     413,   413,   413,   413,   413,   427,'
                            447,   462,   462,   462,   462

ACTUAL SUM FT GR NRES =     594,   594,   594,   594,   594,   614,'
                            641,   669,   669,   669,   669

***  SUMMER SESSION (SUM) - HEADCOUNT
******   PART-TIME (PT)
*                          1978   1979   1980   1981   1982   1983
*    ACADEMIC YEARS        1984   1985   1986   1987   1988   1989
*                          1990   1991   1992   1993
ACTUAL SUM PT UG RES  =    2445,  2445,  2445,  2445,  2445,  2473,'
                           2451,  2421,  2421,  2421,  2421

ACTUAL SUM PT UG NRES =     565,   565,   565,   565,   565,   572,'
                            566,   559,   559,   559,   559

ACTUAL SUM PT GR RES  =     489,   489,   489,   489,   489,   507,'
                            530,   553,   553,   553,   553

ACTUAL SUM PT GR NRES =     377,   377,   377,   377,   377,   389,'
                            407,   425,   425,   425,   425

*
*
***  OFF-CAMPUS (OFC) - FALL QUARTER HEADCOUNT
*
******   FULL-TIME (FT)
*
*                          1978   1979   1980   1981   1982   1983
*    ACADEMIC YEARS        1984   1985   1986   1987   1988   1989
*                          1990   1991   1992   1993
*
ACTUAL OFC FT GR RES  =      61,    61,    61,    61,    61,    62,'
                             63,    65,    65,    65,    65

ACTUAL OFC FT GR NRES =      14,    14,    14,    14,    14,    14,'
                             14,    14,    14,    14,    14

*
***  OFF-CAMPUS (OFC) - FALL QUARTER HEADCOUNT
*
******   PART-TIME (PT)
*
*                          1978   1979   1980   1981   1982   1983
```

```
*    ACADEMIC YEARS          1984    1985    1986    1987    1988    1989
*                            1990    1991    1992    1993
*
ACTUAL OFC PT GR RES  =      1333,   1333,   1333,   1333,   1333,   1342,'
                             1355,   1375,   1375,   1375,   1375

ACTUAL OFC PT GR NRES =       177,    177,    177,    177,    177,    179,'
                              181,    184,    184,    184,    184
```

```
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*       DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*              MISCELLANEOUS ENROLLMENT FACTOR DATA
*
*
** NUMBER OF CREDIT HOURS TO BE CONSIDERED A FULL-TIME STUDENT
*
CREDIT HOUR FT UNDERGRAD = 10
CREDIT HOUR FT GRADUATE  =  9
*
** AVERAGE NUMBER OF CREDIT HOURS FOR PART-TIME STUDENTS, PER QUARTER
*   REGULAR SESSION
CREDIT HOUR REG PT UG RES  = 5.5
CREDIT HOUR REG PT UG NRES = 5.5
CREDIT HOUR REG PT GR RES  = 4.1
CREDIT HOUR REG PT GR NRES = 4.1
*   SUMMER SESSION
CREDIT HOUR SUM PT UG RES  = 6
CREDIT HOUR SUM PT UG NRES = 6
CREDIT HOUR SUM PT GR RES  = 5.4
CREDIT HOUR SUM PT GR NRES = 5.4
*   OFF-CAMPUS (OFC)
CREDIT HOUR OFC PT UG RES  = 0
CREDIT HOUR OFC PT UG NRES = 0
CREDIT HOUR OFC PT GR RES  = 3
CREDIT HOUR OFC PT GR NRES = 3
*
* AVERAGE CREDIT HOURS PER STUDENT PER QUARTER (FULL-TIME AND
*     PART-TIME COMBINED, BASED ON FALL-QUARTER ENROLLMENTS)
*
**  REGULAR SESSION
AVERAGE CREDIT HOUR REG UG = 15*1.006
AVERAGE CREDIT HOUR REG GR = 12*0.765
***    SUMMER SESSION
AVERAGE CREDIT HOUR SUM UG = 15*0.238
AVERAGE CREDIT HOUR SUM GR = 12*0.238
**      OFF-CAMPUS (OFC)
AVERAGE CREDIT HOUR OFC UG = 0
AVERAGE CREDIT HOUR OFC GR = 12*.38
** RETENTION FACTORS = FALL(1.00) + WINTER + SPRING
*
RETENTION FACTOR!-REG SESSION = 1.000 + .966 + .938
RETENTION FACTOR!-OFC = 1.000 + .873 + .806 + .565
*
*
** FRACTION OF STUDENTS RECEIVING TUITION WAIVERS
*
FEE WAIVER REG FT GR NRES = 0.75
FEE WAIVER REG PT GR NRES = 0.75
FEE WAIVER SUM FT GR NRES = 0.75
FEE WAIVER SUM PT GR NRES = 0.75
FEE WAIVER OFC FT GR NRES = 0.75
```

B

```
FEE WAIVER OFC PT GR NRES = 0.75
*
*
** GRADUATE STUDENT EMPLOYMENT RATES
*     PROPORTION OF NON-RESIDENT GRADUATE STUDENTS THAT ARE SIGNIFICANTLY
*      EMPLOYED IN THE INSTRUCTIONAL PROGRAM.
*      (USED IN THE TUITION AND FEE RECOVERY RATE CALCULATION)
*
*                              1978       1979       1980       1981
*                              1982       1983       1984       1985
*                              1986       1987       1988       1989
*                              1990       1991       1992       1993
NR GR EMPLOYMENT RATE =       .700,      .650,      .600,      .600,
                              .570
*
* MAXIMUM LEVEL OF GRADUATE STUDENT EMPLOYMENT RECOGNIZED BY THE
*    STATE'S TUITION AND FEE RECOVERY RATE FORMULA
MAX NR GR EMPLOYMENT RATE = 0.500
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
HISTORY OF TUITION CHARGES AT VPI&SU

	1978	1979	1980	1981	1982	1983
RES_UG	00.00	00.00	00.00	00.00	35.70	46.00
NRES_UG	00.00	00.00	00.00	00.00	77.20	98.50
RES_GR	00.00	00.00	00.00	00.00	44.89	58.67
NRES_GR	00.00	00.00	00.00	00.00	49.89	64.67
RES_GR_OFC	00.00	00.00	43.00	47.00	54.00	69.00
NRES_GR_OFC	00.00	00.00	43.00	47.00	59.00	75.00

	1984	1985	1986	1987	1988	1989
RES_UG	00.00	00.00	00.00	00.00	00.00	00.00
NRES_UG	00.00	00.00	00.00	00.00	00.00	00.00
RES_GR	00.00	00.00	00.00	00.00	00.00	00.00
NRES_GR	00.00	00.00	00.00	00.00	00.00	00.00
RES_GR_OFC	00.00	00.00	00.00	00.00	00.00	00.00
NRES_GR_OFC	00.00	00.00	00.00	00.00	00.00	00.00

```
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*       DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*          MISCELLANEOUS COST DATA, HISTORICAL AND PROJECTIONS
*
* ADJ INSTRUCT E&G = ADJUSTED INSTRUCTIONAL E&G; IT IS USED IN:
*    1) STATE REVENUE REQUIREMENT GOALS
*    2) PERCENTAGE OF AVERAGE COST GOALS
*
*                              1978         1979         1980         1981
*                              1982         1983         1984         1985
*                              1986         1987         1988         1989
*                              1990         1991         1992         1993
ADJ INSTRUCT E&G  = 000000000, 000000000, 000000000, 000000000,'
                    095931818, 104515143, 119142901, 127722440,'
                    136877443, 146292266, 156336674, 167258919

*
*                                      1978     1979     1980     1981
*                                      1982     1983     1984     1985
*                                      1986     1987     1988     1989
*                                      1990     1991     1992     1993
CONSUMER PRICE INDEX =                199.3,   223.4,   251.7,   279.3,'
                                      293.3,   301.8,  PREVIOUS*1.05


*
*                                      1978     1979     1980     1981
*                                      1982     1983     1984     1985
*                                      1986     1987     1988     1989
*                                      1990     1991     1992     1993
TUITION AND FEE RECOVERY RATE  =     0.2640, 0.2800, 0.2870, 0.3000,'
                                     0.3250, 0.3500, 0.0000, 0.0000,'
                                     0.0000, 0.0000, 0.0000, 0.0000,'
                                     0.0000, 0.0000, 0.0000, 0.0000
*
* ANNUAL COST FOR A FULL-TIME STUDENT, FROM TEP. (USED IN THE TUITION
*    AND FEE RECOVERY RATE CALCULATION)
*
*                                      1978     1979     1980     1981
*                                      1982     1983     1984     1985
*                                      1986     1987     1988     1989
*                                      1990     1991     1992     1993
UNDERGRADUATE TEP COST PER STUDENT = 00000., 00000., 01586.

*
*                                      1978     1979     1980     1981
*                                      1982     1983     1984     1985
*                                      1986     1987     1988     1989
*                                      1990     1991     1992     1993
GRADUATE TEP COST PER STUDENT =      00000., 00000., 06974.

*
```

```
* POLICY RATES FOR USE IN THE TUITION AND FEE RECOVERY RATE CALCULATIONS
*   DET: PERCENTAGE OF THE FULL COST OF INSTRUCTION THAT THE STATE
*     EXPECTS EACH STUDENT CATEGORY TO PAY.
*
*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
RESIDENT UNDERGRAD POLICY RATE =     000.0,  000.0,  000.0,  000.0,'
                                     000.0,  000.0,  025.0,  025.0

*
NONRESIDENT UNDERGRAD POLICY RATE =  000.0,  000.0,. 000.0,  000.0,'
                                     000.0,  000.0,  075.0,  075.0

*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
RESIDENT GRADUATE  POLICY RATE =     000.0,  000.0,  000.0,  000.0,'
                                     000.0,  000.0,  025.0,  025.0

*
NONRESIDENT GRADUATE  POLICY RATE = 000.0,  000.0,  000.0,  000.0,'
                                    000.0,  000.0,  075.0,  075.0

*
* MAXIMUM ALLOWABLE YEAR-TO-YEAR PERCENTAGE INCREASE AND DECREASE IN
*    THE TUITION AND FEE RECOVERY RATE
*
MAXIMUM RECOVERY RATE INCREASE    = 100.00, 100.00, 100.00, 100.00,'
                                    100.00, 100.00, 002.50, 002.50

*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
MAXIMUM RECOVERY RATE DECREASE    = 100.00, 100.00, 100.00, 100.00,'
                                    100.00, 100.00, 001.25, 001.25

*
MISC FEES = 350000
```

```
*
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*       DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                        STUDENT FEE DATA
*
*  NOTES:  1) IF A CERTAIN FEE IS INPUT AS 0, THEN FUTURE FEE CHARGES
*             WILL BE BASED ON THE OPERATING EXPENSES.
*          2) IF AN OPERATING EXPENSE IS INPUT AS 0, THEN FUTURE OPERATING
*             EXPENSES WILL BE PROJECTED USING THE APPROPRIATE ESCALATOR.
*          3) IF AN ESCALATOR IS INPUT AS A NEGATIVE VALUE, THEN THE
*             CPI WILL BE USED AS THE INDEX TO PROJECT FUTURE VALUES.
*
*
*
*
*
*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
QUARTERLY ATHLETIC FEE             = 000.00, 000.00, 018.00, 020.00,'
                                     022.00, 024.00, 026.00, 028.00,'
                                     030.00, 032.00, 034.00, 036.00

   ATH OPERATING EXPENSES 000      = 00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.

   ATH ESCALATOR                   = 1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000, 1.0000

   ATH PROP PART TIME STUDENTS     = 0.1000, 0.1000, 0.1000, 0.1000,'
                                     0.1000, 0.1000, 0.1000, 0.1000,'
                                     0.1000, 0.1000, 0.1000, 0.1000
*
*                                    1978     1979     1980     1981
*                                    1982     1983     1984     1985
*                                    1986     1987     1988     1989
*                                    1990     1991     1992     1993
QUARTERLY BUS SYSTEM FEE           = 000.00, 000.00, 000.00, 000.00,'
                                     004.00, 004.00, 005.00, 000.00,'
                                     000.00, 000.00, 000.00, 000.00

   BUS OPERATING EXPENSES 000      = 00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.,'
                                     00000., 00000., 00000., 00000.

   BUS ESCALATOR                   = 1.0000, 1.0000, 1.0000, 1.0000,'
                                     1.0000, 1.0000, 1.0000,-1.0000,'
                                    -1.0000,-1.0000,-1.0000,-1.0000
```

```
    BUS PROP PART TIME STUDENTS  = 0.0000, 0.0000, 0.0000, 0.0000,'
                                   0.0000, 0.0000, 0.0000, 0.0000,'
                                   0.0000, 0.0000, 0.0000, 0.0000

*
*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
QUARTERLY STUDENT CENTER FEE     = 000.00, 000.00, 015.00, 018.00,'
                                   000.00, 000.00, 000.00, 000.00,'
                                   000.00, 000.00, 000.00, 000.00

    STC OPERATING EXPENSES 000   = 01140., 01200., 01275., 01350.,'
                                   00000., 00000., 00000., 00000.,'
                                   00000., 00000., 00000., 00000.

    STC ESCALATOR                = 1.0000, 1.0000, 1.0000, 1.0000,'
                                   1.1000, 1.0700, 1.0700, 1.0700,'
                                   1.0700, 1.0700, 1.0700, 1.0700

    STC PROP PART TIME STUDENTS  = 0.5000, 0.5000, 0.5000, 0.5000,'
                                   0.5000, 0.5000, 0.5000, 0.5000,'
                                   0.5000, 0.5000, 0.5000, 0.5000

*
*                                    1978    1979    1980    1981
*                                    1982    1983    1984    1985
*                                    1986    1987    1988    1989
*                                    1990    1991    1992    1993
QUARTERLY HEALTH SERVICE FEE     = 000.00, 000.00, 014.00, 020.00,'
                                   022.00, 023.00, 024.00, 026.00,'
                                   028.00, 030.00, 031.00, 031.00

    HLT OPERATING EXPENSES 000   = 00000., 00000., 00000., 00000.,'
                                   00000., 00000., 00000., 00000.,'
                                   00000., 00000., 00000., 00000.

    HLT ESCALATOR                = 1.0000, 1.0000, 1.0000, 1.0000,'
                                   1.0000, 1.0000, 1.0000, 1.0000,'
                                   1.0000, 1.0000, 1.0000, 1.0000

    HLT PROP PART TIME STUDENTS  = 0.0000, 0.0000, 0.0000, 0.0000,'
                                   0.0000, 0.0000, 0.0000, 0.0000,'
                                   0.0000, 0.0000, 0.0000, 0.0000
```

```
*          VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
*      DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
*                    DORM AND DINING DATA
*
* NOTES: 1) IF AN OPERATING EXPENSE IS INPUT AS 0, THEN FUTURE CHARGES
*           WILL BE PROJECTED USING THE APPROPRIATE ESCALATOR.
*        2) IF AN ESCALATOR IS INPUT AS A NEGATIVE VALUE, THEN THE
*           CPI WILL BE USED AS THE INDEX TO PROJECT FUTURE VALUES.
*        3) THE RATIOS ARE USED TO ALLOCATE THE OPERATING EXPENSES
*           TO QUARTERLY CHARGES FOR EACH DORMITORY OR DINING GROUP,
*           I.E. IN 1983 LOWER AND PRAIRIE QUADS WERE THE BASE (RATIO
*           SPECIFIED AS 1.000), THE UPPER QUAD CHARGE WAS 11.3% BELOW
*           BASE (RATIO WAS .887), NEW UNDERGRADUATE WAS 38.7% ABOVE
*           THE BASE (RATIO WAS 1.387), ETC.
*
*
*                                    1978      1979      1980      1981
*                                    1982      1983      1984      1985
*   ACADEMIC YEAR                    1986      1987      1988      1989
*                                    1990      1991      1992      1993
*
DORM OPERATING EXPENSES 000   =  0000.0,   0000.0,   0000.0,   0000.0,'
                                 3726.1,   4267.5,   0000.0,   0000.0

DORM CHARGES ESCALATOR        =     1.000,    1.000,    1.000,    1.000,'
                                    1.000,    1.000,    1.070,   -1.000

*
DINING OPERATING EXPENSES 000 =   000.0,   0000.0,   0000.0,   0000.0,'
                                 8589.5,   9240.1,   0000.0,   0000.0

DINING CHARGES ESCALATOR      =     1.000,    1.000,    1.000,    1.000,'
                                    1.000,    1.000,    1.070,   -1.000

*                                1978      1979      1980      1981      1982
*                                1983      1984      1985      1986      1987
*   ACADEMIC YEAR                1988      1989      1990      1991      1992
*                                1993
*
UPPER QUAD RATIO              =  0.000,    0.000,    0.000,    0.000,    0.907,'
                                 0.887

UPPER QUAD STUDENTS           =    1950,     1950,     1950,     1950,     1950,'
                                 1950
*
LOWER QUAD RATIO              =  0.000,    0.000,    0.000,    0.000,    1.000,'
                                 1.000

LOWER QUAD STUDENTS           =    4380,     4380,     4380,     4380,     4380,'
                                 4380
*                                1978      1979      1980      1981      1982
*                                1983      1984      1985      1986      1987
```

(F)

```
*    ACADEMIC YEAR                1988     1989     1990     1991     1992
*                                 1993
*
PRAIRIE QUAD RATIO           =   0.000,   0.000,   0.000,   0.000,   1.026,'
                                 1.000

PRAIRIE QUAD STUDENTS        =    1969,    1969,    1969,    1969,    1969,'
                                 1969
*
NEW UNDERGRAD RATIO          =   0.000,   0.000,   0.000,   0.000,   0.000,'
                                 1.387

NEW UNDERGRAD STUDENTS       =    0000,    0000,    0000,    0000,    0000,'
                                  342
*
*                                1978     1979     1980     1981     1982
*                                1983     1984     1985     1986     1987
*    ACADEMIC YEAR               1988     1989     1990     1991     1992
*                                1993
*
SPEC PURPOSE RATIO           =   0.000,   0.000,   0.000,   0.000,   0.000,'
                                 1.402

SPEC PURPOSE STUDENTS        =     108,     108,     108,     108,     108,'
                                  108
*
GRAD SINGLE OCC RATIO        =   0.000,   0.000,   0.000,   0.000,   0.000,'
                                 2.176

GRAD SINGLE OCC STUDENTS     =    0000,    0000,    0000,    0000,    0000,'
                                   36
*
*                                1978     1979     1980     1981     1982
*                                1983     1984     1985     1986     1987
*    ACADEMIC YEAR               1988     1989     1990     1991     1992
*                                1993
*
GRAD DOUBLE OCC RATIO        =   0.000,   0.000,   0.000,   0.000,   0.000,'
                                 1.377

GRAD DOUBLE OCC STUDENTS     =    0000,    0000,    0000,    0000,    0000,'
                                  120
*
UG 19 MEAL PLAN  RATIO       =   0.000,   0.000,   0.000,   0.000,   1.000,'
                                 1.000

*
GR 15 MEAL PLAN  RATIO       =   0.000,   0.000,   0.000,   0.000,   0.000,'
                                 1.150
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
CAPITAL PROJECTS DATA RELATED TO STUDENT FEES

ATHLETIC ASSOCIATION			
PROJECT IDENTIFIER	XXXXXX	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	00000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0000	0.0000	0.0000
PROJECT LIFE - YEARS	00.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1900	1900	1900
BUS SYSTEM			
PROJECT IDENTIFIER	XXXXXX	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	00000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0000	0.0000	0.0000
PROJECT LIFE - YEARS	00.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1900	1900	1900
STUDENT CENTER			
PROJECT IDENTIFIER	SQ ADD	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	12500.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0950	0.0000	0.0000
PROJECT LIFE - YEARS	20.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1984	1900	1900
HEALTH SERVICE			
PROJECT IDENTIFIER	INTERM	XXXXXX	XXXXXX
AMORTIZED AMOUNT (000)	01000.	00000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0925	0.0000	0.0000
PROJECT LIFE - YEARS	20.000	00.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1985	1900	1900
DORM AND DINING			
PROJECT IDENTIFIER	NEW UG	VARIOUS	XXXXXX
AMORTIZED AMOUNT (000)	02500.	04000.	00000.
UNAMORTIZED AMOUNT (000)	00000.	00000.	00000.
INTEREST RATE	0.0800	0.1000	0.0000
PROJECT LIFE - YEARS	20.000	15.000	00.000
PAYMENTS PER YEAR	01	01	01
YEAR OF PROJECT START	1983	1984	1900

(G)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ALL OF THE POSSIBLE COMPARISON GOALS

RELATIVE-COMPARISON GOALS

Y	2	1	NRES UG	/	RES UG	=	VALUE
Y	3	1	RES GR	/	RES UG	=	VALUE
N	4	1	NRES GR	/	RES UG	=	VALUE
N	5	1	RES GR OFC	/	RES UG	=	VALUE
N	6	1	NRES GR OFC	/	RES UG	=	VALUE
N	3	2	RES GR	/	NRES UG	=	VALUE
N	4	2	NRES GR	/	NRES UG	=	VALUE
N	5	2	RES GR OFC	/	NRES UG	=	VALUE
N	6	2	NRES GR OFC	/	NRES UG	=	VALUE
N	4	3	NRES GR	/	RES GR	=	VALUE
N	5	3	RES GR OFC	/	RES GR	=	VALUE
N	6	3	NRES GR OFC	/	RES GR	=	VALUE
N	5	4	RES GR OFC	/	NRES GR	=	VALUE
N	6	4	NRES GR OFC	/	NRES GR	=	VALUE
N	6	5	NRES GR OFC	/	RES GR OFC	=	VALUE

SINGLE-CATEGORY ABSOLUTE GOALS

N 1	1	RES UG		=	VALUE
N 1	2	NRES UG		=	VALUE
N 1	3	RES GR		=	VALUE
N 1	4	NRES GR		=	VALUE
N 1	5	RES GR OFC		=	VALUE
N 1	6	NRES GR OFC		=	VALUE

PAIRWISE-COMPARISON ABSOLUTE GOALS

N 2	2	1	NRES UG	-	RES UG	=	VALUE
N 2	3	1	RES GR	-	RES UG	=	VALUE
N 2	4	1	NRES GR	-	RES UG	=	VALUE
N 2	5	1	RES GR OFC	-	RES UG	=	VALUE
N 2	6	1	NRES GR OFC	-	RES UG	=	VALUE
N 2	3	2	RES GR	-	NRES UG	=	VALUE
N 2	4	2	NRES GR	-	NRES UG	=	VALUE
N 2	5	2	RES GR OFC	-	NRES UG	=	VALUE
Y 2	4	3	NRES GR OFC	-	NRES UG	=	VALUE
Y 2	4	3	NRES GR	-	RES GR	=	VALUE
Y 2	5	3	RES GR OFC	-	RES GR	=	VALUE
N 2	6	3	NRES GR OFC	-	RES GR	=	VALUE
N 2	5	4	RES GR OFC	-	NRES GR	=	VALUE
N 2	6	4	NRES GR OFC	-	NRES GR	=	VALUE
N 2	6	5	NRES GR OFC	-	RES GR OFC	=	VALUE

FOUR-CATEGORY ABSOLUTE COMPARISON GOALS:
USING THE NUMBERS THAT CORRESPOND TO THE STUDENT CATEGORIES LISTED BELOW,
CONSTRUCT FOUR-CATEGORY ABSOLUTE GOALS. FOR EXAMPLE, TO MAINTAIN THE
SAME DIFFERENTIAL CHARGE BETWEEN RESIDENT AND NON-RESIDENT GRADUATE
STUDENTS, BOTH ON-CAMPUS AND OFF-CAMPUS, ENTER +4 -3 -6 +5.
THIS CORRESPONDS TO (NRES GR - RES GR) - (NRES GR OFC - RES GR OFC)

(H)

```
Y 4    +4 -3 -6 +5
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
N 4    +0 -0 -0 +0
```

NUMBER	CATEGORY
1	RES_UG
2	NRES_UG
3	RES_GR
4	NRES_GR
5	RES_GR_OFC
6	NRES_GR_OFC

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
COMPARISON GOAL TARGET VALUES

RELATIVE COMPARISON GOALS

2	1	NRES_UG	/	RES_UG	= 2.00000	FOR YEAR 1983
					= 2.00000	FOR YEAR 1984
					= 2.00000	FOR YEAR 1985
3	1	RES_GR	/	RES_UG	= 1.20000	FOR YEAR 1983
					= 1.20000	FOR YEAR 1984
					= 1.20000	FOR YEAR 1985

SINGLE-CATEGORY ABSOLUTE GOALS

PAIRWISE-COMPARISON ABSOLUTE GOALS

4	3	NRES_GR	-	RES_GR	= 5.00000	FOR YEAR 1983
					= 8.00000	FOR YEAR 1984
					= 10.0000	FOR YEAR 1985
4	3	NRES_GR	-	RES_GR	= 10.0000	FOR YEAR 1983
					= 16.0000	FOR YEAR 1984
					= 20.0000	FOR YEAR 1985
5	3	RES_GR_OFC	-	RES_GR	= 10.0000	FOR YEAR 1983
					= 15.0000	FOR YEAR 1984
					= 20.0000	FOR YEAR 1985

FOUR-CATEGORY ABSOLUTE COMPARISON GOALS

4 -3 -6 5	NRES_GR	-	RES_GR	- NRES_GR_OFC + RES_GR_OFC		
				= 0000000	FOR YEAR 1983	
				= 0000000	FOR YEAR 1984	
				= 0000000	FOR YEAR 1985	

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUPPLEMENTAL REVENUE GOAL TARGET VALUES

BY YEAR (IN DOLLARS)

1983	000400000.
1984	000500000.
1985	000600000.

OVERALL (IN DOLLARS)

TOTAL	002000000.

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
GOAL TARGETS FOR PERCENTAGE OF AVG. TOTAL COST GOALS

	1983	1984	1985			
RES_UG	25.00	28.00	30.00	00.00	00.00	00.00
NRES_UG	50.00	60.00	65.00	00.00	00.00	00.00
RES_GR	30.00	33.00	35.00	00.00	00.00	00.00
NRES_GR	35.00	48.00	40.00	00.00	00.00	00.00
RES_GR_OFC	40.00	40.00	40.00	00.00	00.00	00.00
NRES_GR_OFC	45.00	45.00	45.00	00.00	00.00	00.00

(K)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SUMMARY OF KEY ECONOMIC INDICATORES THAT ARE HELPFUL IN ESTABLISHING
THE PERCENTAGE-CHANGE-OVER-TIME GOAL TARGET VALUES

	1983	1984	1985
CHANGE IN THE INSTITUTION'S OPERATING EXPENSES OVER THE PREVIOUS YEAR	8.95%	14.00%	7.20%
CHANGE IN THE STATE REVENUE REQUIREMENT OVER THE PREVIOUS YEAR	17.52%	20.23%	7.36%
CHANGE IN THE CONSUMER PRICE INDEX OVER THE PREVIOUS YEAR	2.90%	5.00%	5.00%

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
TARGETED PERCENTAGE CHANGE IN TUITION OVER PREVIOUS YEAR

	1983	1984	1985			
RES_UG	18.00	21.00	10.00	00.00	00.00	00.00
NRES_UG	20.00	25.00	10.00	00.00	00.00	00.00
RES_GR	18.00	23.00	10.00	00.00	00.00	00.00
NRES_GR	18.00	23.00	10.00	00.00	00.00	00.00
RES_GR_OFC	22.00	25.00	10.00	00.00	00.00	00.00
NRES_GR_OFC	22.00	25.00	10.00	00.00	00.00	00.00

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
INFORMATION REQUIRED TO SET THE PEER INSTITUTION GOALS

STUDENT CATEGORY	ANNUAL CREDIT HOURS	BASIS FOR GOAL TARGET		OTHER CHARGES / FEES				
		SOURCE	MEASURE	ATHL	BUS	STUD CNTR	HLTH SERV	ROOM & BRD
RES UG	30.	IG	MN	Y	Y	Y	Y	Y
NRES UG	30.	IG	25	Y	Y	Y	Y	Y
RES GR	27.	IG	MN	Y	Y	Y	Y	N
NRES GR	27.	IG	25	Y	Y	Y	Y	N
RES GR OFC	03.	DI	DI	N	N	N	N	N
NRES GR OFC	03.	DI	DI	N	N	N	N	N

DEFINITIONS OF "BASIS FOR GOAL TARGET" OPTIONS:
SOURCE: IG = LAND-GRANT COLLEGES & UNIVERSITIES
NC = NCHEMS IA ISTITUTIONS
VA = VIRGINIA 4-YEAR COLLEGES & UNIVERSITIES
DI = DIRECT INPUT
MEASURE: MN = ARITHMETIC MEAN
01 - 99 = PERCENTILE
DI = DIRECT INPUT
NOTE: IF DI, DIRECT INPUT, IS SPECIFIED FOR EITHER SOURCE OR MEASURE,
THEN IT MUST BE SPECIFIED FOR BOTH

(2)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ESTIMATED PEER INSTITUTION CHARGES BY STUDENT CATEGORY 05/23/84

	6	7	8
RES UG	3549.	3789.	3894.
NRES UG	4494.	4702.	4826.
RES GR	1440.	1524.	1574.
NRES GR	2356.	2463.	2530.
RES GR OFC	200.	225.	250.
NRES GR OFC	280.	310.	340.

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS
GOAL TARGETS FOR GENERAL INDIVIDUAL GOALS

	1983	1984	1985			
RES_UG	000.00	000.00	000.00	000.00	000.00	000.00
NRES_UG	000.00	000.00	000.00	000.00	000.00	000.00
RES_GR	000.00	000.00	000.00	000.00	000.00	000.00
NRES_GR	000.00	000.00	000.00	000.00	000.00	000.00
RES_GR_OFC	000.00	000.00	000.00	000.00	000.00	000.00
NRES_GR_OFC	000.00	000.00	000.00	000.00	000.00	000.00

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
GOAL PRIORITY STRUCTURE

IF THE GOAL TARGET IS THE MINIMUM ACCEPTABLE VALUE,
ENTER THE PRIORITY LEVEL IN THE 1ST COLUMN
IF THE GOAL TARGET IS THE MAXIMUM ACCEPTABLE VALUE,
ENTER THE PRIORITY LEVEL IN THE 2ND COLUMN

	PRIORITY LEVEL >=	PRIORITY LEVEL <=	SUBGOAL WEIGHTS	
STATE REVENUE REQUIREMENT	01	00	01.000	01.000
SUPPLEMENTAL REVENUE RQMT. - YEARLY	04	00	01.000	01.000
PERCENTAGE OF AVERAGE TOTAL COST GOALS				
RESIDENT UNDERGRADUATE	00	05	01.000	04.000
NON-RESIDENT UNDERGRADUATE	00	08	01.000	01.000
RESIDENT GRADUATE	00	05	01.000	02.000
NON-RESIDENT GRADUATE	00	05	01.000	01.000
RESIDENT GRADUATE OFF-CAMPUS	00	08	01.000	01.000
NON-RESIDENT GRADUATE OFF-CAMPU	00	08	01.000	01.000
CHANGE OVER TIME GOALS				
RESIDENT UNDERGRADUATE	00	03	01.000	01.000
NON-RESIDENT UNDERGRADUATE	00	03	01.000	01.000
RESIDENT GRADUATE	00	03	01.000	01.000
NON-RESIDENT GRADUATE	00	03	01.000	01.000
RESIDENT GRADUATE OFF-CAMPUS	00	03	01.000	01.000
NON-RESIDENT GRADUATE OFF-CAMPU	00	03	01.000	01.000
PEER INSTITUTIONS GOALS				
RESIDENT UNDERGRADUATE	00	09	01.000	01.000
NON-RESIDENT UNDERGRADUATE	00	06	01.000	01.000
RESIDENT GRADUATE	00	09	01.000	01.000
NON-RESIDENT GRADUATE	00	06	01.000	02.000
RESIDENT GRADUATE OFF-CAMPUS	00	09	01.000	01.000
NON-RESIDENT GRADUATE OFF-CAMPU	00	09	01.000	01.000
GENERAL INDIVIDUAL GOALS				
RESIDENT UNDERGRADUATE	00	00	01.000	01.000
NON-RESIDENT UNDERGRADUATE	00	00	01.000	01.000
RESIDENT GRADUATE	00	00	01.000	01.000
NON-RESIDENT GRADUATE	00	00	01.000	01.000
RESIDENT GRADUATE OFF-CAMPUS	00	00	01.000	01.000
NON-RESIDENT GRADUATE OFF-CAMPU	00	00	01.000	01.000
RELATIVE GOALS				
NRES UG VS. RES UG	07	00	01.000	01.000
RES GR VS. RES UG	07	00	01.000	01.000

SINGLE-CATEGORY ABSOLUTE GOALS						
PAIRWISE ABSOLUTE GOALS						
NRES_GR	VS.	RES_GR	02	00	01.000	01.000
NRES_GR	VS.	RES_GR	00	02	01.000	01.000
RES_GR_OFC	VS.	RES_GR	00	02	01.000	01.000
FOUR-CATEGORY ABSOLUTE GOALS						
NRES_GR - RES_GR - NRES_GR_OFC + RES_GR_OFC			02	02	01.000	01.000
SUPPLEMENTAL REVENUE RQMT. - OVERALL			04	00	01.000	01.000

```
 LINDO (UC 1 APRIL 82)
:
:
FILE NUMBER:
?
NAME    TUITION PROBLEM DATA

OBJECTIVE ROW FOUND:     ROW10001
MAX OR MIN ?
?
ROWS=      98 VARS=     212 NO. INTEGER VARS=       0
NONZEROES=     402 CONSTRAINT NONZ=    344(    272 ARE +- 1) DENSITY=0.019
SMALLEST AND LARGEST ELEMENTS IN ABSOLUTE VALUE=     .455000         46763.0
NO. < :    0 NO. =:  97 NO. > :    0, OBJ=MIN, GUBS <=  24
:

MIN
SUBJECT TO
   2)      N001 -  P001 + 438.66309 X11 + 124.875 X21 + 76.453 X31
     + 13.93 X41 + 16.48399 X51 + 0.538 X61 =     36230.30078
   3)      N002 -  P002 + 434.94312 X12 + 123.627 X22 + 81.799 X32
     + 14.798 X42 + 16.65401 X52 + 0.543 X62 =     43559.00391
   4)      N003 -  P003 + 432.7959 X13 + 122.638 X23 + 85.216 X33
     + 15.441 X43 + 16.929 X53 + 0.55 X63 =     46762.99219
   5)      P001 +  N004 -  P004 =     400
   6)      P002 +  N005 -  P005 =     500
   7)      P003 +  N006 -  P006 =     600
   8)      N007 -  P007 + 0.615 X11 =     25
   9)      N008 -  P008 + 0.615 X21 =     50
  10)      N009 -  P009 + 0.554 X31 =     30
  11)      N010 -  P010 + 0.554 X41 =     35
  12)      N011 -  P011 + 0.554 X51 =     40
  13)      N012 -  P012 + 0.554 X61 =     45
  14)      N013 -  P013 + 0.54 X12 =     28
  15)      N014 -  P014 + 0.54 X22 =     60
  16)      N015 -  P015 + 0.486 X32 =     33
  17)      N016 -  P016 + 0.486 X42 =     48
  18)      N017 -  P017 + 0.486 X52 =     40
  19)      N018 -  P018 + 0.486 X62 =     45
  20)      N019 -  P019 + 0.505 X13 =     30
  21)      N020 -  P020 + 0.505 X23 =     65
  22)      N021 -  P021 + 0.455 X33 =     35
  23)      N022 -  P022 + 0.455 X43 =     40
  24)      N023 -  P023 + 0.455 X53 =     40
  25)      N024 -  P024 + 0.455 X63 =     45
  26)      N025 -  P025 +  X11 =     42.12601
  27)      N026 -  P026 +  X21 =     92.64
  28)      N027 -  P027 +  X31 =     52.97
  29)      N028 -  P028 +  X41 =     58.87
  30)      N029 -  P029 +  X51 =     65.88
  31)      N030 -  P030 +  X61 =     71.98
  32)      N031 -  P031 - 1.21 X11 +  X12 =     0
```

R

```
33)     N032 -  P032 - 1.25 X21 +  X22 =     0
34)     N033 -  P033 - 1.23 X31 +  X32 =     0
35)     N034 -  P034 - 1.23 X41 +  X42 =     0
36)     N035 -  P035 - 1.25 X51 +  X52 =     0
37)     N036 -  P036 - 1.25 X61 +  X62 =     0
38)     N037 -  P037 - 1.1 X12 +  X13 =     0
39)     N038 -  P038 - 1.1 X22 +  X23 =     0
40)     N039 -  P039 - 1.1 X32 +  X33 =     0
41)     N040 -  P040 - 1.1 X42 +  X43 =     0
42)     N041 -  P041 - 1.1 X52 +  X53 =     0
43)     N042 -  P042 - 1.1 X62 +  X63 =     0
44)     N043 -  P043 + 30 X11 =     1795
45)     N044 -  P044 + 30 X21 =     2740
46)     N045 -  P045 + 27 X31 =     1224
47)     N046 -  P046 + 27 X41 =     2140
48)     N047 -  P047 + 3 X51 =     200
49)     N048 -  P048 + 3 X61 =     280
50)     N049 -  P049 + 30 X12 =     1790
51)     N050 -  P050 + 30 X22 =     2703
52)     N051 -  P051 + 27 X32 =     1227
53)     N052 -  P052 + 27 X42 =     2166
54)     N053 -  P053 + 3 X52 =     225
55)     N054 -  P054 + 3 X62 =     310
56)     N055 -  P055 + 30 X13 =     1787
57)     N056 -  P056 + 30 X23 =     2719
58)     N057 -  P057 + 27 X33 =     1253
59)     N058 -  P058 + 27 X43 =     2209
60)     N059 -  P059 + 3 X53 =     250
61)     N060 -  P060 + 3 X63 =     340
62)     N061 -  P061 +  X11 =     0
63)     N062 -  P062 +  X21 =     0
64)     N063 -  P063 +  X31 =     0
65)     N064 -  P064 +  X41 =     0
66)     N065 -  P065 +  X51 =     0
67)     N066 -  P066 +  X61 =     0
68)     N067 -  P067 +  X12 =     0
69)     N068 -  P068 +  X22 =     0
70)     N069 -  P069 +  X32 =     0
71)     N070 -  P070 +  X42 =     0
72)     N071 -  P071 +  X52 =     0
73)     N072 -  P072 +  X62 =     0
74)     N073 -  P073 +  X13 =     0
75)     N074 -  P074 +  X23 =     0
76)     N075 -  P075 +  X33 =     0
77)     N076 -  P076 +  X43 =     0
78)     N077 -  P077 +  X53 =     0
79)     N078 -  P078 +  X63 =     0
80)     N079 -  P079 - 2 X11 +  X21 =     0
81)     N080 -  P080 - 2 X12 +  X22 =     0
82)     N081 -  P081 - 2 X13 +  X23 =     0
83)     N082 -  P082 - 1.2 X11 +  X31 =     0
84)     N083 -  P083 - 1.2 X12 +  X32 =     0
85)     N084 -  P084 - 1.2 X13 +  X33 =     0
86)     N085 -  P085 -  X31 +  X41 =     5
87)     N086 -  P086 -  X32 +  X42 =     8
```

```
  88)      N087 -   P087 -   X33 +   X43 =      10
  89)      N088 -   P088 -   X31 +   X41 =      10
  90)      N089 -   P089 -   X32 +   X42 =      16
  91)      N090 -   P090 -   X33 +   X43 =      20
  92)      N091 -   P091 -   X31 +   X51 =      10
  93)      N092 -   P092 -   X32 +   X52 =      15
  94)      N093 -   P093 -   X33 +   X53 =      20
  95)      N094 -   P094 -   X31 +   X41 +   X51 -   X61 =      0
  96)      N095 -   P095 -   X32 +   X42 +   X52 -   X62 =      0
  97)      N096 -   P096 -   X33 +   X43 +   X53 -   X63 =      0
  98)      P001 +   P002 +   P003 +   N097 -   P097 =      2000
END

:
```

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
RECOMMENDED TUITION CHARGES, IN DOLLARS PER CREDIT HOUR

	1982	1983	1984	1985
RES_UG	35.70	42.67	51.63	56.79
NRES_UG	77.20	92.64	115.80	119.32
RES_GR	44.89	52.97	64.41	69.65
NRES_GR	49.89	58.87	72.41	79.65
RES_GR_OTC	54.00	62.97	78.09	84.70
NRES_GR_OTC	59.00	68.87	86.09	94.70

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SUMMARY OF GOAL ACHIEVEMENTS

GOAL ACHVD ?	GOAL DEFINITION	PRIORITY LEVEL OF DEV VAR N	P	VALUE OF THE DEVIATION VARIABLES N	P	GOAL TARGET VALUE	GOAL LEVEL ATTAINED
STATE REVENUE REQUIREMENT							
YES	YEAR: 1983	1.	0.	0.0	0.0	36230.30	36230.30
YES	YEAR: 1984	1.	0.	0.0	900.00	43559.00	44459.00
YES	YEAR: 1985	1.	0.	0.0	1100.00	46762.99	47862.99
SUPPLEMENTAL REVENUE REQUIREMENT-YEARLY							
NO	YEAR: 1983	4.	0.	400.00	0.0	400.00	0.0
YES	YEAR: 1984	4.	0.	0.0	400.00	500.00	900.00
YES	YEAR: 1985	4.	0.	0.0	500.00	600.00	1100.00
PERCENTAGE OF AVERAGE TOTAL COST							
	YEAR: 1983						
NO	RES UG	0.	5.	0.0	1.24	25.00	26.24
NO	NRES UG	0.	8.	0.0	6.97	50.00	56.97
YES	RES GR	0.	5.	0.65	0.0	30.00	29.35
YES	NRES GR	0.	5.	2.39	0.0	35.00	32.61
YES	RES GR OEC	0.	8.	5.11	0.0	40.00	34.89
YES	NRES GR OEC	0.	8.	6.85	0.0	45.00	38.15
	YEAR: 1984						
YES	RES UG	0.	5.	0.12	0.0	28.00	27.88
NO	NRES UG	0.	8.	0.0	2.53	60.00	62.53
YES	RES GR	0.	5.	1.70	0.0	33.00	31.30
YES	NRES GR	0.	5.	12.81	0.0	48.00	35.19

YES	RES_GR_OFC	0.	8.	2.05	0.0	40.00	37.95
YES	NRES GR OFC	0.	8.	3.16	0.0	45.00	41.84
	YEAR: 1985						
YES	RES_UG	0.	5.	1.32	0.0	30.00	28.68
YES	NRES_UG	0.	8.	4.75	0.0	65.00	60.25
YES	RES_GR	0.	5.	3.31	0.0	35.00	31.69
YES	NRES_GR	0.	5.	3.76	0.0	40.00	36.24
YES	RES_GR_OFC	0.	8.	1.46	0.0	40.00	38.54
YES	NRES_GR_OFC	0.	8.	1.91	0.0	45.00	43.09
CHANGE IN TUITION CHARGES OVER TIME							
	YEAR: 1983						
NO	RES_UG	0.	3.	0.0	0.54	42.13	42.67
YES	NRES_UG	0.	3.	0.0	0.0	92.64	92.64
YES	RES_GR	0.	3.	0.0	0.0	52.97	52.97
YES	NRES_GR	0.	3.	0.0	0.0	58.87	58.87
YES	RES_GR_OFC	0.	3.	2.91	0.0	65.88	62.97
YES	NRES_GR_OFC	0.	3.	3.11	0.0	71.98	68.87
	YEAR: 1984						
YES	RES_UG	0.	3.	0.0	0.0	0.0	0.0
YES	NRES_UG	0.	3.	0.0	0.0	0.0	0.0
YES	RES_GR	0.	3.	0.74	0.0	0.0	-0.74
YES	NRES_GR	0.	3.	0.0	0.0	0.0	0.0
YES	RES_GR_OFC	0.	3.	0.63	0.0	0.0	-0.63
YES	NRES_GR_OFC	0.	3.	0.0	0.0	0.0	0.0
	YEAR: 1985						
YES	RES_UG	0.	3.	0.0	0.0	0.0	0.0
YES	NRES_UG	0.	3.	8.06	0.0	0.0	-8.06
YES	RES_GR	0.	3.	1.20	0.0	0.0	-1.20

YES	NRES_GR	0.	3.	0.0	0.0	0.0	0.0
YES	RES_GR_OFC	0.	3.	1.20	0.0	0.0	-1.20
YES	NRES_GR_OFC	0.	3.	0.0	0.0	0.0	0.0

RELATIONSHIP TO PEER INSTITUTIONS

YEAR: 1983

YES	RES_UG	0.	9.	514.95	0.0	1795.00	1280.05
NO	NRES_UG	0.	6.	0.0	39.20	2740.00	2779.20
NO	RES_GR	0.	9.	0.0	206.19	1224.00	1430.19
YES	NRES_GR	0.	6.	550.51	0.0	2140.00	1589.49
YES	RES_GR_OFC	0.	9.	11.09	0.0	200.00	188.91
YES	NRES_GR_OFC	0.	9.	73.39	0.0	280.00	206.61

YEAR: 1984

YES	RES_UG	0.	9.	241.14	0.0	1790.00	1548.86
NO	NRES_UG	0.	6.	0.0	771.00	2703.00	3474.00
NO	RES_GR	0.	9.	0.0	512.07	1227.00	1739.07
YES	NRES_GR	0.	6.	210.93	0.0	2166.00	1955.07
NO	RES_GR_OFC	0.	9.	0.0	9.26	225.00	234.26
YES	NRES_GR_OFC	0.	9.	51.74	0.0	310.00	258.26

YEAR: 1985

YES	RES_UG	0.	9.	83.25	0.0	1787.00	1703.75
NO	NRES_UG	0.	6.	0.0	860.46	2719.00	3579.46
NO	RES_GR	0.	9.	0.0	627.58	1253.00	1880.58
YES	NRES_GR	0.	6.	58.42	0.0	2209.00	2150.58
NO	RES_GR_OFC	0.	9.	0.0	4.09	250.00	254.09
YES	NRES_GR_OFC	0.	9.	55.91	0.0	340.00	284.09

GENERAL INDIVIDUAL GOALS

YEAR: 1983

-	RES_UG	0.	0.	0.0	42.67	0.0	42.67
-	NRES_UG	0.	0.	0.0	92.64	0.0	92.64
-	RES_GR	0.	0.	0.0	52.97	0.0	52.97
-	NRES_GR	0.	0.	0.0	58.87	0.0	58.87
-	RES_GR_OFC	0.	0.	0.0	62.97	0.0	62.97
-	NRES_GR_OFC	0.	0.	0.0	68.87	0.0	68.87
	YEAR: 1984						
-	RES_UG	0.	0.	0.0	51.63	0.0	51.63
-	NRES_UG	0.	0.	0.0	115.80	0.0	115.80
-	RES_GR	0.	0.	0.0	64.41	0.0	64.41
-	NRES_GR	0.	0.	0.0	72.41	0.0	72.41
-	RES_GR_OFC	0.	0.	0.0	78.09	0.0	78.09
-	NRES_GR_OFC	0.	0.	0.0	86.09	0.0	86.09
	YEAR: 1985						
-	RES_UG	0.	0.	0.0	56.79	0.0	56.79
-	NRES_UG	0.	0.	0.0	119.32	0.0	119.32
-	RES_GR	0.	0.	0.0	69.65	0.0	69.65
-	NRES_GR	0.	0.	0.0	79.65	0.0	79.65
-	RES_GR_OFC	0.	0.	0.0	84.70	0.0	84.70
-	NRES_GR_OFC	0.	0.	0.0	94.70	0.0	94.70
RELATIVE COMPARISON GOAL 1							
YES	YEAR: 1983	7.	0.	0.0	7.30	0.0	7.30
YES	YEAR: 1984	7.	0.	0.0	12.54	0.0	12.54
YES	YEAR: 1985	7.	0.	0.0	5.73	0.0	5.73
RELATIVE COMPARISON GOAL 2							
YES	YEAR: 1983	7.	0.	0.0	1.77	0.0	1.77
YES	YEAR: 1984	7.	0.	0.0	2.46	0.0	2.46
YES	YEAR: 1985	7.	0.	0.0	1.50	0.0	1.50

ABSOLUTE COMPARISON GOAL 1						
YES YEAR: 1983	2.	0.	0.0	0.90	5.00	4.10
YES YEAR: 1984	2.	0.	0.0	0.0	8.00	8.00
YES YEAR: 1985	2.	0.	0.0	0.0	10.00	10.00
ABSOLUTE COMPARISON GOAL 2						
YES YEAR: 1983	0.	2.	4.10	0.0	10.00	5.90
YES YEAR: 1984	0.	2.	8.00	0.0	16.00	8.00
YES YEAR: 1985	0.	2.	10.00	0.0	20.00	10.00
ABSOLUTE COMPARISON GOAL 3						
YES YEAR: 1983	0.	2.	0.0	0.0	10.00	10.00
YES YEAR: 1984	0.	2.	1.32	0.0	15.00	13.68
YES YEAR: 1985	0.	2.	4.95	0.0	20.00	15.05
ABSOLUTE COMPARISON GOAL 4						
YES YEAR: 1983	2.	2.	0.0	0.0	0.0	0.0
YES YEAR: 1984	2.	2.	0.0	0.0	0.0	0.0
YES YEAR: 1985	2.	2.	0.0	0.0	0.0	0.0
SUPPLEMENTAL REVENUE REQUIREMENT-OVERALL						
YES	4.	0.	0.0	0.0	2000.00	2000.00

DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALSIS

******** MANAGEMENT REPORTS AVAILABLE FOR PRINTING **********

>>> INSTRUCTIONS: IN THE LEFTHAND FIELD, PRECEEDING EACH REPORT'S NAME AND DESCRIPTION, ENTER EITHER:
L IN ORDER TO PRINT THE REPORT ON THE LASER.
R IN ORDER TO PRINT THE REPORT ON REGULAR COMPUTER PAPER.
N IN ORDER TO NOT PRINT THE REPORT.

Ⓢ N TUIT - TUITION CHARGES PER CREDIT HOUR, BY STUDENT CATEGORY

Ⓣ N DEVVAR - LEVEL OF GOAL ACHIEVEMENT: DESIRED VS. ATTAINED

① R TOTCOST - TOTAL STUDENT CHARGES BY CATEGORY, ROUNDED CHARGES/CRED-HR

Ⓠ N PRIOR - PRIORITY STRUCTURE

② N COSTINFO - SUMMARY OF THE COST DATA USED IN THE ANALYSIS

③ N DDFEEOUT - SUMMARY OF THE STUDENT FEES AND DORM & DINING CHARGES

④ N ENROLOUT - SUMMARY OF THE ENROLLMENT DATA USED IN THE ANALYSIS.

⑤ N CREDOUT - CREDIT HOUR DATA USED IN THE STATE REV RQMT GOAL

Ⓘ N COMPSET - TARGETS FOR THE RELATIVE AND ABSOLUTE GOALS

Ⓜ N CHGTIME - TARGETS FOR THE PERCENTAGE CHANGE IN TUITION OVER TIME

⑥ N PEER - TARGETS FOR THE PEER INSTITUTION GOALS

Ⓝ N PEERGEN - INFORMATION REQUIRED TO SET THE PEER INSTITUTION GOALS

Ⓚ N PERCOST2 - TARGETS FOR THE PERCENTAGE-OF-AVERAGE-TOTAL-COST GOALS

Ⓙ N SUPLMREV - TARGETS FOR THE SUPPLEMENTAL REVENUE GOALS

Ⓡ N FORMAT - PROBLEM FORMULATION IN EQUATION FORM

⑦ N TRACE - VALUE OF THE DECISION VARIABLES AT EACH PRIORITY LEVEL



VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
FULL-TIME ON-CAMPUS RESIDENT UNDERGRADUATE 05/22/84

	1982	1983	1984	1985
TUITION AND FEES:				
TUITION	$1,077	$1,281	$1,542	$1,701
PERCENT INCREASE OVER PRIOR YEAR	-	18.9%	20.4%	10.3%
PERCENT OF AVERAGE COST	23.8%	26.3%	27.8%	28.6%
FEES:				
ATHLETIC ASSOCIATION FEE	66	72	78	84
BUS SYSTEM FEE	12	12	15	15
STUDENT CENTER FEE	60	63	132	138
HEALTH SERVICE FEE	66	69	72	84
A. TOTAL TUITION AND FEES	$1,281	$1,497	$1,839	$2,022
DORM & DINING:				
BASE ROOM	$444	$474	$507	$534
BASE BOARD	$1,023	$1,035	$1,107	$1,164
NEW FACILITIES	-	29	88	88
B. TOTAL ROOM AND BOARD	$1,467	$1,538	$1,702	$1,786
RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS				
TOTAL CHARGES (A)	$1,281	$1,497	$1,839	$2,022
DOLLAR INCREASE OVER PRIOR YEAR	-	$216	$342	$183
PERCENT INCREASE OVER PRIOR YEAR	-	16.9%	22.8%	10.0%
PERCENT INCREASE OVER BASE YEAR	-	16.9%	43.6%	57.8%
RESIDENT UNDERGRADUATE LIVING ON-CAMPUS				
TOTAL CHARGES (A + B)	$2,748	$3,035	$3,541	$3,808
DOLLAR INCREASE OVER PRIOR YEAR	-	$287	$506	$267
PERCENT INCREASE OVER PRIOR YEAR	-	10.4%	16.7%	7.5%
PERCENT INCREASE OVER BASE YEAR	-	10.4%	28.8%	38.6%

(1)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
FULL-TIME ON-CAMPUS NON-RESIDENT UNDERGRADUATE 05/22/84

	1982	1983	1984	1985
TUITION AND FEES:				
TUITION	$2,322	$2,781	$3,477	$3,576
PERCENT INCREASE OVER PRIOR YEAR	-	19.8%	25.0%	2.8%
PERCENT OF AVERAGE COST	51.4%	57.0%	62.6%	60.2%
FEES:				
ATHLETIC ASSOCIATION FEE	66	72	78	84
BUS SYSTEM FEE	12	12	15	15
STUDENT CENTER FEE	60	63	132	138
HEALTH SERVICE FEE	66	69	72	84
A. TOTAL TUITION AND FEES	$2,526	$2,997	$3,774	$3,897
DORM & DINING:				
BASE ROOM	$444	$474	$507	$534
BASE BOARD	$1,023	$1,035	$1,107	$1,164
NEW FACILITIES	-	29	88	88
B. TOTAL ROOM AND BOARD	$1,467	$1,538	$1,702	$1,786
NON-RESIDENT UNDERGRADUATE LIVING OFF-CAMPUS				
TOTAL CHARGES (A)	$2,526	$2,997	$3,774	$3,897
DOLLAR INCREASE OVER PRIOR YEAR	-	$471	$777	$123
PERCENT INCREASE OVER PRIOR YEAR	-	18.6%	25.9%	3.3%
PERCENT INCREASE OVER BASE YEAR	-	18.6%	49.4%	54.3%
NON-RESIDENT UNDERGRADUATE LIVING ON-CAMPUS				
TOTAL CHARGES (A + B)	$3,993	$4,535	$5,476	$5,683
DOLLAR INCREASE OVER PRIOR YEAR	-	$542	$941	$207
PERCENT INCREASE OVER PRIOR YEAR	-	13.6%	20.8%	3.8%
PERCENT INCREASE OVER BASE YEAR	-	13.6%	37.1%	42.3%

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
FULL-TIME ON-CAMPUS RESIDENT GRADUATE 05/22/84

	1982	1983	1984	1985
TUITION AND FEES:				
TUITION	$1,218	$1,428	$1,743	$1,872
PERCENT INCREASE OVER PRIOR YEAR	-	17.2%	22.1%	7.4%
PERCENT OF AVERAGE COST	27.0%	29.3%	31.4%	31.5%
FEES:				
ATHLETIC ASSOCIATION FEE	66	72	78	84
BUS SYSTEM FEE	12	12	15	15
STUDENT CENTER FEE	60	63	132	138
HEALTH SERVICE FEE	66	69	72	84
A. TOTAL TUITION AND FEES	$1,422	$1,644	$2,040	$2,193
DORM & DINING:				
BASE ROOM	-	$744	$798	$837
BASE BOARD	-	$1,191	$1,272	$1,338
NEW FACILITIES	-	29	88	88
B. TOTAL ROOM AND BOARD	-	$1,964	$2,158	$2,263
RESIDENT GRADUATE LIVING OFF-CAMPUS				
TOTAL CHARGES (A)	$1,422	$1,644	$2,040	$2,193
DOLLAR INCREASE OVER PRIOR YEAR	-	$222	$396	$153
PERCENT INCREASE OVER PRIOR YEAR	-	15.6%	24.1%	7.5%
PERCENT INCREASE OVER BASE YEAR	-	15.6%	43.5%	54.2%
RESIDENT GRADUATE LIVING ON-CAMPUS				
TOTAL CHARGES (A + B)	-	$3,608	$4,198	$4,456
DOLLAR INCREASE OVER PRIOR YEAR	-	$3,608	$590	$258
PERCENT INCREASE OVER PRIOR YEAR	-	-	16.4%	6.1%
PERCENT INCREASE OVER BASE YEAR	-	-	-	-

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF POTENTIAL ANNUAL TUITION AND FEE CHARGES
FULL-TIME ON-CAMPUS NON-RESIDENT GRADUATE 05/22/84

	1982	1983	1984	1985
TUITION AND FEES:				
TUITION	$1,353	$1,590	$1,959	$2,142
PERCENT INCREASE OVER PRIOR YEAR	-	17.5%	23.2%	9.3%
PERCENT OF AVERAGE COST	30.0%	32.6%	35.3%	36.1%
FEES:				
ATHLETIC ASSOCIATION FEE	66	72	78	84
BUS SYSTEM FEE	12	12	15	15
STUDENT CENTER FEE	60	63	132	138
HEALTH SERVICE FEE	66	69	72	84
A. TOTAL TUITION AND FEES	$1,557	$1,806	$2,256	$2,463
DORM & DINING:				
BASE ROOM	-	$744	$798	$837
BASE BOARD	-	$1,191	$1,272	$1,338
NEW FACILITIES	-	29	88	88
B. TOTAL ROOM AND BOARD	-	$1,964	$2,158	$2,263
NON-RESIDENT GRADUATE LIVING OFF-CAMPUS				
TOTAL CHARGES (A)	$1,557	$1,806	$2,256	$2,463
DOLLAR INCREASE OVER PRIOR YEAR	-	$249	$450	$207
PERCENT INCREASE OVER PRIOR YEAR	-	16.0%	24.9%	9.2%
PERCENT INCREASE OVER BASE YEAR	-	16.0%	44.9%	58.2%
NON-RESIDENT GRADUATE LIVING ON-CAMPUS				
TOTAL CHARGES (A + B)	-	$3,770	$4,414	$4,726
DOLLAR INCREASE OVER PRIOR YEAR	-	$3,770	$644	$312
PERCENT INCREASE OVER PRIOR YEAR	-	-	17.1%	7.1%
PERCENT INCREASE OVER BASE YEAR	-	-	-	-

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SUMMARY OF TUITION CHARGES PER CREDIT HOUR

	1983	1984	1985
E&G TUITION CHARGES PER CREDIT-HOUR, CALCULATED BY GP MODEL			
RESIDENT UNDERGRAD	42.67	51.63	56.79
NON-RESIDENT UNDERGRAD	92.64	115.80	119.32
RESIDENT GRADUATE	52.97	64.41	69.65
NON-RESIDENT GRADUATE	58.87	72.41	79.65
RESIDENT GRADUATE-OFC	62.97	78.09	84.70
NON-RESIDENT GRADUATE-OFC	68.87	86.09	94.70
ATHLETIC FEE PER CREDIT HOUR			
UNDERGRADUATE	2.40	2.60	2.80
GRADUATE	2.67	2.89	3.11
BUS FEE PER CREDIT HOUR			
UNDERGRADUATE	.40	.50	.50
GRADUATE	.44	.56	.56
ROUNDED TOTAL CHARGES PER CREDIT-HOUR, E&G + ATH + BUS			
RESIDENT UNDERGRAD	45.50	54.50	60.00
NON-RESIDENT UNDERGRAD	95.50	119.00	122.50
RESIDENT GRADUATE	56.00	68.00	73.00
NON-RESIDENT GRADUATE	62.00	76.00	83.00
RESIDENT GRADUATE-OFC	63.00	78.00	85.00
NON-RESIDENT GRADUATE-OFC	69.00	86.00	95.00
NET E&G TUITION CHARGES PER CREDIT-HOUR AFTER ADDITION OF ATHLETIC AND BUS FEES AND AFTER ROUNDING (1)			
RESIDENT UNDERGRAD	42.70	51.40	56.70
NON-RESIDENT UNDERGRAD	92.70	115.90	119.20
RESIDENT GRADUATE	52.89	64.56	69.33
NON-RESIDENT GRADUATE	58.89	72.56	79.33
RESIDENT GRADUATE-OFC	63.00	78.00	85.00
NON-RESIDENT GRADUATE-OFC	69.00	86.00	95.00

(1) UNDERGRADUATE CHARGES PER CREDIT-HOUR ARE ROUNDED TO THE

NEAREST $ 0.50 AND GRADUATE CHARGES ARE ROUNDED TO THE NEAREST $ 1.00. ROUNDING OCCURS AFTER THE ATHLETIC AND BUS FEES ARE ADDED TO E&G

SUMMARY OF EXPECTED ANNUAL REVENUE

	1983	1984	1985
ATHLETIC FEE REVENUE	$1,490,233	$1,617,936	$1,745,174
BUS FEE REVENUE	$248,372	$311,142	$311,638
STATE REVENUE RQMNT	$36,230,300	$43,558,937	$46,762,918
EXPECTED E&G REVENUE	36,246,165	44,384,412	47,782,466
EXTRA E&G REV FROM PT	84,901	94,781	101,380
DIFFERENCE	$100,766	$920,257	$1,120,928

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
SUMMARY OF COST INFORMATION AND DATA 05/23/84

	1983	1984	1985
ADJUSTED INSTRUCTIONAL E&G	104,515,143	119,142,901	127,722,440
TUITION AND FEE RECOVERY RATE (1)	.3500	.3685	.3689
UNADJUSTED STATE REVENUE RQMT	$36,580,300	$43,909,003	$47,112,991
MISCELLANEOUS FEES	-350,000	-350,000	-350,000
STATE REVENUE REQUIREMENT	$36,230,300	$43,559,003	$46,762,991
CHANGE IN RQMT. OVER PRIOR YEAR	17.52%	20.23%	7.36%
COMPONENTS OF THE TUITION AND FEE RECOVERY RATE			
1. STUDENT ENROLLMENT MIX (2)			
RESIDENT UNDERGRADUATE	.6629	.6565	.6525
NON-RESIDENT UNDERGRADUATE	.1906	.1884	.1870
RESIDENT GRADUATE	.1093	.1156	.1195
NON-RESIDENT GRADUATE	.0372	.0395	.0410
(2)BASED ON THE FOLLOWING WEIGHTED AVG. OF NON-RESIDENT GRADUATE STUDENT EMPLOYMENT RATES			
WEIGHTED AVG EMPLOYMENT RATE	.5000	.5000	.5000
2. COST RATIOS			
UNDERGRADUATE	.6677	.6550	.6471
GRADUATE	2.9358	2.8803	2.8456
3. POLICY RATES			
RESIDENT UNDERGRADUATE	-	25.0000	25.0000
NON-RESIDENT UNDERGRADUATE	-	75.0000	75.0000
RESIDENT GRADUATE	-	25.0000	25.0000
NON-RESIDENT GRADUATE	-	75.0000	75.0000
4. PARTIAL RECOVERY RATES (BY STUDENT CATEGORY)			
RESIDENT UNDERGRADUATE	-	.1075	.1056
NON-RESIDENT UNDERGRADUATE	-	.0926	.0908
RESIDENT GRADUATE	-	.0832	.0850
NON-RESIDENT GRADUATE	-	.0852	.0876
RECOVERY RATE	-	.3685	.3689

(2)

BASIS FOR PERCENTAGE OF AVERAGE COST GOAL:			
AVG. ANNUAL COST PER FULL-TIME STUDENT	4876	5551	5940
CONSUMER PRICE INDEX	301.80	316.89	332.73
CHANGE IN CPI	2.90%	5.00%	5.00%

(1) MAXIMUM INCREASE .025/YEAR; MAXIMUM DECREASE .0125/YEAR

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
STUDENT FEE DATA 05/23/84

	1983	1984	1985
ANNUAL EQUIVALENT STUDENTS			
ATHLETIC FEE	62,877	63,024	63,132
BUS FEE	62,093	62,228	62,328
STUDENT CENTER	66,011	66,209	66,351
HEALTH SERVICE FEE	62,093	62,228	62,328
ANNUAL STUDENT FEES			
ATHLETIC ASSOCIATION			
BASE FEE	72	78	84
DEBT SERVICE	-	-	-
TOTAL FEE	72	78	84
BUS SYSTEM			
BASE FEE	12	15	15
DEBT SERVICE	-	-	-
TOTAL FEE	12	15	15
STUDENT CENTER			
BASE FEE	63	69	72
DEBT SERVICE	-	63	63
TOTAL FEE	63	132	138
HEALTH SERVICES			
BASE FEE	69	72	78
DEBT SERVICE	-	-	6
TOTAL FEE	69	72	84
TOTAL ANNUAL FEES	$216	$297	$321

(3)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
DORM AND DINING DATA 05/23/84

	1983	1984	1985
UNDERGRADUATE STUDENTS			
UPPER QUAD	1,950	1,950	1,950
LOWER QUAD	4,380	4,380	4,380
PRAIRIE QUAD	1,969	1,969	1,969
NEW UNDERGRADUATE DORM	342	342	342
SPECIAL PURPOSE HOUSING	108	108	108
TOTAL UNDERGRADUATE STUDENTS	8,749	8,749	8,749
GRADUATE STUDENTS			
GRADUATE SINGLE OCC	36	36	36
GRADUATE DOUBLE OCC	120	120	120
TOTAL GRADUATE STUDENTS	156	156	156
TOTAL ON CAMPUS	8,905	8,905	8,905
BASE DORMITORY CHARGES (QUARTERLY)			
UPPER QUAD	$141	$151	$158
LOWER QUAD	159	170	179
PRAIRIE QUAD	159	170	179
NEW UNDERGRADUATE DORM	220	236	248
SPECIAL PURPOSE HOUSING	223	238	250
UNDERGRADUATE AVERAGE	$158	$169	$178
GRADUATE SINGLE OCC	346	370	389
GRADUATE DOUBLE OCC	219	234	246
GRADUATE AVERAGE	$248	$266	$279
BASE DINING CHARGES (QUARTERLY)			
UNDERGRADUATE	$345	$369	$388
GRADUATE	397	424	446
NEW DEBT SERVICE (QUARTERLY)			
PROJECT 1	$10	$10	$10
PROJECT 2	-	20	20
PROJECT 3	-	-	-

TOTAL DEBT SERVICE	$10	$29	$29

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
DORM AND DINING DATA 05/23/84

SUMMARY OF ANNUAL CHARGES

	1983	1984	1985
UNDERGRADUATE			
BASE DORMITORY	474	507	534
BASE DINING	1035	1107	1164
BASE DORM & DINING	1509	1614	1698
DEBT SERVICE	29	88	88
TOTAL DORM & DINING	1538	1702	1786
GRADUATE			
BASE DORMITORY	744	798	837
BASE DINING	1191	1272	1338
BASE DORM & DINING	1935	2070	2175
DEBT SERVICE	29	88	88
TOTAL DORM & DINING	1964	2158	2263

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)
USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
REGULAR SESSION - FALL QUARTER ENROLLMENT			
UNDERGRADUATE			
FULL-TIME RESIDENT	13,617	13,502	13,447
FULL-TIME NON-RESIDENT	3,993	3,953	3,929
PART-TIME RESIDENT	527	524	519
PART-TIME NON-RESIDENT	73	73	73
TOTAL UNDERGRADUATE	18,210	18,052	17,968
GRADUATE			
FULL-TIME RESIDENT	864	945	982
FULL-TIME NON-RESIDENT	1,713	1,826	1,906
PART-TIME RESIDENT	462	489	511
PART-TIME NON-RESIDENT	279	294	305
TOTAL GRADUATE	3,318	3,554	3,704
TOTAL REGULAR SESSION	21,528	21,606	21,672
SUMMER SESSION			
UNDERGRADUATE			
FULL-TIME RESIDENT	1,997	1,977	1,948
FULL-TIME NON-RESIDENT	432	427	402
PART-TIME RESIDENT	2,473	2,451	2,421
PART-TIME NON-RESIDENT	572	566	559
TOTAL UNDERGRADUATE	5,474	5,421	5,330
GRADUATE			
FULL-TIME RESIDENT	427	447	462
FULL-TIME NON-RESIDENT	614	641	669
PART-TIME RESIDENT	507	530	553
PART-TIME NON-RESIDENT	389	407	425
TOTAL GRADUATE	1,937	2,025	2,109
TOTAL SUMMER SESSION	7,411	7,446	7,439

(4)

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)
USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
OFF-CAMPUS			
GRADUATE			
FULL-TIME RESIDENT	62	63	65
FULL-TIME NON-RESIDENT	14	14	14
PART-TIME RESIDENT	1,342	1,355	1,375
PART-TIME NON-RESIDENT	179	181	184
TOTAL OFF-CAMPUS	1,597	1,613	1,638

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"PROJECTED" ENROLLMENT DATA (HEADCOUNT)
USED TO CALCULATE THE STATE REVENUE REQUIREMENT

	1983	1984	1985
REGULAR SESSION - FALL QUARTER ENROLLMENT			
UNDERGRADUATE			
FULL-TIME RESIDENT	13,617	13,502	13,447
FULL-TIME NON-RESIDENT	3,993	3,953	3,929
PART-TIME RESIDENT	527	524	519
PART-TIME NON-RESIDENT	73	73	73
TOTAL UNDERGRADUATE	18,210	18,052	17,968
GRADUATE			
FULL-TIME RESIDENT	864	945	982
FULL-TIME NON-RESIDENT	1,713	1,826	1,906
PART-TIME RESIDENT	462	489	511
PART-TIME NON-RESIDENT	279	294	305
TOTAL GRADUATE	3,318	3,554	3,704
TOTAL REGULAR SESSION	21,528	21,606	21,672
SUMMER SESSION			
UNDERGRADUATE			
FULL-TIME RESIDENT	1,997	1,977	1,948
FULL-TIME NON-RESIDENT	432	427	402
PART-TIME RESIDENT	2,473	2,451	2,421
PART-TIME NON-RESIDENT	572	566	559
TOTAL UNDERGRADUATE	5,474	5,421	5,330
GRADUATE			
FULL-TIME RESIDENT	427	447	462
FULL-TIME NON-RESIDENT	614	641	669
PART-TIME RESIDENT	507	530	553
PART-TIME NON-RESIDENT	389	407	425
TOTAL GRADUATE	1,937	2,025	2,109
TOTAL SUMMER SESSION	7,411	7,446	7,439

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"ACTUAL" ENROLLMENT DATA (HEADCOUNT)

USED TO CALCULATE EXPECTED CREDIT HOURS AND STUDENT FEES

	1983	1984	1985
OFF-CAMPUS			
GRADUATE			
FULL-TIME RESIDENT	62	63	65
FULL-TIME NON-RESIDENT	14	14	14
PART-TIME RESIDENT	1,342	1,355	1,375
PART-TIME NON-RESIDENT	179	181	184
TOTAL OFF-CAMPUS	1,597	1,613	1,638

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
ENROLLMENT DATA 05/23/84

"PROJECTED" ENROLLMENT (FTE, BASED ON FALL ENROLLMENT)
USED TO CALCULATE STATE REVENUE REQUIREMENT

	1983	1984	1985
UNDERGRADUATE			
RESIDENT	14,229	14,110	14,050
NON-RESIDENT	4,090	4,050	4,026
TOTAL UNDERGRADUATE	18,319	18,160	18,076
GRADUATE			
RESIDENT	1,014	1,097	1,142
NON-RESIDENT	1,524	1,622	1,691
RESIDENT OFF-CAMPUS	534	539	547
NON-RESIDENT OFF-CAMPUS	73	74	75
TOTAL GRADUATE	3,145	3,332	3,456
GRAND TOTAL	21,464	21,492	21,532
ON CAMPUS	20,858	20,879	20,909
OFF CAMPUS	607	613	622
SUMMER UNDERGRAD	1,303	1,290	1,269
SUMMER GRADUATE	461	482	502
TOTAL ANNUAL FTE	64,303	64,393	64,510
TOTAL ANNUAL FTE CH	935,139	934,805	935,408

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
EXPECTED CREDIT HOURS BY STUDENT CATEGORY 05/23/84

	1983	1984	1985
REGULAR AND SUMMER SESSIONS			
RESIDENT UNDERGRADUATE	438,663	434,943	432,796
NON-RESIDENT UNDERGRADUATE	124,875	123,627	122,638
RESIDENT GRADUATE	76,453	81,799	85,216
NON-RESIDENT GRADUATE	13,930	14,798	15,441
OFF-CAMPUS			
RESIDENT GRADUATE	16,484	16,654	16,929
NON-RESIDENT GRADUATE	538	543	550
TOTAL EXPECTED CREDIT HOURS	670,941	672,364	673,570

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS
PEER INSTITUTION GOAL TARGET VALUES

YEAR	1983	1984	1985
****	RESIDENT_UNDERGRADUATE		
EXPECTED PEER CHARGES	3549.	3789.	3894.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	1538.	1702.	1786.
PEER INST GOAL TARGET	1795.	1790.	1787.
****	NON-RESIDENT_UNDERGRADUATE		
EXPECTED PEER CHARGES	4494.	4702.	4826.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	1538.	1702.	1786.
PEER INST GOAL TARGET	2740.	2703.	2719.
****	RESIDENT_GRADUATE		
EXPECTED PEER CHARGES	1440.	1524.	1574.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	1224.	1227.	1253.
****	NON-RESIDENT_GRADUATE		
EXPECTED PEER CHARGES	2356.	2463.	2530.
LESS FEE CHARGES	216.	297.	321.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	2140.	2166.	2209.
****	RESIDENT_GRADUATE_OFF-CAMPUS		
EXPECTED PEER CHARGES	200.	225.	250.
LESS FEE CHARGES	0.	0.	0.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	200.	225.	250.
****	NON-RESIDENT_GRADUATE_OFF-CAMPUS		
EXPECTED PEER CHARGES	280.	310.	340.
LESS FEE CHARGES	0.	0.	0.
LESS DORM & DINING	0.	0.	0.
PEER INST GOAL TARGET	280.	310.	340.

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
DECISION SUPPORT SYSTEM FOR TUITION AND FEE POLICY ANALYSIS

SOLUTION VALUES FOR THE DECISION VARIABLES AT EACH PRIORITY LEVEL

VALUES AT EACH STEP IN THE SEQUENTIAL LINEAR PROGRAMMING

SOLUTION TO THE GOAL PROGRAMMING PROBLEM

PRIOR LEVEL	RES UG	NRES UG	RES GR	NRES GR	RES GR-OFC	NRES GR-OFC
YEAR 1						
1	42.13	92.86	52.97	58.87	75.33	81.23
2	42.09	92.64	55.47	60.47	65.47	70.47
3	42.67	92.64	52.97	58.87	62.97	68.87
4	42.67	92.64	52.97	58.87	62.97	68.87
5	42.67	92.64	52.97	58.87	62.97	68.87
6	42.67	92.64	52.97	58.87	62.97	68.87
7	42.67	92.64	52.97	58.87	62.97	68.87
8	42.67	92.64	52.97	58.87	62.97	68.87
9	42.67	92.64	52.97	58.87	62.97	68.87
YEAR 2						
1	42.09	92.64	55.47	60.47	65.47	70.47
2	42.67	92.64	52.97	58.87	62.97	68.87
3	42.67	92.64	52.97	58.87	62.97	68.87
4	42.67	92.64	52.97	58.87	62.97	68.87
5	51.24	119.85	65.15	72.41	0.0	101.53
6	51.35	111.71	61.62	72.41	75.30	86.09
7	51.63	115.80	64.41	72.41	78.09	86.09
8	51.63	115.80	64.41	72.41	78.09	86.09
9	51.63	115.80	64.41	72.41	78.09	86.09
YEAR 3						
1	42.67	92.64	52.97	58.87	62.97	68.87
2	42.67	92.64	52.97	58.87	62.97	68.87
3	42.67	92.64	52.97	58.87	62.97	68.87
4	51.35	111.71	61.62	72.41	75.30	86.09
5	51.63	115.80	64.41	72.41	78.09	86.09
6	51.63	115.80	64.41	72.41	78.09	86.09
7	56.49	112.97	67.78	79.65	82.83	94.70
8	56.79	119.32	69.65	79.65	84.70	94.70
9	56.79	119.32	69.65	79.65	84.70	94.70

The two page vita has been removed from the scanned document. Page 1 of 2

The two page vita has been removed from the scanned document. Page 2 of 2