A Knowledge Based Expert System for the

Design of Reinforced Concrete Beams

by

Thien Pin Wong

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Civil Engineering

APPROVED:

Dr. Kamal B. Rojiani, Chairman

Dr. Richard M. Barker

Dr. Siegfried M. Holzer

March, 1988

Blacksburg, Virginia

A Knowledge Based Expert System for the Design of Reinforced Concrete Beams

by

Thien Pin Wong

Dr. Kamal B. Rojiani, Chairman

Civil Engineering

(ABSTRACT)

The design of reinforced concrete beams involves the selection of design parameters such as beam dimensions and reinforcement details which result in a safe and economical section. This process of design consists of three stages: preliminary design, structural analysis, and detailed design which includes the selection of dimensions, reinforcement, and stirrups. The design process is an iterative one where considerable judgement and experience are required. This is an ideal situation for the application of expert system technology.

A knowledge based expert system called BEAMDES was developed for the flexure design of reinforced concrete beams in accordance with ACI 318-83 specifications. The expert system was developed using the micro-computer based expert system shell, Insight 2+. BEAMDES can be used to design both rectangular and tee sections. The beams can be simply supported, cantilevered, or continuous.

The results obtained from BEAMDES were tested against several example problems for both simply supported and continuous beams. It was found that the designs recommended by the system were similar to those of the example problems.

Acknowledgements

I would like to express my gratitude to my advisor, Dr. Kamal Rojiani for his guidance throughout the research and preparation of this thesis.

I would also like to thank Dr. Richard Barker and Dr. Siegfried Holzer for their service on my advisory committee.

Last, and certainly not the least, I would like to thank my family for their support throughout my education.

Table of Contents

List of Illustrations

Chapter I. Background of Research

1.1 Introduction

Computer application to structural design has been extensive but limited to algorithmic analysis, proportioning and graphical presentation of results. However, a large number of problems encountered in structural design are not amenable to purely algorithmic solutions. These problems are often ill-structured and require considerable experience and judgement to deal with them. Artificial Intelligent methods provide a mean for using the computer to assist in the solution of these ill-structured problems. This has led to the development of knowledge based expert system (KBES).

Reinforced Concrete beam design is an iterative process which requires a good deal of judgement in order to obtain a " good " solution. A " good " solution means that the beam size and reinforcement selected should meet all the requirements specified by building codes and yet result in an economical solution. If one is given the necessary information concerning the cross-section of a beam and then asked to evaluate its capacity then the problem is pure algorithmic. This process consists of a set of well-defined steps to arrive at a unique answer. On the other hand, if one is given the required capacity or function of a beam and asked to design it then the problem is different. In this case, there is no unique answer and also, there are no well defined-steps which the designer could follow. Therefore, he has to make use of his judgement and experience in order to come up with a " good " design. This process is iterative and is best suited for expert system application.

1.2 Objective

The main objective of this study is to develop a knowledge based expert system called " BEAMDES " using an expert system shell (tool) on microcomputer. The knowledge base consists of general facts and heuristic for the flexural design of reinforced concrete beams. The expert system is limited to the design of simple, continuous and cantilever beams of both rectangular and tee sections. The expert system shell used for developing this knowledge base is the microcomputer based expert system shell Insight 2+ [15].

Chapter II. Literature Review

2.1 Overview of Expert Systems

Artificial Intelligence (AI) is a branch of computer science concerned with making computers act more like human beings. Efforts in the application of AI methods to intelligent problem solving led to the development of expert systems. Expert systems are relatively new to Civil Engineering and there is considerable activity in the development of expert system in Civil Engineering. According to Jenning [16], whether or not expert systems are new is clearly debatable. Programmers have been developing knowledge intensive programs that mimic human decision making processes for years. But if expert systems are defined as programs with true AI that are able to extract and accumulate wisdom, and adapting to evolving knowledge then they are relatively new. However new expert systems are to Civil Engineers, their potential role has been extensively publicized in the last few years.

Expert systems basically use domain specific knowledge and are also referred to as " knowledge based expert systems " (KBES). KBES are computer programs which explicitly capture knowledge of human experts and utilize that knowledge to make knowledgable advice, decision and judgement. Adeli [1] pointed out the difference between the traditional computer programs and expert systems. He said that expert systems are knowledge-intensive programs and expert knowledge is usually divided into many separate rules. Those rules are separated from the methods for applying the knowledge to the current problem. These methods are referred to as inference mechanism, reasoning mechanism, or rule interpreter. Expert systems are also highly interactive, and have a user-friendly/intelligent interface.

There are many advantages to using expert systems in engineering. Expert systems provide a distribution of expertise, such that the user can gain access to knowledge and logical processes that are known to others (the experts). Through this distribution, a greater degree of consistency can be achieved and maintained. Higher accuracy and performance levels will be attained due to a continuous availability of high-level knowledge. Also because of the continuous availability of an expert, time-delays can be eliminated. Cost may also be reduced as there are no consultation fees, only cost of purchase or development of the expert system.

2.2 Features of Knowledge Based Expert Systems.

KBES typically consists of four major components. They are the knowledge base, inference engine, user interface, and database. Figure 1 show a schematic view of an expert system.

Knowledge base. The knowledge base contains knowledge of facts and heuristics (accumulated by an expert over the years). Production rules are the most widely used form of representing knowledge in the knowledge base. Some other forms such as frames (concepts) [26] and semantic nets [13] are also available for representing knowledge. In a rule-based system representation, knowledge is incorporated in IF-THEN rules. These are condition-action or condition-conclusion pairs. If the condition is true then a conclusion is reached or an action is taken.

Inference engine. The inference engine is the mechanism that controls the execution of an expert system. Its main function is to monitor the execution of the program by utilizing the knowledge base to accelerate the solution process of the problem at hand. The two commonly used strategies in designing the inference engine are backward chaining and forward chaining. In backward chaining, a specific hypothesis, conclusion, action or goal is pursued by searching for the antecedents (conditions) which would support that goal. In forward chaining, the inference engine arrives at a hypothesis or goal by attempting to match the contextual data to a pattern or template described by the rules of the knowledge base.

User interface. The user interface provides the user-system interactions. These include help facilities, explanation facilities and information extracting facilities.

Context. The context is the work area of the system. It reflects all the information that has been generated during a particular program execution.

Expert systems should also provide documentation, explanation and advice. The documentation should include not only comments provided by the programmer but also derived information explaining the logic underlying different solution strategies. An explanation facility should be available so as to permit the user to understand why certain questions are asked and gives a justification for specific conclusions or recommendations. Expert systems may also be able to modify existing rules and learn new rules.

2.3 AI Languages and Tools for Development of Expert Systems.

Expert systems in the United States are mostly written in a computer language called LISP (LISt Processing). LISP [5] was invented by McCarthy in 1960 [24]

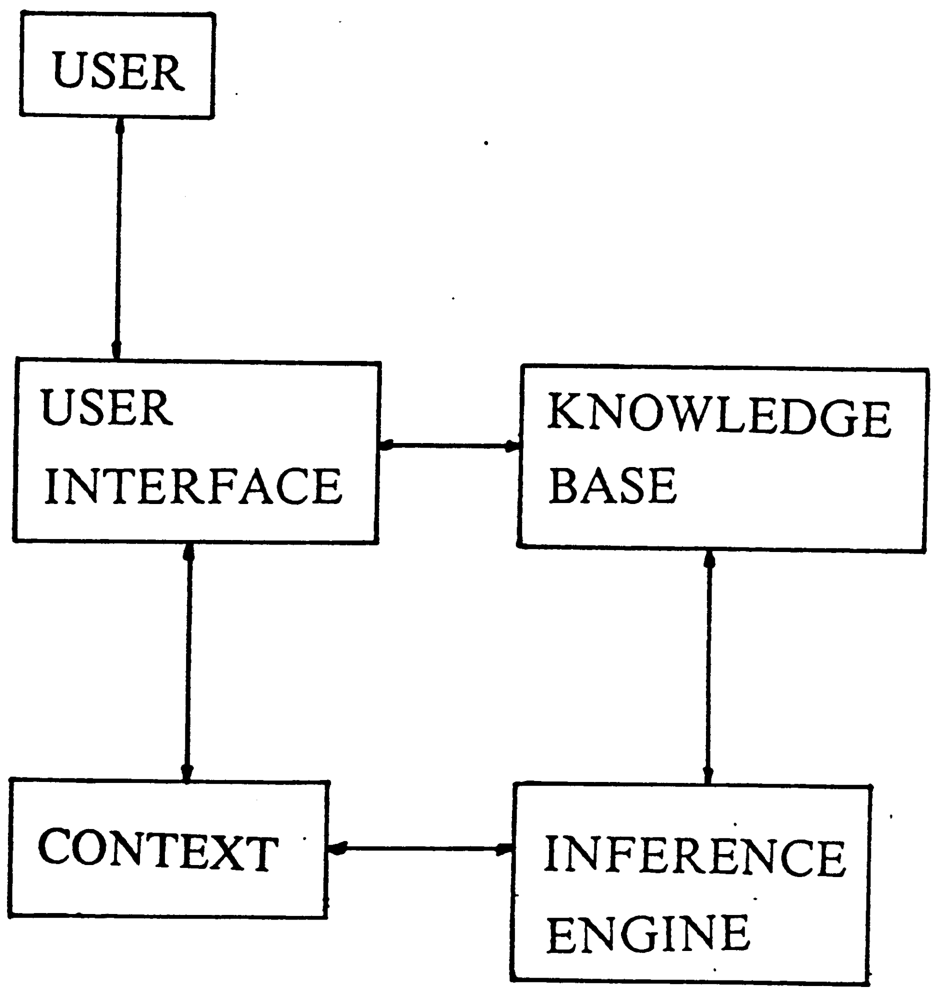


Figure 1. Schematic View of an Expert System.

for nonnumeric computations and is now the most widely-used language among AI researchers. The Japanese and Europeans prefer to use another language called PROLOG (PROgramming in LOGic) [23] developed in France in 1971. PROLOG was selected by the Japanese for their fifth generation computer project which focuses on AI [1]. Even though LISP is most commonly used by researchers in the U.S. and has been approved by the U.S. Department of Defense, a number of companies including DEC, Xerox, and Texas Instrument are developing expert systems using PROLOG [1]. Since LISP and PROLOG offer definite advantages [9,18] over procedural languages (such as FORTRAN, BASIC and PASCAL) with regard to symbolic processing, it is not surprising to find expert systems developed in these procedural languages. A description of other AI languages such as PLANNER, CONNIVER, QLISP, POP-2, SAIL and FUZZY is given in Ref. [5], pp. 30-75.

In developing expert systems for engineering applications, it is highly desirable to make use of existing programs to assist the development process. To facilitate this, expert systems shells (tools) have recently been developed and are available commercially. With these shells, developing an expert system is much more easier than using AI languages. Some of the expert system development frame works include EXPERT EDGE, DECIDING FACTOR, INSIGHT, RULE-MASTER and EXPERT. Many of the currently available shells can be implemented on microcomputers [9] but some require main-frames or special hardware. A comparison of some expert system development tools and their applicability to the solution of Civil Engineering problems is given in Ref. [18].

2.4 *Development of Expert Systems*

Successful expert systems have been developed in the field of medical diagnosis, mineral exploration, and the design of computer layouts. Examples of some of the earlier expert systems are DENDRAL, MYCIN, PROSPECTOR, and XCON.

DENDRAL. DENDRAL [5] was developed at Stanford University in the 1960's. Its aim is to hypothesize on the possible molecular structure of a compound. A chemist, in order to establish the constituent atoms and relative ratios of an unknown compound, has to perform analytical tests and experiments. There is no scientific algorithm by which the chemist can proceed. DENDRAL was originally designed to enumerate all possible configurations of a set of atoms observing the rules of chemical valence and provide the chemist with a checklist of possibilities. The power of DENDRAL had since then been significantly extended. Due to its usefulness and proven reliability, its use is rapidly expanding.

MYCIN. MYCIN [5] was developed at Stanford University in the 1970's. It was designed to help physicians in the diagnosis and treatment of patients such as those suffering from meningitis and bacteria infections. MYCIN assists the physician to decide if the patient is suffering from these diseases, what is the cause of the illness and what drugs are appropriate to cure the illness. MYCIN displays conclusions and its degree of certainty about them, the line of reasoning and gives appropriate references to articles or publications that serve to support its conclusions. Various evaluations made at Stanford indicate that MYCIN is as good as or better than most highly skilled specialists of infectious diseases.

PROSPECTOR. PROSPECTOR [8,10] is a consulting system designed to imitate the reasoning process of an expert exploration geologist for finding an ore deposit in a particular region. It was developed at the Stanford Research Institute. PROSPECTOR contains rules to accommodate several distinct geological models. Those models were tested against known sites of exploration and against judgement of experts. PROSPECTOR was found to be within 7 to 10 percent in agreement with those experts.

XCON. Also known as R1 [1,27] was developed by John McDermott and his colleagues at the Carnegie-Mellon University from 1980. It was developed for designing the configuration of VAX-11/780 computer components for Digital Equipment Corporation (DEC) according to customers' requests.

As mentioned earlier, expert systems are relatively new to Civil Engineering, and their application to structural design is a very recent development. Inspired by the success of expert systems in other fields, structural engineers have begun to look at implementing expert systems in structural design.

As more researchers began to realize the potential application of expert systems to Civil Engineering problems, studies have been conducted on its application to different Civil Engineering problems. Over the last few years a number of expert systems for Civil Engineering had been developed or are still being developed. A brief description of some of those expert system is given below:

HI-RISE. This KBES is being developed by Maher and Fenves at Carnegie-Mellon University. HI-RISE [29] is implemented in PSRL, a frame based production system language being developed at Carnegie-Mellon University. It acts as an aid for the preliminary structural design of high rise buildings. HI-RISE contains declaration and procedural knowledge that is stored in lists to represent a physical hierarchy of known structural types such as lateral load resisting systems or gravity load resisting systems. HI-RISE presents all structurally feasible configurations to the user indicating which configuration is the best and allows the user to select one of them.

SACON. This expert system is designed to act as an automated consultant to advice non-expert engineers in the use of a general structural analysis program. SACON (for Structural Analysis CONsultant) [6] was developed at Stanford University. SACON was implemented in the EMYCIN system which is essentially the MYCIN program without the medical diagnosis knowledge. SACON identifies the most appropriate analysis class to be performed and associated analysis recommendations. It also determines the controlling stress, deflection, nonlinear behavior of substructures and estimates the non-dimensional stress and deflection bounds for each substructure based on its boundary conditions and loading. SACON also constructs a context tree from a fixed hierarchy to represent the relationships between the objects.

DURCON. DURCON (for DURable CONcrete) is being developed by Clifton [7] to give recommendations on the selection of constituents for durable concrete. There are four major causes of deterioration of concrete addressed by DURCON. They are freezing and thawing, corrosion of reinforcing steel, sulfate attack and alkali-aggregate reactions. The knowledge in DURCON is largely based on the ACI Guide to Durable Concrete and is incorporated in production rules. The expert system is written in Pascal and uses a forward chaining inference procedure.

HOWSAFE. HOWSAFE [21] is an expert system which was developed using the expert system shell " The DECIDING FACTOR ". It is designed to evaluate the safety of a construction firm. Its knowledge is represented like an inverted tree, with the top-level diagnosis on the top, supported by lower level inferences, pyramiding down three or four levels to factual assertions, whose validity can be objectively evaluated by the user at the bottom end of each branch. In HOWSAFE, users are asked to give a confidence level or degree of belief to an answer as its application involves the diagnosis of social systems rather than mechanical systems where mostly clear cut responses are expected.

A prototype expert system which aids in the selection of a preliminary configuration of wood trusses is described in Ref. [2]. It recommends the type of wood truss that would be most suitable for a particular application. The applicability of expert systems in synthesizing structural slabs is presented in Ref. [21]. The system developed is able to suggest possible slab configurations based on some simple facts. A microcomputer based expert system used to determine the effects of vehicles and overloaded vehicles on simple span bridges with reinforced concrete deck and prestressed concrete I-beams is presented in Ref. [17]. The use of expert systems for the investigation of reinforced concrete structures subjected to severe loads such as blast or shock is described in Ref. [19]. The developed expert system is able to determine the failure modes of the structure and what causes it to fail by assessing the kind of damages to the structure. A prototype expert system for the design of reinforced concrete columns is given in Ref. [31]. This prototype is part of an expert system which will be developed for the design of whole concrete structural systems.

Chapter III. Flexure Design of Reinforced Concrete Beams

In this chapter, the method of design used by BEAMDES is described. The design is in accordance with the ACI 318-83 Building code [3]. For a more detailed explanation see Nawy [25] Chapter 5. All the variables in the equations given in this chapter have kip and inch units. Refer to Appendix B for notations.

3.1 Singly Reinforced Rectangular Section

The procedure for designing a singly reinforced rectangular section, given span length, adjacent span lengths, factored moments, concrete and steel yield strength, and support conditions, are as follows:

1. Assume a reinforcement ratio of $\rho = 0.5\overline{\rho}_b$, where $\overline{\rho}_b$ is the balance steel ratio and is given by

$$\overline{\rho}_b = \beta_1 \frac{0.85 f_c'}{f_y} \frac{87}{87 + f_y} \qquad [3.1]$$

$$\beta_1 = \begin{cases} 0.85 & \text{if} \quad f_c' < 4 \text{ ksi} \\ 0.85 - 0.05(f_c' - 4) & \text{if} \quad 4 \le f_c' \le 8 \text{ ksi} \\ 0.65 & \text{if} \quad f_c' > 8 \text{ ksi} \end{cases} \qquad [3.2]$$

2. An overall beam depth h, can be assumed using the ACI code guidelines for deflection control, Table 9.5(a). Assume a width b, by taking $b \simeq \frac{d}{2}$, where $d = h - \text{cover} - 1.5$. The reduction of 1.5 inches is for stirrups and half of the diameter of the main reinforcement.

3. Calculate the weight of the beam and the moments due to beam weight.

4. Check the assumed effective depth against required by using the following formulae:

$$\omega = \rho \frac{fy}{f_c'} \qquad [3.3]$$

$$R = \omega f_c'(1 - 0.59\omega) \qquad [3.4]$$

$$d = \sqrt[3]{\frac{M}{\phi r R}} \qquad [3.5]$$

where $r = 0.5$ [3.6]

If the required effective depth is less than the assumed value then the preliminary section is adequate. Otherwise, assume a new effective depth equal or greater than the required depth and repeat Steps 1, 2 and 4.

5. Calculate the area of reinforcement required from $A_s = \rho bd$ and select reinforcing bars and compute the actual area of reinforcement.

6. Analyze the section as follows:.

   a. Check the steel ratio, that is, ρ must be greater than ρ_{min} and less than ρ_{max}, ($\rho_{min} < \rho \leq \rho_{max}$), where

$$\rho = \frac{A_s}{bd} \qquad [3.7]$$

$$\rho_{min} = \frac{200}{f_y} \qquad [3.8]$$

$$\rho_{max} = 0.75\overline{\rho}_b \qquad [3.9]$$

   If $\rho \leq \rho_{min}$, increase the area of reinforcement until it is greater than one third of that required by analysis. If $\rho > \rho_{max}$, then enlarge the section and repeat Steps 2, 4, 5, and 6.

   b. Check capacity using the following formulae:

$$a = \frac{A_s f_y}{0.85 f_c' b} \qquad [3.10]$$

$$\phi M_n = \phi A_s f_y \left(d - \frac{a}{2}\right) \qquad [3.11]$$

If ϕM_n is greater than the total factored moment M_u then the section is adequate, otherwise, increase area of reinforcement and repeat Steps 6a and 6b or enlarge section and repeat Steps 5 and 6.

3.2 Singly Reinforced Tee Sections

The trial and adjustment procedure for designing a singly reinforced tee section is as follows:

1. A preliminary rectangular section is estimated using Steps 1 through 4 of Section 3.1. This estimated rectangular section will form part of the web of the tee section.

2. Check whether the span/depth ratio is reasonable, namely between 12 and 18. If not, adjust the preliminary section.

3. Calculate the flange width as follows:

$$b \leq \begin{cases} \dfrac{L}{4} \\ b_w + 16h_f \\ \dfrac{L_1 + L_2}{2} \end{cases} \qquad [3.12]$$

4. Estimate area of reinforcement required by

$$A_s = \frac{M_n}{f_y jd} \qquad [3.13]$$

where $jd \simeq 0.9d$

and select suitable reinforcement.

5. Check steel ratio, $\rho_w > \rho_{min}$ and $\rho \leq \rho_{max}$ where

$$\rho = \frac{A_s}{bd} \qquad [3.14]$$

$$\rho_w = \frac{A_s}{b_w d} \qquad [3.15]$$

$$\rho_{max} = 0.75\rho_b \qquad [3.16]$$

$$\rho_f = \frac{0.85 f_c'(b - b_w)h_f}{f_y b_w d} \qquad [3.17]$$

$$\rho_b = \frac{b_w}{b}(\overline{\rho}_b + \rho_f) \qquad [3.18]$$

If $\rho_w \leq \rho_{\min}$ then increase the area of reinforcement until it is greater than one third of that required by analysis. If $\rho > \rho_{\max}$, enlarge the section and repeat Steps 2, 3, and 4. Otherwise, the steel ratio is within the limits, and the design is acceptable.

6. Check if the neutral axis lies within or outside the flange so as the section can be analyzed as appropriate. The location of the neutral axis can be obtained from,

$$c = \frac{1.18\overline{\omega}d}{\beta_1} \qquad [3.19]$$

$$a = \beta_1 c \qquad [3.20]$$

a. If $a < h_f$ then the beam can be analyzed as a rectangular section with a width b equal to the flange width determined in Step 3.

b. If $a \geq h_f$ then the beam has to be analyzed as a tee section as follows:

$$A_{sf} = \frac{0.85 f_c'(b - b_w)h_f}{f_y} \qquad [3.21]$$

$$a = \frac{(A_s - A_{sf})f_y}{0.85 f_c' b_w} \qquad [3.22]$$

$$M_{n1} = (A_s - A_{sf}) f_y \left(d - \frac{a}{2} \right) \qquad [3.23]$$

$$M_{n2} = A_{sf} f_y \left(d - \frac{h_f}{2} \right) \qquad [3.24]$$

$$\phi M_n = \phi (M_{n1} + M_{n2}) \qquad [3.25]$$

ϕM_n has to be greater than the total factored moment M_u , otherwise, increase reinforcement and repeat from Step 4 or enlarge section and repeat from Step 2.

3.3 Doubly Reinforced Rectangular Section

BEAMDES will only design a doubly reinforced rectangular section at the negative moment regions (supports). The size of the beam at these regions is taken to be the same as for positive moment region (near midspan), therefore, the beam dimensions are known. The reinforcement is selected as follows:

1. Approximately half of the positive moment reinforcement is carried through the supports as compression reinforcement if there are 3 or more bars. All of the reinforcement from the positive moment is carried through the support if there are only two bars.

2. Estimate area of reinforcement required using the following equations:

$$M_{n2} = A_s' f_y (d - d') \quad [3.26]$$

$$M_{n1} = M_n - M_{n2} \quad [3.27]$$

$$A_{s1} = \frac{M_{n1}}{f_y \, jd} \quad [3.28]$$

$$jd \simeq 0.85d \quad [3.29]$$

$$A_s = A_{s1} + A_s' \quad [3.30]$$

and select a suitable set of reinforcement.

3. Check steel ratio.

   a. $\rho \leq \rho_{\min}$. If $\rho \leq \rho_{\min}$ then increase area of reinforcement until it is greater than one third of that required by analysis.

   b. $\rho > \rho_{\max}$ where

$$\rho_{\max} = 0.75\overline{\rho}_b + \rho'\left(\frac{f_s'}{f_y}\right) \quad [3.31]$$

$$\rho' = \frac{A_s'}{b_w d} \quad [3.32]$$

$$\text{If} \quad (\rho - \rho') \geq \beta_1 \frac{0.85 f_c' d'}{f_y d} \frac{87}{87 - f_y} \qquad [3.33]$$

$$\text{then} \quad f_s' = f_y \qquad [3.34]$$

$$\text{else} \quad f_s' = 87\left[1 - \frac{0.85 f_c' \beta_1 d'}{(\rho - \rho') f_y d}\right] < f_y \qquad [3.35]$$

If $\rho > \rho_{max}$ then enlarge section and repeat from Step 2.

4. Check capacity.

$$\phi M_n = \phi\left[(A_s f_y - A_s' f_s')\left(d - \frac{a}{2}\right) + A_s' f_s'(d - d')\right] \qquad [3.36]$$

$$a = \frac{(A_s f_y - A_s' f_s')}{0.85 f_c' b} \qquad [3.37]$$

If ϕM_n is smaller than the total factored moment M_u , increase the reinforcement area and repeat from Step 3 or enlarge section and repeat from Step 2.

Chapter IV. Description of Insight 2+ and The Expert System.

4.1 Overview of Insight 2+

Insight 2+ is an expert system building tool designed to assist in the development of knowledge based expert systems. It is distributed by Level Five Research, Inc.. It is designed to run on the IBM PC/XT, and PC/AT. It requires two double sided floppy disk drives and a minimum of 256k bytes of RAM and the PC-DOS or MS-DOS version 2.0 or later operating system. Insight 2+ will also run on most IBM compatible computers. Although Insight 2+ will run with the above configuration, 640k bytes RAM and a hard-disk drive is recommended to take full advantage of all its features.

When building expert systems with Insight 2+, one specifies a goal or hierarchy of goals which are then proven or disproven by a network of interdependent rules. Insight 2+ makes use of a simple yet very versatile knowledge representation language called Production Rule Language (PRL). These rules can be thought of as rules which an expert uses to define tests needed to prove a hypothesis. The use of PRL results in a cause and effect organization of knowledge bases that is very similar to the human thought process. When using PRL, ordering of rules in a knowledge base is not important and rules can be added at any time. Rules can take many forms, but all rules must have as a minimum, three parts: the rule name, supporting conditions or procedure statements and a conclusion. A typical PRL rule is:

```
RULE name
IF   condition 1
AND  condition 2
 .     .
 .     .
THEN conclusion
```

PRL also allows the user to specify procedural rules, the execution of which are not dependent on the satisfaction of any antecedents (IF condition). PRL also allows mathematical operations and equations in its rules. Numeric data can be manipulated using boolean, arithmetic, or other mathematical functions. The mathematical functions supported in Insight 2+ include relational operators, trigonometric and logarithmic functions.

Insight 2+ provides three functions for communicating additional information to the user. These are the EXPAND, DISPLAY, and TEXT functions. The EXPAND function provides a means for the knowledge engineer to provide explanatory information for a supporting condition, goal, or conclusion. The DISPLAY facility allows the expert engineer to specify that a given body of text will be displayed when the rule containing the text is processed without the user requesting for it. TEXT is useful whenever the expert engineer wants to phrase the queries Insight 2+ presents to the user rather than using the preformatted queries.

Insight 2+ knowledge bases may activate other Insight 2+ knowledge bases and communicate with them. Another feature of Insight 2+ is that it allows for the execution of programs written in another language from within the expert system. This is useful when computation are necessary to support a condition. It is very cumbersome to perform computations within Insight 2+. The external programs can be written in any procedural language provided they are compiled into either *.COM or *.EXE files.

Insight 2+ also provides a report system. This report system is a comprehensive explanation facility that gives the user complete access to the inferential process while running a knowledge base. With this system, the user can examine the state of all of the facts of the knowledge base, review the answer provided to Insight 2+ queries, see the rules of the knowledge base or trace the line of reasoning being pursued.

4.2 The Expert System

In the development of expert systems, the usual approach is to extract knowledge directly from an expert and incorporate it into the expert system. This approach usually results in hours of interaction between the expert engineer and the expert. The approach taken to develop BEAMDES differs from the usual approach in that the knowledge is not taken directly from an expert. Instead, it is taken from text books written by experts. Although a few text books are used, most of the knowledge is taken from Nawy [25].

The knowledge here refers to the heuristics which experts use for reinforced concrete beam design. These heuristics control the design procedure, beam size estimation, and reinforcement selection. Using these heuristics will result in adequate and economical designs which will also meet all the requirements specified by the ACI 318-83 Building code [3]. The heuristics are as follows:-

1. Beam sizes

   - The web width of the beam is taken approximately equal to half of the effective depth of the beam.

   - For a tee section, the span/effective depth ratio should be between 12 and 18.

- Overall dimensions are in whole inches and should be in multiples of 2 or 3.

2. Reinforcing bars

- Bar symmetry must be maintained about the centroidal axis which lies at right angles to the bending axis (i.e., the vertical axis).

- At least two and no more than six bars are to be placed in a layer of reinforcement.

- Use no more than two bar sizes and no more than two standard sizes apart for steel in one face at a given location in the span.

- Use bars # 11 and smaller for usual size beams.

- No more than two layers of reinforcement are placed at one face of the beam.

3. Design procedures

- For a continuous beam, section at the positive moment region (i.e., near midspan) is designed first, before the section at negative moment regions (supports) is designed.

- Supports of continuous beams are designed as doubly reinforced sections with approximately half of the midspan bars carried through the supports as compression reinforcement.

In this expert system, the above heuristics and other requirements for flexure design of reinforced concrete beams are incorporated into a knowledge base named BEAMDES and nine computational routines. Although the computational routines are employed specifically to perform computation procedures, some of them also contain rules, especially those for the selection of beam sizes and reinforcements. The rest of this chapter discusses the knowledge base and the computational routines. For a listing of the knowledge base and computational routines, refer to Appendix C. Figure 3 on page 40 shows a schematic view of the expert system.

4.2.1 Knowledge Base: BEAMDES

The knowledge base contains the heuristic mentioned earlier. With these rules and with the help of the computational routines, BEAMDES is able to achieve an economical design satisfying requirements specified by the ACI building code. BEAMDES will need some initial information about the beam. This information is usually what an expert would be given when asked to design a beam. With this information, BEAMDES is able to analyze the beam and recommend an adequate section (i.e., the beam dimensions and reinforcement) to the user. The rest of this Section describes the structure and rules in the knowledge base. Figure 2 shows the structure of the knowledge base.

As this knowledge base was developed using Insight 2+, it uses the PRL type of knowledge representation scheme. The knowledge base used in BEAMDES is represented using the production rule language. Insight 2+ uses a backward chaining inference engine to monitor the execution of the knowledge base. There are currently 76 rules in the knowledge base of BEAMDES. Some of these rules are procedural rules which are provided to perform specific tasks such as to execute an external program or to request the user to enter certain parameters. The rules contain in the system can basically be divided into two major categories. The first category of rules is generally used to obtain information from the user and for analysis. The second category of rules relate to the detailed design which involves the selection of the beam section and reinforcement.

The first category consists of 27 rules in which all the needed information about the beam is obtain from the user or determined by BEAMDES itself. The design parameters required are the concrete and steel yield strength, slab thickness, span length, support width and beams spacing. There are two ways in which BEAMDES determines the factored moments depending on which options the user has chosen. In the first option, the user analyzes the beam and furnishs BEAMDES with the maximum positive and negative factored moments at the span. Those moments should include all super-imposed dead and live loads including the weight of the slab. Moments due to self-weight are added by BEAMDES after a section is estimated.

In the second option, BEAMDES performs the analysis. To do this, it needs to know the boundary conditions and loads. The user is asked to furnish the boundary conditions. To determine the live loading, nine rules are provided which relate the function or occupation (provided by user) of the structure to the intensity of the uniform live loads (Ref. [4], pp. 12-13). Those live loads range from 40 psf to 250 psf depending on where the beam is located. With the loading, BEAMDES can determine the maximum positive and negative factored moments due to live load and weight of the slab, with the help of the computational routine MOMENT. The super-imposed dead load other than the weight of the slab is not accounted for in this option. Therefore, if there is a significant amount of super-imposed dead load then it is advisable to choose the first option.

The first part of the knowledge base also consists of guidelines for the determination of minimum thickness (h) requires for deflection control (ACI 318-83, Table 9.5(a)) and cover required for reinforcement protection (ACI 318-83, Section 7.7.1). Four rules are provided which relate the boundary conditions to minimum thickness while three rules are provided to relate the exposure of the beam to cover requirements.

Insight 2+ is a goal driven expert system shell, therefore, when building an Insight 2+ knowledge base one has to specify a goal. In the case of BEAMDES,

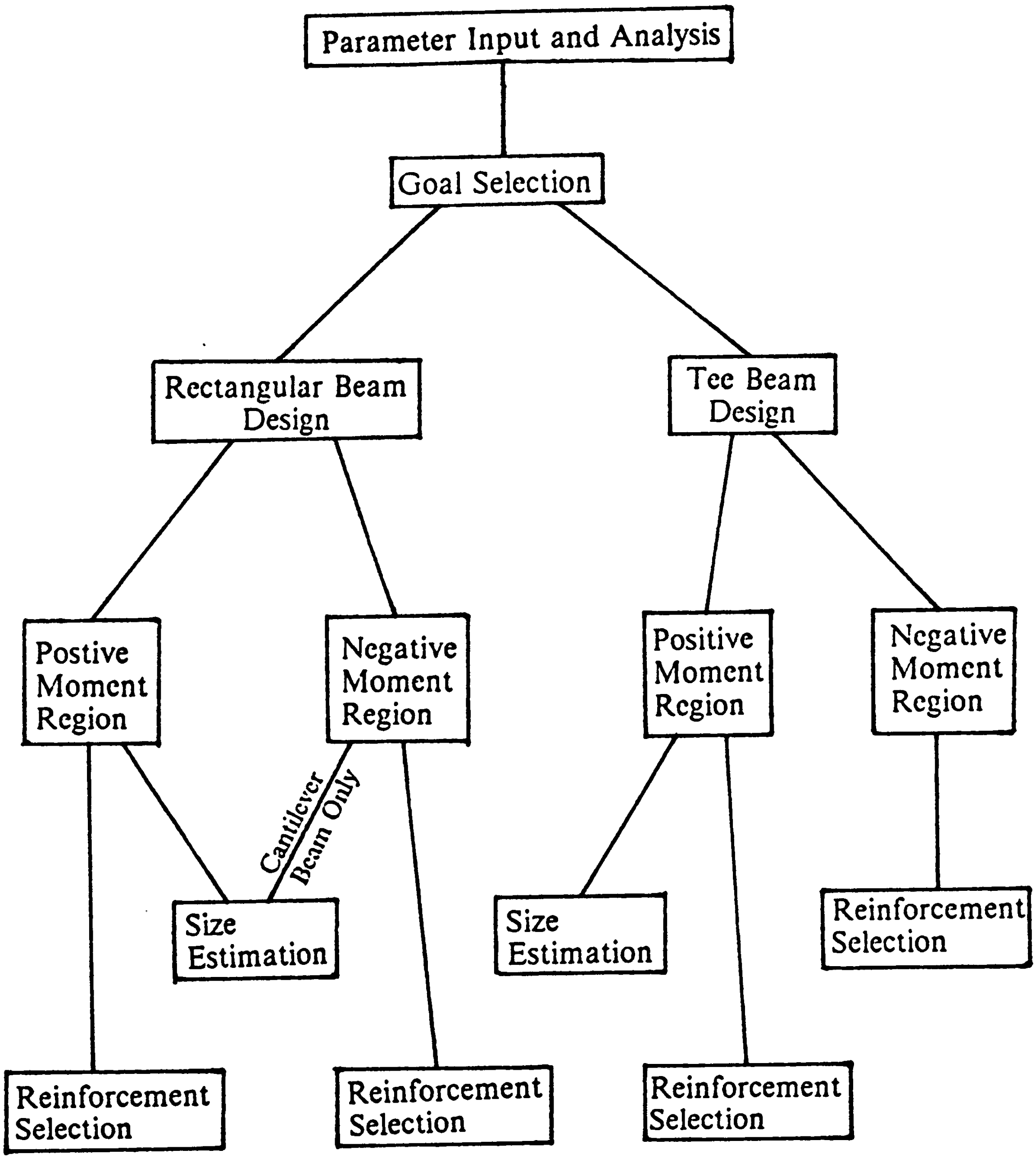


Figure 2. Structure of The Knowledge Base.

the goal is either to design a rectangular or a tee beam. A goal menu is provided for the user to choose a goal if he knows which one to choose. If not, four rules are provided so that BEAMDES will be able to determine the goal by itself. If the beam is a cantilever then it is designed as a rectangular section. For other boundary conditions, if the slab is reinforced concrete and cast monolithically with the beam then the beam is designed as a tee beam, otherwise, a rectangular beam is designed.

The rules for the detailed design can be divided into two parts, i.e., those that relate to the design of a rectangular beam and those that relate to the design of a tee beam. The rules can be further subdivided into rules for the design of the positive moment region (near midspan) and rules for the design of the negative moment region (at supports). An example of a rule for the design of a rectangular section for the positive moment region is as follows:

```
RULE Rectangular positive moment
IF   we have estimated rectangular dimensions
AND  we have self weight
AND  we have self weight moments
AND  we have selected positive reinforcement
THEN we have designed for rectangular positive moment
```

As can be seen, the rule contains several conditions. If all the conditions are true then the goal of a satisfactory design is satisfied. When this rule is activated, BEAMDES will first estimate a rectangular section using routine RECTDIM and then reanalyze the beam for the self-weight moments. These moments will be added to the moments determined earlier for reinforcement selection.

There are ten rules for estimating a section. Those rules will estimate a rectangular or tee section depending on which one is required and then allow the user to change it if desired. There are 17 rules for the reinforcement selection. These rules will result in a suitable set of reinforcement for the positive moment region, negative moment region and for cantilever beams. These rules are able to recommend suitable reinforcement, enlarge the section if necessary, and check the adequacy of the reinforcement chosen by the user if he decides to change the one recommended by BEAMDES. A typical rule for acceptable steel reinforcement is:

RULE Selection of reinforcement
IF we have reinforcement
AND NOT section is to be enlarged
AND NOT user want to change reinforcement
THEN we have selected reinforcement.

The idea is that if we have an acceptable arrangement of reinforcement, and the section does not need to be enlarged, and the user does not want to change it then the reinforcement is adequate. Beside all of these rules, the knowledge base also contains text for questions, information displays and explanatory notes. Whenever a question is asked or the user is requested to enter a parameter, text is provided so as to make the question more understandable. If the user is not sure how to respond to a question, he can look at an explanatory note that is provided for that question by requesting it. This explanatory note provides a more detailed explanation to the question. Information displays are provided to display messages, results and any information to the user at different stages of the execution of the program. This allows the user to know what is going on in the execution and provides a user-friendly interface.

4.2.2 Computational Routines

As mentioned earlier, it is more convenient to perform computations using external routines written in a procedural language rather than using Insight 2+. From this point of view, nine external computational routines, written in Microsoft QuickBasic, are provided to assist BEAMDES to perform computations. Each of these nine routines perform a specific computation or task which will be discribed in the following sub-sections. When necessary, Insight 2+ will call and activate the designated routine.

Parameters are passed between the knowledge base and the routines through a disk file named DATA.DAT. When Insight 2+ activates a routine, the parameters that need to be passed to the routine are written on the disk file DATA.DAT. The routine when activated will first read all the parameters in DATA.DAT before proceeding to perform the computations. After activation is completed, the procedure is reversed so that the results of the computations are passed to the knowledge base. Each time parameters are written to DATA.DAT, the parameters previously written will be erased completely and replaced by the new values.

This allows the same file to be used every time information is passed between the knowledge base and external routines.

4.2.2.1 Routine: SELFWEHT

This routine assists BEAMDES to compute the self-weight of the beam section. For a Tee section, only the weight of the web is computed because the weight of the flange is included in the weight of the slab computed in routine MOMENT.

4.2.2.2 Routine: MOMENT

This routine assists BEAMDES in the computation of the factored maximum positive and negative moments in a beam. It computes the moment due to self-weight of the beam, or due to live and dead load separately, according to BEAMDES's request. The live and dead load moments only consist of moments due to slab weight and live load. The factored total load is computed according to ACI 318-83, Section 9.2.1. The method used to compute moments is the approximate method of ACI 318-83, Section 8.3. Routine MOMENT returns only the maximum positive and negative moments to BEAMDES.

4.2.2.3 Routine: RECTDIM

This routine assists BEAMDES in the selection of a rectangular section for maximum positive moment. It uses the method described in Section 3.1. First, it estimates a section for the moment due to live and dead load only. After obtaining this estimated section, it calculates the self-weight and the moment due to self-weight. It then selects an adequate section to resist the combined moment due to self-weight, dead load and live load. It returns this section to BEAMDES.

4.2.2.4 Routine: TEEDIM

This routine selects a tee section for the maximum positive moment region. It uses the method described in Section 3.2. In this method, a rectangular beam section is first estimated using procedures similar to those in RECTDIM. The web width and effective depth is adjusted so as to get a beam section with a span to depth ratio between 12 and 18 (see Nawy [25], Section 5.10, pp. 126). The flange width is calculated according to ACI 318-83, Section 8.10.2. The thickness of the flange (slab) forms part of the depth of the beam.

4.2.2.5 Routine: POSREIN

This routine assists BEAMDES in the selection of reinforcement for the positive moment region i.e., near midspan. It follows either the procedure described in Section 3.1 or 3.2 depending on whether the section is a rectangular or tee section. After estimating the area of reinforcement required, it selects the bar sizes. For each bar size, it calculates the number of bars required and then calls a subroutine to check if this bar size is adequate. It performs checks to insure that the reinforcement can fit into the beam in one or two layers, that the reinforcement percentage is within specified and that the capacity of the beam is adequate. If this bar size is adequate, it stores the information obtained in an array. It then proceeds to the next bar size. After checking all available bar sizes, it selects the bar size that result in the least area of reinforcement.

4.2.2.6 Routine: NEGREIN

This routine assists BEAMDES in the selection of suitable reinforcement for the negative moment region i.e., at the supports. As the supports are doubly reinforced, it also determines how many of the midspan bars are to be carried through the supports as compression reinforcement. It follows the method described in Section 3.3 and the procedures for selecting a suitable bar size are similar to those in routine POSREIN.

4.2.2.7 Routine: ENLARGE

Whenever necessary BEAMDES calls upon ENLARGE to increase the size of a section. This is done by first increasing the web width of the section by one or two inches depending on which results in a dimension that is a multiple of 2 or 3. The depth is increased proportionally to the web width. If the section is tee then a new flange width is calculated.

4.2.2.8 Routine: BARCHAN

If the user desires to change the recommended reinforcement, BEAMDES will call upon this routine to allow the user to enter the reinforcement desired. This routine will display the current selected reinforcement, the required area of reinforcement and a table of bar sizes, bar diameters and areas of bars. The user is allowed to enter a combination of two bar sizes.

4.2.2.9 Routine: BARCHECK

After the user has changed the recommended reinforcement, BEAMDES will check if the reinforcement the user selected meets code requirements. Rules are provided to check the reinforcement against the following (see Ref. [30] pp. 59-62.):

1. The number of bars must be at least 2 and less than 12.

2. The difference in bar sizes should not be more than 2 standard sizes apart.

3. The bar must be placed in one or two layers with adequate spacing.

4. The reinforcement ratio (percentage) must be within the limits specified by ACI 318-83.

5. The capacity provided by the section must be sufficient to resist the total factored moment.

If all the above conditions are true then the reinforcement will be accepted.

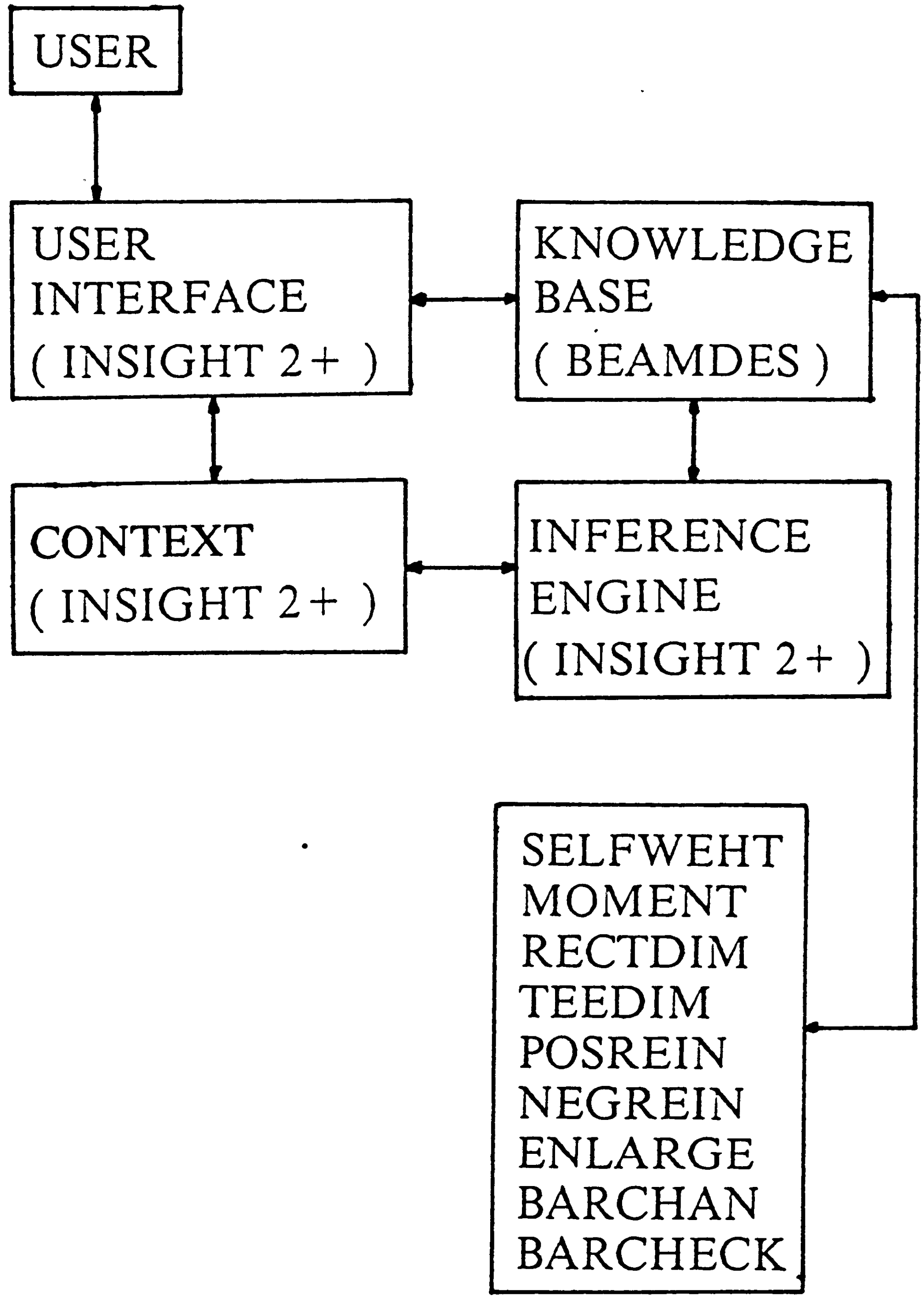


Figure 3. Schematic View of the Expert System.

Chapter V. User Guide For BEAMDES

5.1 Starting BEAMDES

As BEAMDES was developed using Insight 2+, it can only be run from Insight 2+. See Ref. [15] for system requirements and how to load and start Insight 2+ for hard-disk system. For a dual floppy disk system, boot the system using DOS version 2.0 or higher. After this, insert the Insight 2+ disk A and disk B into the A drive and B drive respectively. At the A prompt, type I2. A brief sign-on message will appear on the screen. After a few moments, the menu will be presented. At this point, remove Insight 2+ disk A from the A drive and replace it with the BEAMDES disk. Do not remove the Insight 2+ disk B from drive B. In order to run BEAMDES without any run-time errors, the BEAMDES disk must contain the following files:

- BEAMDES.KNB
- SELFWEHT.EXE

- MOMENT.EXE
- RECTDIM.EXE
- TEEDIM.EXE
- POSREIN.EXE
- NEGREIN.EXE
- ENLARGE.EXE
- BARCHAN.EXE
- BARCHECK.EXE
- BRUN40.EXE
- DATA.DAT (optional)

The BRUN40.EXE file is a Quickbasic run-time module file. This file is required to run all the *.EXE files as those files are written in Quickbasic. The DATA.DAT file is used to pass data between BEAMDES and the computational routines. This file will be created by the system when it is required.

5.2 Running BEAMDES

This section discusses in detail the features of BEAMDES and the steps involved in running BEAMDES. After replacing the Insight 2+ disk A with BEAMDES disk, select the " Run a knowledge base " option by using the direction keys and then press RETURN. Insight 2+ will show a directory listing of all compiled knowledge bases on the disk in drive A. In this case, there is only BEAMDES, therefore, just press RETURN to load BEAMDES.

After loading BEAMDES, the first page of the title appears on the screen. The title is two pages (screens) in length. Press F1, PAGE, to see next page or F3, STRT, to start a design session. When F3 is pressed, BEAMDES will ask the user to enter a name for this session. Enter a name of not more than 80 characters. After entering the name, the user will be prompted for information on the beam, i.e., the concrete compressive strength (f_c') , steel yield strength (f_y) , slab thickness (h_f) , span length, adjacent span lengths, width of support, and transverse spacing of the beams. If the user is not sure what value to use for those parameters, he can press the function key marked EXPL to see an explanatory display. A value greater than zero must be entered for concrete and steel yield strength and span length, otherwise, the beam would not be designed. For the slab thickness, width of support and spacing of beams, enter zero if those values are not available. Note that all numerical values throughout BEAMDES must be entered in inches and kips. If a wrong value is entered and the user has already pressed RETURN, the user can restart the session again any time by pressing the function key marked STRT.

A list of boundary conditions will appear on the screen. Use the up-down direction keys to select a boundary condition that is best suited for the beam being designed and press RETURN. After this, the user will be asked whether the factored moments are known. Answer appropriately using the up-down direction keys and press RETURN. If the factored moments are known, the user will be asked to enter the maximum positive and negative moments depending on which one is needed, otherwise, a list of occupancy will be shown on the screen. This list is three pages in length, press F1, PAGE, to advance to the next page. Select an occupancy that best describes the beam location and the function of the structure. BEAMDES will then determine the uniform design live load (see Ref. [4] pp. 12-13) for the beam. BEAMDES will then execute the routine MOMENT to compute the maximum moments due to this load. After this, the minimum thickness (h) is determined for deflection requirements (ACI 318-83 Table 9.5(a)). This is done internally and not shown to the user. The next display lists exposure conditions. Select the one that is most suitable for the beam and press RETURN. This will allow BEAMDES to determine the cover required for reinforcement protection (ACI 318-83 Section 7.7.1).

At this point, BEAMDES is ready to design the beam and the goal selection menu is displayed. The goal selection menu is shown below.

Can you identify the area of interest ?

= = > Design for RECTANGULAR beam

Design for Tee beam

As you can see the goal in BEAMDES is either to design a rectangular or Tee beam. If you know which kind of beam (rectangular or Tee) to design for then select one and press RETURN, otherwise, press F2, UNKN. If F2, UNKN, is pressed, BEAMDES will have to decide for itself which beam to design. To do this, it needs to know if the slab is reinforced concrete and if the beam is cast monolithically with slab. Answer " TRUE" to the next two questions if the slab is made of reinforced concrete and cast monolithically with the beam, otherwise, answer " FALSE ". If the first question is false then the second question will not be asked. On the other hand, if the beam is cantilever then the user will not be asked the above two questions as it will be able to determine that a cantilever beam is always designed as a rectangular beam. After deciding which beam to design, a message is displayed and the user is asked to press F2, CONT, to go to the goal selection menu and select the one BEAMDES has recommended.

The next display shown on the screen is a message informing the user at which region (positive or negative moment) of the beam will be designed next. If the beam is simply supported or continuous then a messsage saying that positive moment region is to be designed will be displayed, otherwise, the message will say that negative moment is to be designed. Press F2, CONT, to continue.

When F2, CONT, is pressed, BEAMDES will start the design process. First, it will estimate a beam size by executing the routine RECTDIM or TEEDIM, depending on whether a rectangular or tee section is selected at the goal selection menu. This estimated section will be displayed next. A feature of BEAMDES is that it allows the user to change the estimated section since, in some situations, it may be necessary to limit the beam depth for architectural or some other reasons. This provides some flexibility in the selection of beam sizes. If another section is to be used then answer "TRUE " to the next question, otherwise, answer " FALSE ". If the answer is " TRUE ", then the user will be prompted for the flange width, web width, and thickness of the section. For a rectangular section, the flange width is not needed. The new section will be redisplayed and the user will be asked again whether he wants to change it. At this point, if the user answer " FALSE ", BEAMDES will go ahead and compute the beam weight and the moments due to beam weight by executing SELFWEHT and MOMENT respectively.

The next task is the selection of steel reinforcement. This is accomplished by calling the routine POSREIN. If POSREIN cannot find suitable reinforcement for the section selected, obviously, the size is not adequate. If this is so, a message will be displayed and BEAMDES will execute the routine ENLARGE to obtain a larger section. The new section will be displayed on the next screen. POSREIN is again executed to select the reinforcement. This process is repeated until a beam section and reinforcement is found to be adequate. The final selected reinforcement is displayed on the screen.

After selecting the reinforcement, the user is allowed to change the selected reinforcement. This feature is provided because the bar sizes recommended by BEAMDES may not be available to the user or there may be other factors which requires the selection of different size bars. If " TRUE " is selected, BEAMDES will execute routine BARCHAN to allow the user to select a new set of reinforcement. BARCHAN will then display the current selected reinforcement. A bar-table and the required area of reinforcement are printed as a reference for the user. A combination of two bar sizes may be used. Enter the two bar numbers separated by a comma. If only one bar number is used, enter the bar number followed by a comma. The user is then asked to enter the number of bars for each sizes. Use the same format as for bar number but be sure that it is entered in the same sequence as the bar number. BEAMDES will again display the reinforcement entered and ask whether the user want to change it, in case a wrong value was entered. If the reinforcement is changed, BARCHECK will be executed to check whether the reinforcement entered is adequate. If it is found that the reinforcement is not adequate then a message will be displayed. Press F2, CONT to reenter another set of reinforcement. This process is repeated until the reinforcement selected is adequate.

At this stage, BEAMDES has completed designing the section of the beam. If the beam is simply supported or cantilever then the next two screens will show the input parameters and the beam section (results). If the beam is continuous, the section at the supports (i.e., the negative moment region) will be designed.

As the size of beam is known, only the reinforcement need to be selected. This is done using the routine NEGREIN. If maximum reinforcement is inadequate for the negative moment section then the size of the beam recommended for positive moment design is not adequate, and the beam size has to be increased. Since the size of the beam both for positive and negative moment regions has to be the same, an increase in size at the negative moment region simultaneously causes an increased in size at the positive moment region. Consequently, the reinforcement at the positive moment region can be reduced. In this case, BEAMDES will go back and redesign the section at the positive moment region using the enlarged section. The procedures for this are the same as described earlier. After redesigning the section at the positive moment region, BEAMDES will come back and design the section at the negative moment region. This procedure is repeated until reinforcement is selected for the negative moment region. Then the user is allowed to change the reinforcement, if desired, using the same procedures as described for the positive moment region. After designing the section at the negative moment region, BEAMDES has concluded a design session. The next three screens will show the input parameters, and the designed sections at positive and negative regions. At the third screen, press F3, STRT, to start another design session or press F2, CONT, to continue and look at the report menu. The report menu provides the user with a complete accounting of the line of reasoning. It also summarizes, in chronological order, the facts provided by the user and conclusions that were reached as a result.

5.3 Supplementary Functions

Insight 2+ is a user-friendly system and most of its functions can be accessed via function keys. Function keys that are active at a specific screen are highlighted at the bottom of the screen. A user can activate the functions corresponding to any of those function keys by pressing the appropriate function key. Some of the function keys applicable to BEAMDES are as follows:

BACK Returns you to the previous screen.

CONT Resumes knowledge base execution.

CURR Displays the currently active rule being pursued.

EXIT Gives you the opportunity to leave Insight 2+.

EXPL Activates a display of explanatory information provided by the knowledge engineer.

HELP Gives you helpful information.

MENU Returns you to the main menu.

OFFK Toggles the highlighted display of function key designators on/off allowing you to use the print screen (PrtSc) key to send any screen of text to your printer without printing the row of function key designators. The keys remain active.

PAGE Advances to the next screen or " page " of information.

PRNT Allows you to send textual displays to your printer.

RULE Activates the Rules Menu of the Report system.

STRT Starts the knowledge base currently in memory.

UNKN Indicates that you are unable to answer the question.

WHY? Entry point to the Report system.

YES Indicates that you wish to execute the action Insight 2+ is starting. This will either be to exit or restart a knowledge base.

Those keys can only be activated when they are highlighted. For an explanation of other keys, see Section 1.5.1 of the Insight 2+ reference manual [15]. Appendix D of this reference manual provides a description of all the Insight 2+ messages that are displayed during run time.

Chapter VI. Limitations, Future Improvements and Conclusions

6.1 Limitations

BEAMDES, as mentioned earlier, is limited to the flexural design of reinforced concrete beams with simply supported, continuous or cantilevered spans. The beam section may be rectangular or tee section. BEAMDES can only design one span of a continuous beam at a time, i.e., the exterior or interior span. For an interior span, BEAMDES assumes that the moments at both ends of the beam are approximately the same, therefore, it only designs for the larger of the two moments.

At present, if there is a significant amount of super-imposed dead load, it is not accounted for by BEAMDES in its load determination. Also, the method of analysis does not account for concentrated loads. In order to take those loadings into consideration, the user has to analyze the beam and supply BEAMDES with the maximum positive and negative moments.

The program is limited in flexibility as regards to design for a singly or a doubly reinforced section. BEAMDES will design a singly reinforced section for the positive moment region and a doubly reinforced section at the negative moment region. This may not be convenient if the size of a beam is restricted. For example, if the beam size is limited and it is found that a singly reinforced section would not work, BEAMDES will not design the beam as a doubly reinforced section. Instead, it would choose to increase the size of the beam.

6.2 Future Improvements

BEAMDES could be further improved by providing knowledge to account for the limitations mentioned in the previous section. That is, to design for both ends of an interior span of a continuous beam if the moments at both ends differ by a significant amount. Rules should also be implemented to account for superimposed dead loads. A more sophisticated method of analysis could be employed so that BEAMDES would be able to analyze beams subjected to more complicated loadings. It would be better if BEAMDES could also change its design from a singly reinforced section to doubly reinforced section when necessary.

The design of a reinforced concrete beam is not complete without the design for shear and development of reinforcement. Therefore, to make it complete, rules for shear design and reinforcement development should be added. Another suggested improvement is to provide rules so that BEAMDES could determine what concrete and steel yield strength to use, instead of requesting this information from the user.

Some other features for reinforced concrete beam design such as beam-column compatibility and formwork construction consideration should be included into the system. ACI 318-83, Section 10.6.6 requirement for distribution of flexural reinforcement in tee beams should also be included.

6.3 Conclusions

Over the last few years, interest in expert system applications has increased quite significantly among Civil Engineering researchers. Many applications to Civil Engineering problems have been look at and studied. These studies indicates that there is great potential for the application of expert systems in Civil Engineering. It is anticipated that in the next few years, there will be many commercial versions of expert system in the construction industry.

BEAMDES's knowledge is taken mainly from a reinforced concrete text book [25] written by a human expert. Therefore, BEAMDES tries to imitate as close as possible the human expert's method of designing reinforced concrete beams. The heuristics used for reinforced concrete beams design will be the same for BEAMDES and this expert. But it is not necessary that it would be the same as other experts. Those other experts may have their own way of designing beams and may use different approaches. In Appendix A, the results given by BEAMDES are compared with those obtained by two human experts. From this, we can see that BEAMDES's recommendations are identical to one expert's recommendations and differ only slightly with the second expert's recommendations.

When tested against other experts, it is possible that BEAMDES may sometimes recommend sections quite different from those of the experts. But when checked, the sections recommended by both the experts and BEAMDES will work. It is a matter of choice that cause the recommendations to be different, because different experts use different methods to design reinforced concrete beams.

The performance of BEAMDES shows that it is successful in capturing the knowledge of a human expert. It also shows that there is great potential for the application of expert systems to reinforced concrete design.

References

1. Adeli, H., " Knowledge-Based Expert Systems in Structural Engineering ", *Engineering Analysis,* Vol. 3, Prentice-Hall, Sept. 1986, pp. 154-160.

2. Alam, J., Stanton, T.R., and Ritter, J.F., " Knowledge Based Expert System Truss Advisor ", *Proceedings of the 4th National Conference on Microcomputers in Civil Engineering,* Orlando, Florida, Nov. 1986, pp. 320-322.

3. American Concrete Institute, " Building Code Requirements for Reinforced Concrete ", ACI 318-83, Detroit, Michigan, 1983.

4. American National Standard Institute, " Minimum Design Loads For Buildings And Other Structures ", ANSI A58.1-1981, New York, N.Y., 1981.

5. Barr, A., and Feigenbaum, E.A., *A Handbook of Artificial Intelligence,* Vol. 2, Addison Wesley, London, 1986.

6. Bennett, J.S., and Engelmore, R.S., " SACON: A Knowledge-Based Consultant for Structural Analysis ", *Proceedings of the 6th International Joint Conference on Artificial Intelligence,* Tokyo, Japan, 1979, pp. 47-49.

7. Clifton, J.R., and Oltikar B.C., " Expert System for Selecting Concrete Constituents ", *Computer Applications in Concrete Technology,* SP-98, American Concrete Institute, Detroit, Michigan, 1987, pp. 1-24.

8. Duda, R., Gaschnig, J., and Hart, P., " Model Design in the PROSPECTOR Consultant System for Mineral Exploitation ", Edited by Michie, D., *Expert Systems in the Micro Electronic Age,* Edinburgh University Press, 1979, pp. 153-167.

9. Finn, G.A., and Reinschmidt, K.F., " Expert Systems in an Engineering-Construction Firm ", Edited by Kostem, C.N., and Maher, M.L., *Expert Systems in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 40-54.

10. Gashnig, J., " Preliminary Performance Analysis of the PROSPECTOR Consultant System for Mineral Exploration ", *Proceedings of the 6th International Joint Conference on Artificial Intelligence,* Tokyo, Japan, 1979, pp. 308-310.

11. Godfrey, K.A. Jr., " Expert Systems Enter the Marketplace ", *Civil Engineering,* ASCE, May 1986, pp. 70-73.

12. Harris, L.R., and Dwight, B.D., *Artificial Intelligence Enters the Marketplace,* Bantam Computer Books, June 1986.

13. Hayes, P.J., " On Semantic Nets, Frames and Associations ", *Proceedings of the 5th International Joint Conference on Artificial Interlligence,* MIT, Cambridge, Massachusetts, 1977, pp. 99-107.

14. Hayes-Roth, F., Waterman, D., and Lenat, D., *Building Expert Systems,* Addison Wesley, London, 1983.

15. *Insight 2+ Reference Manual,* Level Five Research, Inc., Indialantic, Florida.

16. Jennings, A., " AI Expert Systems for Automated Review of Hazardous Waste Facility Permit Applications ", *Proceedings of the 4th National Conference on Microcomputers in Civil Engineering,* Orlando, Florida, Nov. 1986, pp. 110-115.

17. Kostem, C.N., " Design of an Expert System for the Rating of Multibeam Highway Bridges ", Edited by Kostem, C.N., and Maher, M.L., *Expert Systems in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 87-95.

18. Kostem, C.N., " Attributes and Characteristics of Expert Systems ", Edited by Kostem, C.N., and Maher, M.L., *Expert Systems in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 30-39.

19. Krauthammer, T., and Kohler, S., " RC Structures under Severe Loads - An Expert System Approach ", Edited by Kostem, C.N., and Maher, M.L., *Expert System in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 96-108.

20. Lakmazaheri, S., " Slab Synthesis By A Microcomputer-Based Expert System ", *Proceedings of the 4th National Conference on Microcomputers in Civil Engineering,* Orlando, Florida, Nov. 1986, pp. 323-327.

21. Levitt, R.E., " HOWSAFE: A Microcomputer-Based Expert System to Evaluate the Safety of a Construction Firm ", Edited by Kostem, C.N., and Maher, M.L., *Expert Systems in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 55-66.

22. Maher, M.L., " Problem Solving Using Expert Systems Techniques ", Edited by Kostem, C.N., and Maher, M.L., *Expert System in Civil Engineering,* ASCE, New York, N.Y., 1986, pp. 7-17.

23. Malpas, J., *PROLOG: A Relational Language and Its Applications,* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1987.

24. McCarthy, J., " Recursive Functions of Symbolic Expressions and Their Computation by Machine ", *Communications of the ACM,* Vol. 3, No. 4, 1960, pp. 184-195.

25. Nawy, E.G., *Reinforced Concrete: A Fundamental Approach,* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1985.

26. Rosenberg, S., " Frames-Based Text Processing ", *Proceeding of the 5th International Joint Conference on Artificial Intelligence,* MIT, Cambridge, Massachusetts, 1977, pp. 108-109.

27. Sell, P.S., *Expert Systems - A Practical Introduction,* John Wiley & Sons, 1986.

28. Spiegel, L., and Limbrunner, G.F., *Reinforced Concrete Design,* Second Edition, Prentice-Hall, Englewood Cliffs, New Jersey, 1986.

29. Sriram, D., Maher, M.L., and Fenves, S.J., "Knowledge-Based Expert Systems in Structural Design ", *Computers & Structures,* Vol. 20, No. 1-3, 1985, pp. 1-9.

30. Wang, C.K., and Salmon, C.G., *Reinforced Concrete Design,* Fourth Edition, Harper & Row, Publishers, New York, 1985.

31. Weigel, T.A., and Fenske, T.E., " A Demonstration Prototype of An Expert System for Reinforced Concrete Column Design ", *Proceedings of the 4th National Conference on Microcomputers in Civil Engineering,* Orlando, Florida, Nov. 1986, pp. 116-118.

32. Winter, G., and Nilson, A.H., *Design of Concrete Structures,* Ninth Edition, McGraw-Hill Book Company.

Appendix A. Tests Against Human Experts.

In order to evaluate the performance of an expert system, it is necessary to test it against human experts. BEAMDES, was tested against several example design problems which are found in text books written by human experts. Some of the results of these tests are given below:

Test 1: Nawy [25], 5.5.1, Example 5.3

In this example, a reinforced concrete simply supported beam has a span of 30 ft and is subjected to a service uniform load w_u = 1500 lb/ft. A rectangular beam section was designed to resist the factored external bending load. Given: f_c' = 4 ksi, f_y = 60 ksi.

A maximum positive moment of 3442.5 in-kips (simply supported moment due to w_u) is entered. Zero values were entered for the slab thickness, support width and beam spacing. The beam was chosen as not exposed to earth or weather so as to give a cover of 1.5 inches which is the same value used by Nawy. With these parameters BEAMDES recommended a section which is similar to that of Nawy's, b = 12 in, d = 23 in, h = 26 in and A_s = 3.81 in^2 (3 # 10's).

Test 2: Nawy [25], 5.10.2, Example 5.9

An interior beam having a clear span of 25 ft and carrying a working live load of 8000 lb/ft in addition to its self-weight was designed. The beam is assumed to have a 4-in slab cast monolithically with it. Given: f_c' = 4 ksi, f_y = 60 ksi.

A maximum positive moment of 6375 in-kips and a maximum negative moment of 9272.73 in-kips were entered for BEAMDES. Support width and beams spacing were entered as zero as those values were not known. The beam was chosen as not exposed to earth or weather. As this is an interior span, boundary condition of continuous at both ends was chosen. The sections, both at positive moment region (midspan) and negative moment region (supports), recommended by BEAMDES are similar to those given by Nawy. At midspan section, a Tee beam was designed. The recommended section is b = 75 in, b_w = 14 in, d = 25 in, h = 28 in, and A_s = 5.08 in^2 (4 # 10's). At the supports, a doubly reinforced rectangular beam was designed. The recommended section is b_w = 14 in, d = 23.9 in, h = 28 in, A_s' = 2.54 in^2 (2 # 10's) and A_s = 8.89 in^2 (7 # 10's placed in two layers).

Test 3: Spiegel [28], 2.14, Example 2.8

Design a simply supported rectangular reinforced concrete beam with tension steel only, to carry a service dead load of 0.9 kip/ft and a service live load of 2.0 kip/ft (the dead load does not include the weight of the beam). The span is 18 ft. Use $f_c' = 4$ ksi and $f_y = 60$ ksi. The section recommended by Spiegel is $b = 11$ in, $d = 20.06$ in, $h = 22.5$ in, and $A_s = 3.00$ in^2 (3 # 9's).

For BEAMDES, a value of 2268 in-kips was entered for the maximum positive moment. Support width, slab thickness and beam spacing were zero. The beam is not exposed to earth or weather. The section recommended by BEAMDES is $b = 10$ in, $d = 20.0$ in, $h = 23$ in, and $A_s = 2.54$ in^2 (2 # 10's).

The sections recommended by both Spiegel and BEAMDES are almost similar. BEAMDES is less conservative in choosing the reinforcement.

Test 4: Spiegel [28], 3.5, Example 3.4

A Tee-beam for a floor system was designed. The floor has a 4 in slab supported by 22 ft-span-length beams cast monolithically with the slab. Beams are 8 ft on center and have a web width of 12 in and a total depth of 22 in; $f_c' = 3$ ksi, $f_y = 60$ ksi. Service loads are 0.125 ksf live load and 0.200 ksf dead load. The given dead load does not include the weight of the floor system. The beam is simply supported. With b_w = 12 in, and h = 22 in, Spiegel recommended a flange width b of 66 in and A_s = 3.81 in^2 (3 # 10's).

The maximum positive moment was entered as 3492 in-kips with support width as zero. The beam is not exposed to earth or weather. First, BEAMDES recommended a beam with b = 66 in, b_w = 12 in, and h = 25 in. But the depth h is restricted to 22 in, therefore, the h recommended by BEAMDES was changed to 22 in. With this, it recommended a steel area, A_s = 3.81 in^2 (3 # 10's) which is similar to that recommended by Spiegel.

Appendix B. List of Symbols

a	Depth of rectangular stress block as defined in ACI [3], Section 10.2.7., in.
A_s	Area of tension reinforcement, sq. in.
A_{sf}	A compressive steel area which results in a force capacity same as that in the compression flange overhang, sq. in.
A_{s1}	The difference in area between the tension and compressive reinforcement of a doubly reinforced section, sq. in.
A_s'	Area of compression reinforcement, sq. in.
b	Width of compression face of member or effective compressive flange width of a structural member, in.
b_w	Web width of member, in.
c	Distance from extreme compression fiber to neutral axis, in.
d	Distance from extreme compression fiber to centroid of tension reinforcement or effective depth of section, in.
d'	Distance of extreme compression fiber to centroid of compression reinforcement, in.
f_c'	Specified compressive strength of concrete, ksi.
f_s'	Tensile stress in compression reinforcement, ksi.

f_y	Specified yield strength of reinforcement, ksi.
h	Overall thickness of member, in.
h_f	Flange thickness, in.
j	Ratio of moment arm of internal couple to effective depth d for an ideally reinforced beam.
L	Length of span in the direction in which moments are determined, measured center to center of supports, in.
L_1	Length of span (left side) transverse to L, measured center to center of supports, in.
L_2	Length of span (right side) transverse to L, measured center to center of supports, in.
M	Design moment, in-kip.
M_n	Nominal moment strength at section, in-kip.
M_{n1}	Nominal moment strength due to A_{s1} at section, in-kip.
M_{n2}	Nominal moment strength due to A_{sf} or A_s' at section, in-kip.
M_u	Design moment strength at section, in-kip.
r	Ratio of width of compression face to effective depth of a section.
R	A strength factor relating nominal strength to b and d at section.

Appendix C. Program Listings

C.1 Knowledge Base: BEAMDES.PRL

TITLE Expert System for Reinforced Concrete Beam Design DISPLAY

```
REINFORCED CONCRETE BEAM DESIGN                    Page 1 of 2

            Developed by
            Thien Pin Wong
              for his
          Master of Science
             Thesis at
   The Department of Civil Engineering
 Virginia Polytechnic Institute & State University
  Under the direction of Professor K.B. Rojiani
  Press F1 to Advance to next page.

  Press F3 to Start
```

REINFORCED CONCRETE BEAM DESIGN   Page 2 of 2

This knowledge base is developed for the flexure design of Reinforced Concrete beams. The beams can be simply supported, continuous or cantilevered of both rectangular and Tee sections.

Answer any questions as best as you can.

If you do not understand any questions, press the EXPL key to see an explanatory note.
If no explanatory note is provided for that particular question, refer to the user guide for explanation.

Only the function keys that are highlighted at the bottom of the screen are active at that particular screen.

Press F3 to Start

```
!-------------------------------------------------------------------------
!   Variables description
!-------------------------------------------------------------------------
!
!   Bar1,Bar2 - Number of bar of size one and two respectively
!   Size1,Size2 - Bar size one and two respectively
!   As provided - Area of tension steel provided
!   Bot1,Bot2 - Number of bottom layer bar of size one and two respectively
!   Top1,Top2 - Number of top layer bar of size one and two respectively
!   Asp providedN - Area of compression steel provided
!   Barp1N,Barp2N - Number of compression bar of size one and two respectively
!
!   * Note : Variables ending with an "N" denotes values at negative moment
!         region
!
!-------------------------------------------------------------------------
!   Declaring some facts as specific type
!-------------------------------------------------------------------------
!
SIMPLEFACT discontinuous end unrestrained
AND        discontinuous end has a spandrel beam as support
STRING  Title of job
AND     Section
AND     Section1
AND     Section2
AND     Status
OBJECT  Beam boundary
AND     Beam
AND     Occupancy
NUMERIC Cover
AND     Minimum h
AND     Fc
AND     Fy
AND     Span length
```

```
AND    Right span length
AND    Left span length
AND    Support width
AND    Beam spacing
AND    Maximum positive moment
AND    Maximum negative moment
AND    Self weight
AND    Self weight negative moment
AND    Self weight positive moment
AND    Loading
AND    Boundary
AND    Slab thickness
AND    b
AND    bw
AND    d
AND    d prime
AND    h
AND    Positive moment capacity
AND    Negative moment capacity
AND    Bar1
AND    Bar2
AND    Size1
AND    Size2
AND    As provided
AND    As required
AND    Layer of steel
AND    Bot1
AND    Bot2
AND    Top1
AND    Top2
AND    Bar1N
AND    Bar2N
AND    Size1N
AND    Size2N
AND    As providedN
AND    As requiredN
AND    Revised positive d
AND    Revised negative d
AND    Layer of steelN
AND    Top1N
AND    Top2N
AND    Bot1N
AND    Bot2N
AND    Barp1N
AND    Barp2N
AND    Asp providedN
AND    Number of span

INIT Section = " "
AND  Self weight = 0
AND  Loading = 0
AND  Asp providedN = 0
AND  Section1 = "SINGLY"
AND  Section2 = "DOUBLE"
AND  Top2 = 0
AND  Top2N = 0
```

```
AND  Bot2 = 0
AND  Bot2N = 0
AND  Size2 = 0
AND  Size2N = 0
AND  Bar2 = 0
AND  Bar2N = 0
AND  d prime = 0
AND  Maximum positive moment = 0
AND  Maximum negative moment = 0
!
MULTI Design for
!
!  Turn on the goalselect to allow user to select shape of section
!
GOALSELECT ON
!
!  Goal selection menu

1.   The beam may be designed

1.1  Design for \RECTANGULAR beam
1.1.1 Perform the RECTANGULAR beam design

1.2  Design for \TEE beam
1.2.1 Perform the TEE beam design

!-------------------------------------------------------------------------
!  Gathering information from user
!-------------------------------------------------------------------------
!
RULE  For determining if we are ready
IF    Title of job < > ""
AND   we have design parameters
AND   we have determined boundary
AND   we have moments
AND   we have minimum h
AND   we have cover
THEN  The beam may be designed
ELSE  DISPLAY need info

RULE  For gathering design parameters
IF    Fc > 0
AND   Fy > 0
AND   Slab thickness > = 0
AND   Span length > 0
AND   Right span length > = 0
AND   Left span length > = 0
AND   Support width > = 0
AND   Beam spacing > = 0
THEN  we have design parameters

!-------------------------------------------------------------------------
!  Determine boundary condition
!-------------------------------------------------------------------------
!
RULE Boundary for simply supported
```

```
IF   Beam boundary IS simply supported
THEN we have determined boundary
AND  Boundary := 1

RULE Boundary for both ends continuous
IF   Beam boundary IS continuous at both ends
THEN we have determined boundary
AND  Boundary := 2

RULE Boundary for one end continuous
IF   Beam boundary IS continuous at one end
THEN we have determined boundary
AND  Boundary := 3

RULE Boundary for cantilever
IF   Beam boundary IS cantilever
THEN we have determined boundary
AND  Boundary := 4

!--------------------------------------------------------------------------
!  Get moments
!--------------------------------------------------------------------------
!
RULE For gathering moment if known and beam is simply supported
IF   Loads and moments ARE known
AND  Boundary = 1
AND  ASK Maximum positive moment
THEN we have moments

RULE For gathering moments if known and beam is continuous
IF   Loads and moments ARE known
AND  Boundary = 2
OR   Boundary = 3
AND  ASK Maximum positive moment
AND  ASK Maximum negative moment
THEN we have moments

RULE For gathering moment if known and beam is cantilever
IF   Loads and moments ARE known
AND  Boundary = 4
AND  ASK Maximum negative moment
THEN we have moments

RULE To calculate moments if not known
IF   Loads and moments ARE not known
AND  we have determined loading
AND  we have calculated moments
THEN we have moments

RULE For 40 psf loading
IF   Occupancy IS hospital wards and rooms
OR   Occupancy IS residential
OR   Occupancy IS classrooms
THEN we have determined loading
AND  Loading := 40
```

```
RULE For 50 psf loading
IF   Occupancy IS assembly areas with fixed seating
OR   Occupancy IS garages for private pleasure car storage
OR   Occupancy IS offices
THEN we have determined loading
AND  Loading := 50

RULE For 60 psf loading
IF   Occupancy IS hospital operating rooms and laboratory rooms
OR   Occupancy IS library reading rooms
THEN we have determined loading
AND  Loading := 60

RULE For 75 psf loading
IF   Occupancy IS light manufacturing
OR   Occupancy IS upper floors of retail stores
THEN we have determined loading
AND  Loading := 75

RULE For 80 psf loading
IF   Occupancy IS school corridors above first floor
THEN we have determined loading
AND  Loading := 80

RULE For 100 psf loading
IF   Occupancy IS assembly areas with movable seating
OR   Occupancy IS exit facilities
OR   Occupancy IS garages for general storage and repair
OR   Occupancy IS dance halls and ballrooms or restaurants
OR   Occupancy IS gynasium main floor
OR   Occupancy IS reviewing stand and grandstand
OR   Occupancy IS first floor of retail stores
THEN we have determined loading
AND  Loading := 100

RULE For 125 psf loading
IF   Occupancy IS assembly areas with stages and enclosed platforms
OR   Occupancy IS library stack rooms
OR   Occupancy IS heavy manufacturing
OR   Occupancy IS light storage
OR   Occupancy IS wholesale stores
THEN we have determined loading
AND Loading := 125

RULE For 150 psf loading
IF   Occupancy IS armories
THEN we have determined loading
AND  Loading := 150

RULE For 250 psf loading
IF   Occupancy IS sidewalks and driveways
OR   Occupancy IS heavy storage
THEN we have determined loading
AND  Loading := 250
```

```
RULE To calculate moments
IF   Boundary = 1
OR   Boundary = 2
OR   Boundary = 4
AND  ACTIVATE MOMENT.EXE
DISK     DATA.DAT
SEND     Boundary
SEND     Loading
SEND     Self weight
SEND     Beam spacing
SEND     Span length
SEND     Right span length
SEND     Left span length
SEND     Support width
SEND     Slab thickness
RETURN   Maximum positive moment
RETURN   Maximum negative moment
THEN we have calculated moments

RULE To calculate moment for one end continuous beam
IF   Boundary = 3
AND  we have asked for more beam information
AND  ACTIVATE MOMENT.EXE
DISK     DATA.DAT
SEND     Boundary
SEND     Loading
SEND     Self weight
SEND     Beam spacing
SEND     Span length
SEND     Right span length
SEND     Left span length
SEND     Support width
SEND     Slab thickness
SEND     discontinuous end unrestrained
SEND     discontinuous end has a spandrel beam as support
SEND     Number of span
RETURN   Maximum positive moment
RETURN   Maximum negative moment
THEN we have calculated moments

RULE To ask for more beam info
IF   discontinuous end unrestrained
OR   NOT discontinuous end unrestrained
AND  discontinuous end has a spandrel beam as support
OR   NOT discontinuous end has a spandrel beam as support
AND  ASK Number of span
THEN we have asked for more beam information

!------------------------------------------------------------------------
!   Calculate minimum h for deflection requirement
!------------------------------------------------------------------------
!
RULE Minimum h for simply supported
IF   Beam boundary IS simply supported
THEN we have minimum h
AND  Minimum h := Span length/16
```

```
RULE Minimum h for continuous at both ends
IF   Beam boundary IS continuous at both ends
THEN we have minimum h
AND  Minimum h := Span length/21

RULE Minimum h for continuous at one end
IF   Beam boundary IS continuous at one end
THEN we have minimum h
AND  Minimum h := Span length/18.5

RULE Minimum h for cantilever
IF   Beam boundary IS cantilever
THEN we have minimum h
AND  Minimum h := Span length/8

!------------------------------------------------------------------------
!  Determine cover requirement
!------------------------------------------------------------------------
!
RULE For three inches cover
IF   Beam IS cast against and permanently exposed to earth
THEN we have cover
AND  Cover := 3

RULE For two inches cover
IF   Beam IS exposed to earth or weather
THEN we have cover
AND  Cover := 2

RULE For one and a half inches cover
IF   Beam IS not exposed to earth or weather
THEN we have cover
AND  Cover := 1.5

!------------------------------------------------------------------------
!  Decide whether to design a rectangular or tee section if user
!  selects unknown during goal selection
!------------------------------------------------------------------------
!
RULE Cantilever rectangular
IF   Beam boundary IS cantilever
THEN Design for \RECTANGULAR beam
AND  Section := "RECTANGULAR"

RULE Criteria for tee section
IF   NOT Beam boundary IS cantilever
AND  Slab is reinforced concrete
AND  Beam is cast monolithically with slab
THEN Design for \TEE beam
AND  Section := "TEE"

RULE Criteria for rectangular section
IF   NOT Beam boundary IS cantilever
AND  NOT Slab is reinforced concrete
OR   NOT Beam is cast monolithically with slab
```

```
THEN Design for \RECTANGULAR beam
AND  Section := "RECTANGULAR"

RULE For displaying the selected goal
IF   Section <> ""
THEN Design for \dummy
AND  DISPLAY section to be
AND  FORGET Design for
AND  CYCLE

!-----------------------------------------------------------------------
!  Perform rectangular beam design
!-----------------------------------------------------------------------
!
RULE Rectangular simply supported beam
IF   Boundary = 1
AND  Display positive moment design message
AND  we have designed for rectangular positive moment
THEN Perform the RECTANGULAR beam design
AND  DISPLAY input parameters
AND  DISPLAY rectangular beam section


RULE Rectangular continuous beam
IF   Boundary = 2
OR   Boundary = 3
AND  Display positive moment design message
AND  we have designed for rectangular positive moment
AND  Display negative moment design message
AND  we have designed for negative moment
THEN Perform the RECTANGULAR beam design
AND  DISPLAY input parameters
AND  DISPLAY rectangular beam section
AND  DISPLAY negative beam section

RULE Rectangular cantilever beam
IF   Boundary = 4
AND  Display negative moment design message
AND  we have designed for cantilever negative moment
THEN Perform the RECTANGULAR beam design
AND  DISPLAY input parameters
AND  DISPLAY cantilever beam section

!-----------------------------------------------------------------------
!  Perform tee beam design
!-----------------------------------------------------------------------
!
RULE Tee simply supported beam
IF   Boundary = 1
AND  Display positive moment design message
AND  we have designed for tee positive moment
THEN Perform the TEE beam design
AND  DISPLAY input parameters
AND  DISPLAY tee beam section
```

```
RULE Tee continuous beam
IF  Boundary = 2
OR  Boundary = 3
AND  Display positive moment design message
AND  we have designed for tee positive moment
AND  Display negative moment design message
AND  we have designed for negative moment
THEN Perform the TEE beam design
AND  DISPLAY input parameters
AND  DISPLAY tee beam section
AND  DISPLAY negative beam section

RULE Tee cantilever beam
IF  Boundary = 4
AND  Display cantilever tee beam message
AND  we have designed for cantilever negative moment
THEN Perform the TEE beam design
AND  DISPLAY input parameters
AND  DISPLAY cantilever beam section

RULE To display positive moment design message
DISPLAY positive moment design
THEN Display positive moment design message

RULE To display negative moment design message
DISPLAY negative moment design
THEN Display negative moment design message

RULE To display cantilever tee beam message
DISPLAY cantilever tee beam
THEN Display cantilever tee beam message

!-----------------------------------------------------------------------
!  Rectangular positive moment design
!-----------------------------------------------------------------------
!
RULE Rectangular positive moment
IF  we have estimated rectangular dimensions
AND  we have self weight
AND  we have self weight moments
AND  we have selected positive reinforcement
THEN we have designed for rectangular positive moment

RULE To estimate rectangular dimensions
IF  we have rectangular dimensions
AND  NOT user want to change rectangular dimensions
THEN we have estimated rectangular dimensions

RULE Estimate rectangular section
ACTIVATE RECTDIM.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    Minimum h
SEND    Maximum positive moment
SEND    Span length
```

```
SEND     Support width
SEND     Cover
SEND     Boundary
RETURN  b
RETURN  bw
RETURN  d
RETURN  h
IF h > = Minimum h
THEN we have rectangular dimensions
AND  DISPLAY current rectangular dimensions

RULE To let user change dimensions
IF   You want to change dimensions
AND  we have asked for rectangular dimensions
THEN user want to change rectangular dimensions
AND FORGET user want to change rectangular dimensions
AND FORGET You want to change dimensions
AND FORGET we have asked for rectangular dimensions
AND FORGET we have estimated rectangular dimensions
AND DISPLAY changed rectangular dimensions
AND CYCLE

RULE To ask for rectangular dimensions
ASK  bw
ASK  h
IF   h > = Minimum h
THEN we have asked for rectangular dimensions
AND  d := h - Cover - 1.5
ELSE DISPLAY deflection requirement
AND  FORGET we have asked for rectangular dimensions
AND  CYCLE

!-------------------------------------------------------------------------
!  Tee positive moment design
!-------------------------------------------------------------------------
!
RULE Tee positive moment
IF   we have estimated tee dimensions
AND  we have self weight
AND  we have self weight moments
AND  we have selected positive reinforcement
THEN we have designed for tee positive moment

RULE To estimate tee dimensions
IF   we have tee dimensions
AND  NOT user want to change tee dimensions
THEN we have estimated tee dimensions

RULE Estimate tee section
ACTIVATE TEEDIM.EXE
DISK     DATA.DAT
SEND     Fc
SEND     Fy
SEND     Minimum h
SEND     Maximum positive moment
SEND     Span length
```

```
SEND    Support width
SEND    Slab thickness
SEND    Beam spacing
SEND    Cover
SEND    Boundary
RETURN  b
RETURN  bw
RETURN  d
RETURN  h
IF h > = Minimum h
THEN we have tee dimensions
AND  DISPLAY current tee dimensions

RULE To let user change tee dimensions
IF   You want to change dimensions
AND  we have asked for tee dimensions
THEN user want to change tee dimensions
AND FORGET user want to change tee dimensions
AND FORGET You want to change dimensions
AND FORGET we have asked for tee dimensions
AND FORGET we have estimated tee dimensions
AND DISPLAY changed tee dimensions
AND CYCLE

RULE To ask for tee dimensions
ASK  b
ASK  bw
ASK  h
IF   h > = Minimum h
THEN we have asked for tee dimensions
AND  d := h - Cover - 1.5
ELSE DISPLAY deflection requirement
AND  FORGET we have asked for tee dimensions
AND  CYCLE

!------------------------------------------------------------------------
!  Negative moment design
!------------------------------------------------------------------------
!
RULE For continuous negative moment
IF   Boundary = 2
OR   Boundary = 3
AND  we have selected negative reinforcement
THEN we have designed for negative moment

RULE For cantilever negative moment
IF   Boundary = 4
AND  we have estimated cantilever rectangular dimensions
AND  we have self weight
AND  we have self weight moments
AND  we have selected cantilever reinforcement
THEN we have designed for cantilever negative moment

RULE To estimate cantilever rectangular dimensions
IF   we have cantilever rectangular dimensions
```

```
AND  NOT user want to change rectangular dimensions
THEN we have estimated cantilever rectangular dimensions

RULE Estimate cantilever rectangular dimensions
ACTIVATE RECTDIM.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    Minimum h
SEND    Maximum negative moment
SEND    Span length
SEND    Support width
SEND    Cover
SEND    Boundary
RETURN  b
RETURN  bw
RETURN  d
RETURN  h
IF h > = Minimum h
THEN we have cantilever rectangular dimensions
AND  DISPLAY current rectangular dimensions

!------------------------------------------------------------------------
!  Reinforcement selection
!------------------------------------------------------------------------

RULE Positive reinforcement
IF   we have positive reinforcement
AND  NOT section is enlarged
AND  NOT user want to change positive reinforcement
THEN we have selected positive reinforcement

RULE To get positive reinforcement
ACTIVATE POSREIN.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    b
SEND    Slab thickness
SEND    bw
SEND    d
SEND    Maximum positive moment
SEND    Self weight positive moment
SEND    Cover
RETURN  Bar1
RETURN  Size1
RETURN  As provided
RETURN  As required
RETURN  Positive moment capacity
RETURN  Revised positive d
RETURN  Layer of steel
RETURN  Bot1
RETURN  Top1
THEN we have positive reinforcement
```

```
RULE For enlarging positive section
IF   As provided = 0
THEN section is enlarged
AND  DISPLAY current section not adequate
AND  ACTIVATE ENLARGE.EXE
DISK DATA.DAT
SEND b
SEND bw
SEND d
SEND h
SEND Slab thickness
SEND Span length
SEND Beam spacing
SEND Minimum h
SEND Cover
RETURN b
RETURN bw
RETURN d
RETURN h
AND DISPLAY enlarged section
AND FORGET we have self weight
AND FORGET we have self weight moments
AND FORGET we have positive reinforcement
AND FORGET we have selected positive reinforcement
AND FORGET section is enlarged
AND CYCLE

RULE To let user change positive reinforcement
DISPLAY current positive reinforcement
IF   You want to change positive reinforcement
AND  we have asked for positive reinforcement
AND  we have checked reinforcement adequacy
THEN user want to change positive reinforcement
AND  FORGET user want to change positive reinforcement
AND  FORGET You want to change positive reinforcement
AND  FORGET we have asked for positive reinforcement
AND  FORGET we have checked reinforcement adequacy
AND  FORGET we have selected positive reinforcement
AND  CYCLE

RULE Ask user for positive reinforcement desired
ACTIVATE BARCHAN.EXE
DISK     DATA.DAT
SEND     As provided
SEND     As required
SEND     Bar1
SEND     Bar2
SEND     Size1
SEND     Size2
RETURN   Bar1
RETURN   Size1
RETURN   Bar2
RETURN   Size2
RETURN   As provided
THEN we have asked for positive reinforcement
```

```
RULE To check positive reinforcement adequacy
ACTIVATE BARCHECK.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    b
SEND    bw
SEND    d
SEND    d prime
SEND    Slab thickness
SEND    Cover
SEND    Maximum positive moment
SEND    Self weight positive moment
SEND    As provided
SEND    Asp providedN
SEND    As required
SEND    Bar1
SEND    Bar2
SEND    Size1
SEND    Size2
SEND    Section1
RETURN  Positive moment capacity
RETURN  Revised positive d
RETURN  Layer of steel
RETURN  Bot1
RETURN  Bot2
RETURN  Top1
RETURN  Top2
RETURN  Status
IF Status = "OK"
THEN we have checked reinforcement adequacy
ELSE DISPLAY positive reinforcement not adequate
AND  FORGET we have asked for positive reinforcement
AND  FORGET we have checked reinforcement adequacy
AND  CYCLE

RULE Negative reinforcement
IF   we have negative reinforcement
AND  NOT negative section is enlarged
AND  NOT user want to change negative reinforcement
THEN we have selected negative reinforcement

RULE To get negative reinforcement
ACTIVATE NEGREIN.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    bw
SEND    d
SEND    h
SEND    Maximum negative moment
SEND    Self weight negative moment
SEND    As provided
SEND    Bot1
SEND    Bot2
SEND    Layer of steel
```

```
SEND    Size1
SEND    Size2
SEND    Cover
RETURN  Bar1N
RETURN  Size1N
RETURN  As providedN
RETURN  As requiredN
RETURN  Negative moment capacity
RETURN  Revised negative d
RETURN  d prime
RETURN  Layer of steelN
RETURN  Top1N
RETURN  Bot1N
RETURN  Barp1N
RETURN  Barp2N
RETURN  Asp providedN
THEN we have negative reinforcement

RULE To enlarge negative section
IF   As providedN = 0
THEN negative section is enlarged
AND  DISPLAY section not adequate
AND  ACTIVATE ENLARGE.EXE
DISK DATA.DAT
SEND b
SEND bw
SEND d
SEND h
SEND Slab thickness
SEND Span length
SEND Beam spacing
SEND Minimum h
SEND Cover
RETURN b
RETURN bw
RETURN d
RETURN h
AND DISPLAY enlarged section
AND FORGET we have self weight
AND FORGET we have self weight moments
AND FORGET we have selected positive reinforcement
AND FORGET we have designed for rectangular positive moment
AND FORGET we have designed for tee positive moment
AND FORGET we have positive reinforcement
AND FORGET user want to change positive reinforcement
AND FORGET You want to change positive reinforcement
AND FORGET we have asked for positive reinforcement
AND FORGET we have checked reinforcement adequacy
AND FORGET we have negative reinforcement
AND FORGET we have selected negative reinforcement
AND FORGET negative section is enlarged
AND FORGET Display positive moment design message
AND FORGET Display negative moment design message
AND CYCLE
```

```
RULE To let user change negative reinforcement
DISPLAY current negative reinforcement
IF  You want to change negative reinforcement
AND  we have asked for negative reinforcement
AND  we have checked negative reinforcement adequacy
THEN user want to change negative reinforcement
AND  FORGET user want to change negative reinforcement
AND  FORGET You want to change negative reinforcement
AND  FORGET we have asked for negative reinforcement
AND  FORGET we have checked negative reinforcement adequacy
AND  FORGET we have selected negative reinforcement
AND  CYCLE

RULE Ask user for negative reinforcement desired
ACTIVATE BARCHAN.EXE
DISK     DATA.DAT
SEND     As providedN
SEND     As requiredN
SEND     Bar1N
SEND     Bar2N
SEND     Size1N
SEND     Size2N
RETURN   Bar1N
RETURN   Size1N
RETURN   Bar2N
RETURN   Size2N
RETURN   As providedN
THEN we have asked for negative reinforcement

RULE To check negative reinforcement adequacy
ACTIVATE BARCHECK.EXE
DISK     DATA.DAT
SEND     Fc
SEND     Fy
SEND     b
SEND     bw
SEND     d
SEND     d prime
SEND     Slab thickness
SEND     Cover
SEND     Maximum negative moment
SEND     Self weight negative moment
SEND     As providedN
SEND     Asp providedN
SEND     As requiredN
SEND     Bar1N
SEND     Bar2N
SEND     Size1N
SEND     Size2N
SEND     Section2
RETURN   Negative moment capacity
RETURN   Revised negative d
RETURN   Layer of steelN
RETURN   Top1N
RETURN   Top2N
RETURN   Bot1N
```

```
RETURN   Bot2N
RETURN   Status
IF Status = "OK"
THEN we have checked negative reinforcement adequacy
ELSE DISPLAY negative reinforcement not adequate
AND  FORGET we have asked for negative reinforcement
AND  FORGET we have checked negative reinforcement adequacy
AND  CYCLE

RULE Cantilever reinforcement
IF   we have cantilever reinforcement
AND  NOT cantilever section is enlarged
AND  NOT user want to change cantilever reinforcement
THEN we have selected cantilever reinforcement

RULE To get cantilever reinforcement
ACTIVATE POSREIN.EXE
DISK     DATA.DAT
SEND     Fc
SEND     Fy
SEND     b
SEND     Slab thickness
SEND     bw
SEND     d
SEND     Maximum negative moment
SEND     Self weight negative moment
SEND     Cover
RETURN   Bar1N
RETURN   Size1N
RETURN   As providedN
RETURN   As requiredN
RETURN   Negative moment capacity
RETURN   Revised negative d
RETURN   Layer of steelN
RETURN   Top1N
RETURN   Bot1N
THEN we have cantilever reinforcement

RULE For enlarging cantilever section
IF   As providedN = 0
THEN cantilever section is enlarged
AND  DISPLAY current section not adequate
AND  ACTIVATE ENLARGE.EXE
DISK DATA.DAT
SEND b
SEND bw
SEND d
SEND h
SEND Slab thickness
SEND Span length
SEND Beam spacing
SEND Minimum h
SEND Cover
RETURN b
RETURN bw
RETURN d
```

```
RETURN h
AND DISPLAY enlarged section
AND FORGET we have self weight
AND FORGET we have self weight moments
AND FORGET we have cantilever reinforcement
AND FORGET cantilever section is enlarged
AND CYCLE

RULE To let user change cantilever reinforcement
DISPLAY current negative reinforcement
IF   You want to change negative reinforcement
AND  we have asked for negative reinforcement
AND  we have checked cantilever reinforcement adequacy
THEN user want to change cantilever reinforcement
AND  FORGET user want to change cantilever reinforcement
AND  FORGET You want to change cantilever reinforcement
AND  FORGET we have asked for negative reinforcement
AND  FORGET we have checked cantilever reinforcement adequacy
AND  FORGET we have selected cantilever reinforcement
AND  CYCLE

RULE To check cantilever reinforcement adequacy
ACTIVATE BARCHECK.EXE
DISK    DATA.DAT
SEND    Fc
SEND    Fy
SEND    b
SEND    bw
SEND    d
SEND    d prime
SEND    Slab thickness
SEND    Cover
SEND    Maximum negative moment
SEND    Self weight negative moment
SEND    As providedN
SEND    Asp providedN
SEND    As requiredN
SEND    Bar1N
SEND    Bar2N
SEND    Size1N
SEND    Size2N
SEND    Section1
RETURN  Negative moment capacity
RETURN  Revised negative d
RETURN  Layer of steelN
RETURN  Top1N
RETURN  Top2N
RETURN  Bot1N
RETURN  Bot2N
RETURN  Status
IF Status = "OK"
THEN we have check cantilever reinforcement adequacy
ELSE DISPLAY negative reinforcement not adequate
AND  FORGET we have ask for cantilever reinforcement
AND  FORGET we have check cantilever reinforcement adequacy
AND  CYCLE
```

```
!-------------------------------------------------------------------------
!  Calculate self-weight
!-------------------------------------------------------------------------
!
RULE To calculate self weight
ACTIVATE SELFWEHT.EXE
DISK     DATA.DAT
SEND     b
SEND     bw
SEND     h
SEND     Slab thickness
RETURN   Self weight
THEN     we have self weight

RULE To calculate self weight moments
IF   Boundary = 1
OR   Boundary = 2
OR   Boundary = 4
AND  ACTIVATE MOMENT.EXE
DISK     DATA.DAT
SEND     Boundary
SEND     Loading
SEND     Self weight
SEND     Beam spacing
SEND     Span length
SEND     Right span length
SEND     Left span length
SEND     Support width
SEND     Slab thickness
RETURN   Self weight positive moment
RETURN   Self weight negative moment
THEN we have self weight moments

RULE To calculate moment for one end continuous beam
IF   Boundary = 3
AND  we have asked for more beam information
AND  ACTIVATE MOMENT.EXE
DISK     DATA.DAT
SEND     Boundary
SEND     Loading
SEND     Self weight
SEND     Beam spacing
SEND     Span length
SEND     Right span length
SEND     Left span length
SEND     Support width
SEND     Slab thickness
SEND     discontinuous end unrestrained
SEND     discontinuous end has a spandrel beam as support
SEND     Number of span
RETURN   Self weight positive moment
RETURN   Self weight negative moment
THEN we have self weight moments
```

```
!------------------------------------------------------------------------------
!   Text for questions
!------------------------------------------------------------------------------
!
TEXT Title of job
```

Please enter a job name:

TEXT Fc

Please enter concrete compressive strength , Fc′ in ksi.

TEXT Fy

Please enter steel yield strength , Fy in ksi.

TEXT Slab thickness

Please enter slab thickness in inches.

TEXT Span length

Please enter span length in inches.

TEXT Right span length

Please enter right adjacent span length in inches.

TEXT Left span length

Please enter left adjacent span length in inches.

TEXT Support width

Please enter support width in inches.

TEXT Beam Spacing

Please enter beam spacing (center to center of beams) in inches.

TEXT Beam

Select exposure conditions.

TEXT Loads and moments

Do you know the loading and moments ?

TEXT Beam boundary

Select support conditions.

TEXT Maximum positive moment

Please enter the maximum positive moment in in-kip.

TEXT Maximum negative moment

Please enter the maximum negative moment in in-kip.

TEXT Occupancy

Select occupancy or function of the structure in which the beam is located. Press F1 for more occupancy.

TEXT You want to change dimensions

Do you want to change the selected dimensions ? If yes, select "True".

TEXT b

Please enter flange width 'b' in inches.

TEXT bw

Please enter web width 'bw' in inches.

TEXT h

Please enter overall depth 'h' in inches.

TEXT You want to change positive reinforcement

Do you want to change the selected reinforcement ? If yes, select "True".

TEXT You want to change negative reinforcement

Do you want to change the selected reinforcement ? If yes, select "True".

TEXT discontinuous end unrestrained

If the discontinuous end of the beam is unrestrained, select "True". Select "False" if it is fixed.

TEXT discontinuous end has a spandrel beam as support

If the discontinuous end of the beam is supported on a spandrel beam then select "True". Select "False" if it is supported on a column.

TEXT Number of span

Please enter the number of spans.

```
!---------------------------------------------------------------------------
!   Informations display
!---------------------------------------------------------------------------
!
```

DISPLAY need info

We need to know the following information before the beam can be designed.

* Concrete compressive strength, Fc′
* Steel yield strength, Fy
* Span length
* Support conditions
* Exposure conditions

Please obtain all this information prior to designing the beam.

Press F3 STRT to restart the session.

DISPLAY section to be

The beam can be designed as a :

[Section] section.

Press F2 CONT to continue and select the section to be designed from the menu.

OR Press F3 STRT to restart the session.

DISPLAY positive moment design

Designing for maximum positive moment i.e., the mid-span section.

Press F2 CONT to continue

DISPLAY negative moment design

Designing for maximum negative moment i.e., the section at the supports.

Press F2 CONT to continue.

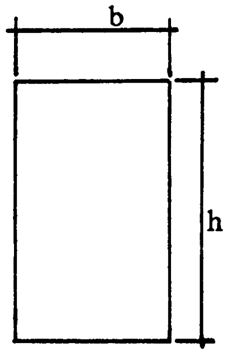
DISPLAY cantilever tee beam

For a cantilever beam, even though the beam is tee-section, it will be designed as a rectangular section. This is because the flange of the beam is in tension and would not add any strength to the beam as concrete is weak in tension.

Press F2 CONT to continue

DISPLAY current rectangular dimensions

Recommended dimensions :



b = [bw (5,1)] in

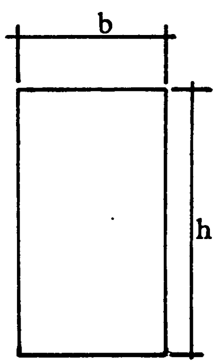
h = [h (6,1)] in

You may change the section if you have some other dimensions which you would like to use.

Press F2 CONT to continue OR change section.

DISPLAY changed rectangular dimensions

These are the dimensions you entered :



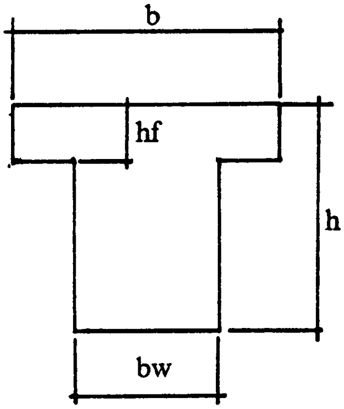
b = [bw (5,1)] in

h = [h (6,1)] in

Press F2 CONT to continue OR change section.

DISPLAY current tee dimensions

Recommended dimensions :



b = [b (6,1)] in

bw = [bw (5,1)] in

h = [h (6,1)] in

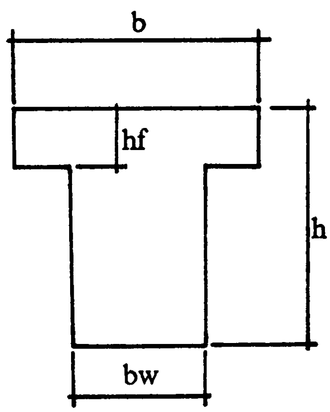
hf = [Slab thickness(4,1)] in

You may change the section if you have some other dimensions which you would like to use.

Press F2 CONT to continue OR change section.

DISPLAY changed tee dimensions

These are the dimensions you entered :



b = [b (6,1)] in

bw = [bw (5,1)] in

h = [h (6,1)] in

hf = [Slab thickness(4,1)] in

Press F2 CONT to continue OR change section.

DISPLAY deflection requirement

The overall depth 'h' of the beam must satisfy the minimum depth requirement for deflection specified by the Building Code ACI 318-83, Section 9.5.2.2.

Therefore, the depth has to be increased.

For this case, minimum h required is : [Minimum h] in

Press F2 CONT to reenter a new dimensions

DISPLAY current positive reinforcement

This is currently the selected reinforcement
at the positive moment section :

Total area = [As provided (5,2)] sq. in
Size one : [Bar1(2,0)] # [Size1(2,0)]
Size two : [Bar2(2,0)] # [Size2(2,0)]

Press F2 CONT to Continue

DISPLAY current negative reinforcement

This is currently the selected reinforcement
at the negative moment section :

Total area = [As providedN (5,2)] sq. in
Size one : [Bar1N(2,0)] # [Size1N(2,0)]
Size two : [Bar2N(2,0)] # [Size2N(2,0)]

Press F2 CONT to Continue

DISPLAY current section not adequate

The section selected has to be enlarged as no suitable
reinforcement can be found to fit in the beam and
at the same time satisfy all the necessary requirements.

Press F2 CONT to continue

DISPLAY section not adequate

The section previously selected for positive moment, i.e., at the mid-span is not adequate for negative moment, i.e., at the support. Therefore, the section at positive moment region has to be enlarged and redesigned.

Press F2 CONT to Continue

DISPLAY enlarged section

The enlarged section is as follow :

Flange width, b if any = [b (6,1)] in

Web width , bw = [bw (5,1)] in

Effective depth, d = [d (5,1)] in

Overall depth, h = [h (6,1)] in

Press F2 CONT to continue

DISPLAY positive reinforcement not adequate

This is the reinforcement you selected :

Total area = [As provided (5,2)] sq. in
Size one : [Bar1(2,0)] # [Size1(2,0)]
Size two : [Bar2(2,0)] # [Size2(2,0)]

The above reinforcement is not adequate. It does not satisfy rule number [Status].

Press F2 to continue OR F1 to see explanation.

Reinforcement has to satisfy the following rules :-

1. Total number of bars should be more than 2 and less than 12, for proper placement.

2. Difference in bar sizes should not be more than 2 standard sizes.

3. Steel selected cannot be placed in 2 layers, increase bar sizes and reduce number of bars.

4. Symmetry cannot be maintained about the vertical axis or there is only one bar in the second layer.

5. Steel is placed in one layer but symmetry cannot be maintained about the vertical axis.

6. Steel percent is greater than allowed by ACI 318-83, reduce steel area.

7. Steel percent is smaller than allowed by ACI 318-83, increase steel area.

8. Capacity of the section is insufficient, increase steel area.

Press F2 to continue.

DISPLAY negative reinforcement not adequate

This is the reinforcement you selected :

Total area = [As provided (5,2)] sq. in
Size one : [Bar1(2,0)] # [Size1(2,0)]
Size two : [Bar2(2,0)] # [Size2(2,0)]

The above reinforcement is not adequate. It does not satisfy rule number [Status].

Press F2 to continue OR F1 to see explanation.

Reinforcement has to satisfy the following rules :-

1. Total number of bars should be more than 2 and less than 12, for proper placement.

2. Difference in bar sizes should not be more than 2 standard sizes.

3. Steel selected cannot be placed in 2 layers, increase bar sizes and reduce number of bars.

4. Symmetry cannot be maintained about the vertical axis or there is only one bar in the second layer.

5. Steel is placed in one layer but symmetry cannot be maintained about the vertical axis.

6. Steel percent is greater than allowed by ACI 318-83, reduce steel area.

7. Steel percent is smaller than allowed by ACI 318-83, increase steel area.

8. Capacity of the section is insufficient, increase steel area.

Press F2 to continue.

DISPLAY input parameters

Input information :

Title : [Title of job]

Concrete compressive strength, Fc′ = [Fc (2,0)] ksi
Steel yield strength, Fy = [Fy (3,0)] ksi
Slab thickness = [Slab thickness (4,1)] in.
Span length = [Span length] in.
Support width = [Support width (4,1)] in.
Beam spacing = [Beam spacing] in.

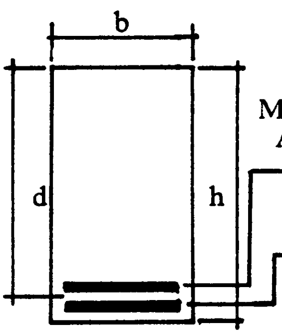
Maximum positive moment = [Maximum positive moment] in-kip
Self weight positive moment = [Self weight positive moment] in-kip
Maximum negative moment = [Maximum negative moment] in-kip
Self weight negative moment = [Self weight negative moment] in-kip

Beam is [Beam]
Beam is [Beam boundary]

Press F2 CONT to see result.

DISPLAY rectangular beam section

The beam is designed as a rectangular section.
Recommended mid-span section is as follows:



b = [bw (5,1)] in.
d = [Revised positive d (5,1)] in.
h = [h (6,1)] in.
cover = [Cover (3,1)] in.
Moment capacity = [Positive moment capacity] in-kip
Area of steel = [As provided (5,2)] sq. in.
Top layer : [Top1(3,0)] # [Size1(2,0)]
& [Top2(3,0)] # [Size2(2,0)]

Bottom layer: [Bot1(3,0)] # [Size1(2,0)]
& [Bot2(3,0)] # [Size2(2,0)]

Press F2 CONT to see support section for continuous beam

OR Press F3 STRT to restart session for other beams

DISPLAY tee beam section

The beam is designed as tee section.
Recommended mid-span section is as follows:
b
b = [b (6,1)] in.
bw = [bw (5,1)] in.
d = [Revised positive d (5,1)] in.
hf h = [h (6,1)] in.
hf = [Slab thickness(4,1)] in.
cover = [Cover (3,1)] in.
d h Moment capacity = [Positive moment capacity] in-kip
Area of steel = [As provided (5,2)] sq. in.
Top layer : [Top1(3,0)] # [Size1(2,0)]
& [Top2(3,0)] # [Size2(2,0)]

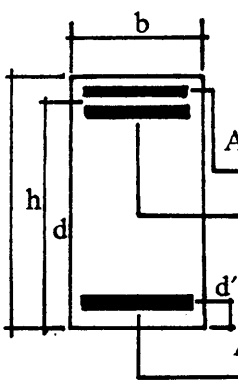
Bottom layer: [Bot1(3,0)] # [Size1(2,0)]
& [Bot2(3,0)] # [Size2(2,0)]

Press F2 CONT to see support section for continuous beam

OR Press F3 STRT to restart session for other beams

DISPLAY negative beam section

The negative moment section is designed as a rectangular
doubly reinforced section.
Results are as follows:
b = [bw (5,1)] in.
d = [Revised negative d (5,1)] in.
d′ = [d prime (4,1)] in.
h = [h (6,1)] in.
cover = [Cover (3,1)] in.



Moment capacity = [Negative moment capacity] in-kip
Area of tension steel = [As providedN (5,2)] sq. in.
Top layer : [Top1N(3,0)] # [Size1N(2,0)]
& [Top2N(3,0)] # [Size2N(2,0)]

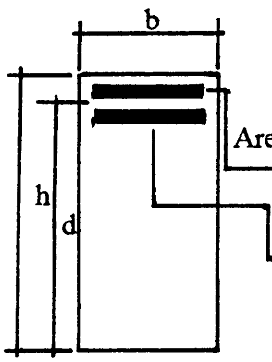
Bottom layer: [Bot1N(3,0)] # [Size1N(2,0)]
& [Bot2N(3,0)] # [Size2N(2,0)]

Area of compression steel = [Asp providedN (5,2)] sq. in.
Bottom steel: [Barp1N(3,0)] # [Size1(2,0)]
& [Barp2N(3,0)] # [Size2(2,0)]

Press Function Key 3 STRT to restart the session

DISPLAY cantilever beam section

The beam is designed as a rectangular section.
Results are as follows:



b = [bw (5,1)] in.
d = [Revised negative d (5,1)] in.
h = [h (6,1)] in.
cover = [Cover (3,1)] in.
Moment capacity = [Negative moment capacity] in-kip
Area of tension steel = [As providedN (5,2)] sq. in.
Top layer : [Top1N(3,0)] # [Size1N(2,0)]
& [Top2N(3,0)] # [Size2N(2,0)]
Bottom layer: [Bot1N(3,0)] # [Size1N(2,0)]
& [Bot2N(3,0)] # [Size2N(2,0)]

Press Function Key 3 STRT to restart the session

!---
! Explanatory information displays
!---
!
EXPAND Fc

CONCRETE YIELD STRENGTH

Commonly used structural concrete is concrete with
a compressive yield strength of 3 ksi to 8 ksi.

For most purposes, concrete with a compressive
strength of 4 ksi is adequate.

EXPAND Fy

STEEL YIELD STRENGTH

Most commonly used steel reinforcement is that which has
a tensile yield strength ranging from 40 ksi to 60 ksi.

For most purposes, steel with a yield strength
of 60 ksi is adequate.

EXPAND Span length

$\boxed{\text{SPAN LENGTH}}$

Span length is taken center to center of supports for simply supported beams.

For a continuous beam, span length can be taken as the clear span (i.e. face to face of supports) provided that the support width will be entered as zero later on, otherwise, take span length from center to center of supports.

For cantilever beams, span length is measured from the free end to the face of support.

EXPAND Right span length

$\boxed{\text{RIGHT ADJACENT SPAN LENGTH}}$

This is the length of the span right adjacent to the beam. For simply supported and cantilevered beams, enter zero.

For a continuous beam, it can be taken as the clear span (i.e. face to face of supports) provided that the support width will be entered as zero later on, otherwise, measure from center to center of supports.

EXPAND Left span length

$\boxed{\text{LEFT ADJACENT SPAN LENGTH}}$

This is the length of the span left adjacent to the beam. For simply supported and cantilever beams, enter zero.

supported beams.

For a continuous beam, it can be taken as the clear span (i.e. face to face of supports) provided that the support width will be entered as zero later on, otherwise, measure from center to center of supports.

For cantilever beams, it is measured from the free end to the face of support.

EXPAND Support width

SUPPORT WIDTH

Support width for simply supported and cantilever beams can be taken as zero.

For a continuous beam, if span length is taken as clear span then enter zero for support width, otherwise, enter as given.

EXPAND Maximum positive moment

MAXIMUM POSITIVE MOMENT

This moment shall include moment due to all super-imposed dead load including slab weight and all live loads. Moment due to self-weight will be automatically included.

EXPAND Maximum negative moment

MAXIMUM NEGATIVE MOMENT

This moment shall include moment due to all super-imposed dead load including slab weight and all live loads. Moment due to self-weight will be automatically included.

EXPAND Occupancy

$\boxed{\text{OCCUPANCY OR FUNCTION}}$

Occupancy is required for determining the minimum uniform live load the beam is going to carry.

EXPAND Beam

$\boxed{\text{EXPOSURE}}$

The exposure conditions of the beam are required for determining clear cover for concrete protection according to guidelines of ACI 318-83, Section 7.7.1.

EXPAND Beam spacing

$\boxed{\text{BEAM SPACING}}$

The spacing of beams shall be taken as the center to center distance of beams if all spacing is the same.

If spacing of beam is different, take spacing as the average of the spacing of two adjacent beams.

END

C.2 Routine: SELFWEHT.BAS

```
'    ******************************************************************
'    *                SELFWEHT.BAS                                    *
'    *                                                   *
'    *       A routine to calculate the self-weight of a             *
'    *       section.                                  *
'    ******************************************************************
'
     OPEN "DATA.DAT" FOR INPUT AS #1
     INPUT #1, Total
     INPUT #1, b$
     INPUT #1, bw$
     INPUT #1, h$
     INPUT #1, hf$
     CLOSE #1
'
     b = VAL(RIGHT$(b$, LEN(b$) - 1))
     bw = VAL(RIGHT$(bw$, LEN(bw$) - 1))
     h = VAL(RIGHT$(h$, LEN(h$) - 1))
     hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
'
     IF b = 0 THEN
       SW = 150 * (bw / 12) * (h / 12)
     ELSE
       SW = 150 * (bw / 12) * ((h - hf) / 12)
     END IF
'
     OPEN "DATA.DAT" FOR OUTPUT AS #1
     PRINT #1, "1"
     PRINT #1, "N"; SW
     CLOSE #1
'
     END
```

C.3 Routine: MOMENT.BAS

```
'   *******************************************************************
'   *            ROUTINE : MOMENT.BAS                                 *
'   *                                                    *
'   *     This routine is written to calculate moments                *
'   *     for a beam using the approximate method given               *
'   *     in Section 8.3.3 of Building Code ACI 318-83.It is          *
'   *     meant to be used with the expert system program             *
'   *     "BEAMDES.KNB".                                              *
'   *******************************************************************
'
'-----------Variable Descriptions--------------------------------------
'
'    Boundary - Beam boundary condition (receive from BEAMDES.KNB)
'    LOAD     - Loading (psf)
'    Beamspac - Spacing of beams
'    L        - Span length
'    SLR      - Right adjacent span length
'    SLL      - Left adjacent span length
'    POSMOM   - Maximun positive moment
'    NEGMOM   - Maximum negative moment
'    LEFMOM   - Left end moment
'    RIGMOM   - Right end moment
'    Support  - Width of support
'
'-----------Read data from DATA.DAT-------------------------------------
'
   OPEN "DATA.DAT" FOR INPUT AS #1
   INPUT #1, TOTAL
   INPUT #1, Boundary$
   INPUT #1, LOADS$
   INPUT #1, SW$
   INPUT #1, Beamspac$
   INPUT #1, L$
   INPUT #1, SLR$
   INPUT #1, SLL$
   INPUT #1, Support$
   INPUT #1, hf$
   Boundary = VAL(RIGHT$(Boundary$, LEN(Boundary$) - 1))
   IF Boundary = 3 THEN
     INPUT #1, Disc$
     INPUT #1, Supp$
     INPUT #1, Nospan$
     Disc$ = RIGHT$(Disc$, LEN(Disc$) - 1)
     Supp$ = RIGHT$(Supp$, LEN(Supp$) - 1)
     Nospan% = VAL(RIGHT$(Nospan$, LEN(Nospan$) - 1))
   END IF
   CLOSE #1
'
   LOADS = VAL(RIGHT$(LOADS$, LEN(LOADS$) - 1))
   SW = VAL(RIGHT$(SW$, LEN(SW$) - 1))
   Beamspac = VAL(RIGHT$(Beamspac$, LEN(Beamspac$) - 1))
   L = VAL(RIGHT$(L$, LEN(L$) - 1))
```

```
    SLR = VAL(RIGHT$(SLR$, LEN(SLR$) - 1))
    SLL = VAL(RIGHT$(SLL$, LEN(SLL$) - 1))
    Support = VAL(RIGHT$(Support$, LEN(Support$) - 1))
    hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
'
    IF SW <> 0 THEN
      Wu = 1.4 * SW / 1000                'factored self-weight(klf)
    ELSE
      DL = 1.4 * (.15 * (hf / 12) * (Beamspac / 12))'factored dead load (klf)
      LL = 1.7 * (Beamspac / 12) * LOADS / 1000'factored live load (klf)
      Wu = DL + LL                        'factored total load (klf)
    END IF
'
'----------Calculate moments------------------------------------------
'
    IF SLR > 0 AND SLL > 0 THEN
      ln = ((L - Support) + (SLR - Support) + (SLL - Support)) / 3
    ELSEIF SLR > 0 AND SLL = 0 THEN
      ln = ((L - Support) + (SLR - Support)) / 2
    ELSEIF SLR = 0 AND SLL > 0 THEN
      ln = ((L - Support) + (SLL - Support)) / 2
    ELSEIF SLR = 0 AND SLL = 0 THEN
      ln = L - Support
    END IF
    ln1 = L - Support
    IF Boundary = 1 THEN                 'simply supported
      POSMOM = (Wu * L^2) / (8 * 12)
    ELSEIF Boundary = 2 THEN               'both ends continuous
      POSMOM = (Wu * ln1^2) / (16 * 12)
      LEFMOM = (Wu * ln^2) / (11 * 12)
      RIGMOM = LEFMOM
    END IF
'
    IF Boundary = 3 THEN                 'one end continuous
'
'     assuming left end discontinuous
'
      IF Disc$ = "T" THEN                'discontinuous end unrestrained
        POSMOM = (Wu * ln1^2) / (11 * 12)
      ELSE                               'discontinuous end restrained
        POSMOM = (Wu * ln1^2) / (14 * 12)
        IF Supp$ = "T" THEN              'support is a spandrel beam
          LEFMOM = (Wu * ln^2) / (24 * 12)
        ELSE                             'support is a column
          LEFMOM = (Wu * ln^2) / (16 * 12)
        END IF
      END IF
'
      IF Nospan% = 2 THEN                'beam has two span
        RIGMOM = (Wu * ln^2) / (9 * 12)
      ELSEIF Nospan% > 2 THEN              'beam has more than two span
        RIGMOM = (Wu * ln^2) / (10 * 12)
      END IF
'
    END IF
'
```

```
    IF Boundary = 4 THEN                    'cantilever beam
'
'     assuming right end is free
'
      LEFMOM = (Wu * L ^ 2) / 24
'
    END IF
'
    NEGMOM = LEFMOM
    IF NEGMOM < RIGMOM THEN
      NEGMOM = RIGMOM
    END IF
'
'---------Write to DATA.DAT--------------------------------------------
'
    OPEN "DATA.DAT" FOR OUTPUT AS #1
    PRINT #1, "2"
    PRINT #1, "N"; POSMOM
    PRINT #1, "N"; NEGMOM
    CLOSE #1
'
    END
```

C.4 Routine: RECTDIM.BAS

```
'     ****************************************************************
'     *                RECTDIM.BAS                          *
'     *                                           *
'     *     A routine to assist the expert system program       *
'     *     "BEAMDES" to select a suitable rectangular beam      *
'     *     section.                                  *
'     ****************************************************************
'
'----------Variable Descriptions----------------------------------------
'
'    Mu     = Ultimate moment without self-weight
'    SW     = Factored self-weight
'    Musw   = Ultimate moment with self-weight
'    L      = Span length
'    covpro = cover provision from center of steel to bottom face of beam
'    Support = width of support
'
DECLARE SUB MOMENT (SW, Boundary%, L, Musw)
'
'---------Read parameters from a disk file------------------------------
'
     OPEN "DATA.DAT" FOR INPUT AS #1
     INPUT #1, TOTAL
     INPUT #1, Fc$
     INPUT #1, Fy$
     INPUT #1, hmin$
     INPUT #1, Mu$
     INPUT #1, L$
     INPUT #1, Support$
     INPUT #1, cover$
     INPUT #1, Boundary$
     CLOSE #1
'
'    convert the string variables into real numbers
'
     Fc = VAL(RIGHT$(Fc$, LEN(Fc$) - 1))
     Fy = VAL(RIGHT$(Fy$, LEN(Fy$) - 1))
     hmin = VAL(RIGHT$(hmin$, LEN(hmin$) - 1))
     Mu = VAL(RIGHT$(Mu$, LEN(Mu$) - 1))
     L = VAL(RIGHT$(L$, LEN(L$) - 1))
     Support = VAL(RIGHT$(Support$, LEN(Support$) - 1))
     cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
     Boundary% = VAL(RIGHT$(Boundary$, LEN(Boundary$) - 1))
'
'
'----------ASSUME A RHO AND FIRST ESTIMATE OF bw and h--------------------
'
    IF Fc < 4 THEN
      Beta1 = .85
    ELSEIF Fc >= 4 AND Fc <= 8 THEN
      Beta1 = .85 - .05 * (Fc - 4)
    ELSE
```

```
        Beta1 = .65
    END IF
'
    RhoBal = (Beta1 * .85 * Fc / Fy) * (87 / (87 + Fy))
    Rho = .5 * RhoBal
'
    w = Rho * Fy / Fc
    R = w * Fc * (1 - .59 * w)
'
    IF cover = 1.5 THEN
       covpro = 3
    ELSEIF cover = 2 THEN
       covpro = 3.5
    ELSEIF cover = 3 THEN
       covpro = 4.5
    END IF
'
    d = (Mu / (.9 * .5 * R))^.3333       'assuming bw=0.5d
    bw = CINT(.5 * d)
    h = CINT(d + covpro)
'
'   check if h satisfy deflection requirement
'
    IF h < hmin THEN
      h = (FIX(hmin)) + 1
      bw = CINT(.5 * (h - covpro))
    END IF
'
    SW = .15 * (h / 12) * (bw / 12) * 1.4   'self-weight
    CALL MOMENT(SW, Boundary%, L, Musw)    'calculate self-weight moment
    Mtotal = Mu + Musw                      'total moment
'
'---------SECOND ESTIMATE OF bw and h--------------------------------------
'
    d = (Mtotal / (.9 * .5 * R))^.3333   'new d
    bw = CINT(.5 * d)                    'new bw
'
'   choose a 'bw' which is multiples of 2 or 3
'
    IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
       bw = bw
    ELSE
       bw = bw + 1
    END IF
'
    IF bw < 6 THEN                       'smallest practical beam width
      bw = 6
    END IF
'
    h = d + covpro                       'new h
'
    IF h <> FIX(h) THEN
      h = FIX(h) + 1                     'round h to next whole number
    END IF
    IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
      h = h
```

```
    ELSE
      h = h - 1
    END IF
'
'   check if h satisfy deflection requirement
'
    IF h < hmin THEN
      h = FIX(hmin + 1)
      IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
        h = h
      ELSE
        h = h + 1
      END IF
      bw = FIX(.5 * (h - covpro))                'adjust bw
      IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
        bw = bw
      ELSE
        bw = bw + 1
      END IF
    END IF
'
    d = h - covpro
'
'----------Return parameter to disk file---------------------------------
'
    OPEN "DATA.DAT" FOR OUTPUT AS #1
    PRINT #1, "4"
    PRINT #1, "N"; b
    PRINT #1, "N"; bw
    PRINT #1, "N"; d
    PRINT #1, "N"; h
    CLOSE #1
'
END

'
'   Subroutine to calculate self-weight moment
'
SUB MOMENT (SW, Boundary%, L, Musw) STATIC
'
     SHARED Support
     Ln = L - Support
     IF Boundary% = 1 THEN             'simply supported
       Musw = (SW * L^2) / (8 * 12)
     ELSEIF Boundary% = 2 THEN          'both ends continuous
       Musw = (SW * Ln^2) / (16 * 12)
     ELSEIF Boundary% = 3 THEN          'one end continuous
       Musw = (SW * Ln^2) / (11 * 12)
     ELSEIF Boundary% = 4 THEN          'cantilever
       Musw = (SW * L^2) / 24
     END IF
END SUB
```

C.5 Routine: TEEDIM.BAS

```
'     ****************************************************************
'     *                 TEEDIM.BAS                     *
'     *                                                *
'     *   A routine to assist the expert system program         *
'     *   "BEAMDES" to select a suitable tee beam section.      *
'     *                                                *
'     ****************************************************************
'
'----------Variable Descriptions----------------------------------------
'
'     Mu      = Ultimate moment without self-weight
'     SW      = Factored self-weight
'     Musw    = Ultimate moment with self-weight
'     L       = Span length
'     covpro  = cover provision from center of steel to bottom face of beam
'     Support = width of support
'
DECLARE SUB MOMENT (SW, Boundary%, L, Musw)

'---------Read parameters from a disk file------------------------------
'
     OPEN "DATA.DAT" FOR INPUT AS #1
     INPUT #1, TOTAL
     INPUT #1, Fc$
     INPUT #1, Fy$
     INPUT #1, hmin$
     INPUT #1, Mu$
     INPUT #1, L$
     INPUT #1, Support$
     INPUT #1, hf$
     INPUT #1, Beamspac$
     INPUT #1, cover$
     INPUT #1, Boundary$
     CLOSE #1
'
'    convert the string variables into real numbers
'
     Fc = VAL(RIGHT$(Fc$, LEN(Fc$) - 1))
     Fy = VAL(RIGHT$(Fy$, LEN(Fy$) - 1))
     hmin = VAL(RIGHT$(hmin$, LEN(hmin$) - 1))
     Mu = VAL(RIGHT$(Mu$, LEN(Mu$) - 1))
     L = VAL(RIGHT$(L$, LEN(L$) - 1))
     Support = VAL(RIGHT$(Support$, LEN(Support$) - 1))
     hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
     Beamspac = VAL(RIGHT$(Beamspac$, LEN(Beamspac$) - 1))
     cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
     Boundary% = VAL(RIGHT$(Boundary$, LEN(Boundary$) - 1))
'
'
'----------ASSUME A RHO AND FIRST ESTIMATE OF bw and h--------------------
'
    IF Fc < 4 THEN
```

```
    Beta1 = .85
  ELSEIF Fc >= 4 AND Fc <= 8 THEN
    Beta1 = .85 - .05 * (Fc - 4)
  ELSE
    Beta1 = .65
  END IF
'
  RhoBal = (Beta1 * .85 * Fc / Fy) * (87 / (87 + Fy))
  Rho = .5 * RhoBal
'
  w = Rho * Fy / Fc
  R = w * Fc * (1 - .59 * w)
'
  IF cover = 1.5 THEN
     covpro = 3
  ELSEIF cover = 2 THEN
     covpro = 3.5
  ELSEIF cover = 3 THEN
     covpro = 4.5
  END IF
'
  d = (Mu / (.9 * .5 * R))^ .3333        'assuming bw=0.5d
  bw = CINT(.5 * d)
  h = CINT(d + covpro)
'
'   check if h satisfy deflection requirement
'
  IF h < hmin THEN
    h = (FIX(hmin)) + 1
    bw = CINT(.5 * (h - covpro))
  END IF
'
  SW = .15 * ((h - hf) / 12) * (bw / 12) * 1.4 'self-weight
  CALL MOMENT(SW, Boundary%, L, Musw)       'calculate self-weight moment
  Mtotal = Mu + Musw                        'total moment
'
'---------SECOND ESTIMATE OF bw and h--------------------------------------
'
  d = (Mtotal / (.9 * .5 * R))^ .3333   'new d
  bw = CINT(.5 * d)                     'new bw
'
'   choose a 'bw' which is multiples of 2 or 3
'
  IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
     bw = bw
  ELSE
     bw = bw + 1
  END IF
'
  IF bw < 6 THEN                        'smallest practical beam width
    bw = 6
  END IF
'
'   check if the span/overall depth ratio is reasonable
'
```

```
      spandept = L / d
'
      IF spandept < = 12 THEN
         d = L / 12
      ELSEIF spandept > = 18 THEN
         d = L / 18
      END IF
'
      h = d + covpro                    'new h
'
      IF h < > FIX(h) THEN
        h = FIX(h) + 1                  'round h to next whole number
      END IF
      IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
        h = h
      ELSE
        h = h - 1
      END IF

'
'     check if h satisfy deflection requirement
'
      IF h < hmin THEN
         h = FIX(hmin + 1)
         IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
           h = h
         ELSE
           h = h + 1
         END IF
         bw = FIX(.5 * (h - covpro))              'adjust bw
         IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
           bw = bw
         ELSE
           bw = bw + 1
         END IF
      END IF
'
'----------Calculate effective flange width-----------------------------
'
      b1 = L / 4
      b2 = bw + 16 * hf
      b3 = Beamspac
'
      IF b1 < = b2 THEN                 'get smallest b
        b = b1
      ELSE
        b = b2
      END IF
      IF b3 < = b AND b3 < > 0 THEN
        b = b3
      END IF
'
      d = h - covpro                    'new d
'
'----------Return parameter to disk file--------------------------------
'
```

```
    OPEN "DATA.DAT" FOR OUTPUT AS #1
    PRINT #1, "4"
    PRINT #1, "N"; b
    PRINT #1, "N"; bw
    PRINT #1, "N"; d
    PRINT #1, "N"; h
    CLOSE #1
'
END

'
'  Subroutine to compute self-weight moment
'
SUB MOMENT (SW, Boundary%, L, Musw) STATIC
'
     SHARED Support
     Ln = L - Support
     IF Boundary% = 1 THEN                'simply supported
       Musw = SW * L^2 / (8 * 12)
     ELSEIF Boundary% = 2 THEN             'both ends continuous
       Musw = SW * Ln^2 / (16 * 12)
     ELSEIF Boundary% = 3 THEN             'one end continuous
       Musw = SW * Ln^2 / (11 * 12)
     ELSEIF Boundary% = 4 THEN             'cantilever
       Musw = SW * L^2 / 24
     END IF
END SUB
```

C.6 Routine: POSREIN.BAS

```
'      ***************************************************************
'      *                POSREIN.BAS                                  *
'      *                                                             *
'      *     A routine to assist the expert system program           *
'      *     "BEAMDES" to select suitable positive moment            *
'      *     reinforcement for both the rectangular and tee          *
'      *     section.                                                *
'      ***************************************************************
'
'---------Variable Descriptions----------------------------------------
'
'     Ast  = Area of tension steel
'     Asprov = Area of steel provided
'     dreal = Revised d for two layer of steel
'     Mdes = Moment capacity of the section
'     db1 = Bar size one and two
'     Num1 = Number of bar of size one
'     BLA1 = Number of bottom layer bar of size one
'     TLA1 = Number of top layer bar of size one
'     Nolayer = Number layer the steel is placed into.
'     Cn   = Compressive force
'
      DEFINT I
      DECLARE SUB CHECK (db1, Num1, Asprov, dreal, Md, Nolayer, Status, BL1, TL1)
      DIM BAR.NUM(9), BAR.DIA(9), BAR.AREA(9), db(9), Num(9), Aspro(9)
      DIM d2(9), Md(9), Layer(9), BL1(9), TL1(9)
'
'---------Read parameters from a disk file------------------------------
'
      OPEN "DATA.DAT" FOR INPUT AS #1
      INPUT #1, TOTAL
      INPUT #1, Fc$
      INPUT #1, Fy$
      INPUT #1, b$
      INPUT #1, hf$
      INPUT #1, bw$
      INPUT #1, d$
      INPUT #1, Mu$
      INPUT #1, Musw$
      INPUT #1, cover$
      CLOSE #1
'
'     convert the string variables into real numbers
'
      Fc = VAL(RIGHT$(Fc$, LEN(Fc$) - 1))
      Fy = VAL(RIGHT$(Fy$, LEN(Fy$) - 1))
      b = VAL(RIGHT$(b$, LEN(b$) - 1))
      hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
      bw = VAL(RIGHT$(bw$, LEN(bw$) - 1))
      d = VAL(RIGHT$(d$, LEN(d$) - 1))
      Mu = VAL(RIGHT$(Mu$, LEN(Mu$) - 1))
      Musw = VAL(RIGHT$(Musw$, LEN(Musw$) - 1))
```

```
    cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
'
'----------Reading Data for Bar variables--------------------------------
'
'
'      read bar sizes, bar diameters and bar areas
'
    FOR I = 1 TO 9
        READ BAR.NUM(I), BAR.DIA(I), BAR.AREA(I)
    NEXT I
'
'      data statements
'
    DATA 3,0.375,0.11
    DATA 4,0.500,0.20
    DATA 5,0.625,0.31
    DATA 6,0.750,0.44
    DATA 7,0.875,0.60
    DATA 8,1.000,0.79
    DATA 9,1.128,1.00
    DATA 10,1.270,1.27
    DATA 11,1.410,1.56
'
'----------Calculate RhoBal-------------------------------------------
'
    IF Fc < 4 THEN
       Beta1 = .85
    ELSEIF Fc >= 4 AND Fc <= 8 THEN
       Beta1 = .85 - .05 * (Fc - 4)
    ELSE
       Beta1 = .65
    END IF
'
    RhoBal = (Beta1 * .85 * Fc / Fy) * (87 / (87 + Fy))
'
'----------Calculate Ast for rectangular section-------------------------
'
    IF b = 0 THEN
       Rho = .5 * RhoBal
       Ast = Rho * bw * d
    END IF
'
'----------First trail, estimate Ast for tee section---------------------
'
    Mtotal = Mu + Musw
    IF b <> 0 THEN
       Cn = .85 * Fc * b * hf
       Mn = Mtotal / .9              ' phi = 0.9
       jd = .9 * d                   ' moment arm
       Ast = Mn / (Fy * jd)
    END IF
'
'------------SELECT REINFORCEMENT---------------------------------------
'
   FOR I = 1 TO 9
     db(I) = BAR.NUM(I)
```

```
        Num(I) = FIX(Ast / BAR.AREA(I))
'
        CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
'
        IF Status = 6 OR Status = 3 THEN
           Num(I) = Num(I) - 1
           CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
        ELSEIF Status = 4 OR Status = 7 OR Status = 8 THEN
           Num(I) = Num(I) + 1
           CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
        END IF
'
        IF Status < > 0 THEN
           Num(I) = 0
        END IF
     NEXT I
'
'    get the provided minimum steel area
'
     FOR I = 1 TO 9
        IF Num(I) < > 0 THEN
           MinAspro = Aspro(I)
           EXIT FOR
        END IF
     NEXT I
     FOR I = 1 TO 9
        IF Num(I) < > 0 THEN
          IF Aspro(I) < = MinAspro THEN
             MinAspro = Aspro(I)
             db1 = db(I)
             Num1 = Num(I)
             Asprov = Aspro(I)
             dreal = d2(I)
             Mdes = Md(I)
             Nolayer = Layer(I)
             BLA1 = BL1(I)
             TLA1 = TL1(I)
          END IF
        END IF
     NEXT I
'
'-----------Return parameters to disk file-------------------------------
'
      OPEN "DATA.DAT" FOR OUTPUT AS #1
      PRINT #1, "9"
      PRINT #1, "N"; Num1
      PRINT #1, "N"; db1
      PRINT #1, "N"; Asprov
      PRINT #1, "N"; Ast
      PRINT #1, "N"; Mdes
      PRINT #1, "N"; dreal
      PRINT #1, "N"; Nolayer
      PRINT #1, "N"; BLA1
      PRINT #1, "N"; TLA1
```

```
    CLOSE #1
'
END

DEFSNG I
'
'   Subroutine to check reinforcement adequacy
'
SUB CHECK (db1, Num1, Asprov, dreal, Md, Nolayer, Status, BL1, TL1) STATIC
'
    DEFINT I
    SHARED BAR.NUM(), BAR.DIA(), BAR.AREA(), b, d, bw, hf
    SHARED Fy, Fc, Mtotal, cover, RhoBal, Beta1, Ast
'
    FOR I = 1 TO 9
       IF db1 = BAR.NUM(I) THEN
          dia1 = BAR.DIA(I)
          Area1 = BAR.AREA(I)
       END IF
    NEXT I
'
    Asprov = Num1 * Area1
'
    IF Status < > 0 THEN
       Status = 0
    END IF
'
'   check if there is two or more bars and less than 12 bars
'
    IF Num1 < 2 OR Num1 > 12 THEN
       Status = 1
       EXIT SUB
    END IF
'
'   check if the bar can be placed in one or two layer
'
    space = dia1
'
    IF space < = 1 THEN
       space = 1
    END IF
'
    breq = (Num1 * dia1) + (cover + .5) * 2 + (Num1 - 1) * space
'
    IF breq > bw THEN
       Nolayer = (breq + ((cover + .5) * 2) - space) / bw
       IF Nolayer > 2 THEN
          Status = 3
          EXIT SUB
       ELSE
          Nolayer = 2
          BL1 = FIX(((bw - (cover + .5) * 2) + space) / (dia1 + space))
          TL1 = Num1 - BL1
          IF BL1 < TL1 THEN
             Status = 3
             EXIT SUB
```

```
      END IF
      IF BL1 < 2 OR TL1 < 2 THEN
        Status = 4
        EXIT SUB
      END IF
    END IF
  ELSE
    Nolayer = 1
    BL1 = Num1
    TL1 = 0
  END IF
'
'  check steel percentage
'
  IF Nolayer = 2 THEN                  'reduce d for two layer of steel
    dreal = d - ((dia1 / 2) + .5)
  ELSEIF Nolayer = 1 THEN
    dreal = d
  END IF
'
  RhoMin = 200 / (Fy * 1000)
'
  IF b = 0 THEN                          'rectangular section
    Rho = Asprov / (bw * dreal)
    RhoMax = .75 * RhoBal
'
    IF Rho < = RhoMin THEN
      IF Asprov < = Ast / 3 THEN          'ACI 318-83 Sec. 10.5.2
        Status = 7
        EXIT SUB
      END IF
    END IF
  ELSE                               'tee section
    Rhof = .85 * Fc * (b - bw) * hf / (Fy * bw * dreal)
    RhoBal1 = (bw / b) * (RhoBal + Rhof)
    Rho = Asprov / (b * dreal)
    Rhow = Asprov / (bw * dreal)
    RhoMax = .75 * RhoBal1
'
    IF Rhow < = RhoMin THEN
      IF Asprov < = Ast / 3 THEN          'ACI 318-83 Sec. 10.5.2
        Status = 7
        EXIT SUB
      END IF
    END IF
  END IF
'
  IF Rho > RhoMax THEN
    Status = 6
    EXIT SUB
  END IF
'
'  check capacity of section
'
  IF b = 0 THEN                     'rectangular section
    a = Asprov * Fy / (.85 * Fc * bw)
```

```
      Md = .9 * Asprov * Fy * (dreal - (a / 2))
    ELSE                              'tee section
      w = Asprov * Fy / (b * dreal * Fc)
      a = 1.18 * w * dreal
'
      IF a < hf THEN                  'analyze as rectangular section
        a = Asprov * Fy / (.85 * Fc * b)
        Md = .9 * Asprov * Fy * (dreal - (a / 2))
      ELSE                            'analyze as tee section
        Asf = .85 * Fc * (b - bw) * hf / Fy
        a = (Asprov - Asf) * Fy / (.85 * Fc * bw)
        Mn1 = (Asprov - Asf) * Fy * (dreal - (a / 2))
        Mn2 = Asf * Fy * (dreal - (hf / 2))
        Md = .9 * (Mn1 + Mn2)
      END IF
    END IF

'
   IF Md < Mtotal THEN
      Status = 8
   END IF
'
END SUB
```

C.7 Routine: NEGREIN.BAS

```
'     ******************************************************************
'     *                   NEGREIN.BAS                           *
'     *                                                   *
'     *        A routine to assist the expert system program        *
'     *        "BEAMDES" to select suitable negative moment           *
'     *        reinforcement for the rectangular section at        *
'     *        the support.                                  *
'     ******************************************************************
'
'---------Variable Descriptions----------------------------------------
'
'     Ast  = Area of tension steel
'     Asp  = Area of compression steel
'     Asprov = Area of steel provided for positive moment
'     BLA1,BLA2 = Number of bottom layer bar of size one and two
'              respectively, for positive moment
'     db1 = Bar size one for tension steel
'     dbp1 = Bar size one for compression steel
'     Num1 = Number of tension bar of size one
'     Nump1,Nump2 = Number of compressive bar of size one and two
'
      DEFINT I
      DECLARE SUB CHECK (db1, Num1, Aspr, dreal, Md, Nolayer, Status, BL1, TL1)
      DIM BAR.NUM(9), BAR.DIA(9), BAR.AREA(9), db(9), Num(9), Aspro(9)
      DIM d2(9), Md(9), Layer(9), BL1(9), TL1(9)
'
'---------Read parameters from a disk file------------------------------
'
      OPEN "DATA.DAT" FOR INPUT AS #1
      INPUT #1, TOTAL
      INPUT #1, Fc$
      INPUT #1, Fy$
      INPUT #1, bw$
      INPUT #1, d$
      INPUT #1, h$
      INPUT #1, Mu$
      INPUT #1, Musw$
      INPUT #1, Asprov$
      INPUT #1, BLA1$
      INPUT #1, BLA2$
      INPUT #1, Nolaye$
      INPUT #1, dbp1$
      INPUT #1, dbp2$
      INPUT #1, cover$
      CLOSE #1
'
'     convert the string variables into real numbers
'
      Fc = VAL(RIGHT$(Fc$, LEN(Fc$) - 1))
      Fy = VAL(RIGHT$(Fy$, LEN(Fy$) - 1))
      bw = VAL(RIGHT$(bw$, LEN(bw$) - 1))
      d = VAL(RIGHT$(d$, LEN(d$) - 1))
```

```
h = VAL(RIGHT$(h$, LEN(h$) - 1))
Mu = VAL(RIGHT$(Mu$, LEN(Mu$) - 1))
Musw = VAL(RIGHT$(Musw$, LEN(Musw$) - 1))
Asprov = VAL(RIGHT$(Asprov$, LEN(Asprov$) - 1))
BLA1 = VAL(RIGHT$(BLA1$, LEN(BLA1$) - 1))
BLA2 = VAL(RIGHT$(BLA2$, LEN(BLA2$) - 1))
dbp1 = VAL(RIGHT$(dbp1$, LEN(dbp1$) - 1))
dbp2 = VAL(RIGHT$(dbp2$, LEN(dbp2$) - 1))
cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
'
'----------Reading Data for Bartable------------------------------------
'
'    read bar sizes, bar diameters and bar areas
'
    FOR I = 1 TO 9
        READ BAR.NUM(I), BAR.DIA(I), BAR.AREA(I)
    NEXT I
'
'    data statements
'
    DATA 3,0.375,0.11
    DATA 4,0.500,0.20
    DATA 5,0.625,0.31
    DATA 6,0.750,0.44
    DATA 7,0.875,0.60
    DATA 8,1.000,0.79
    DATA 9,1.128,1.00
    DATA 10,1.270,1.27
    DATA 11,1.410,1.56
'
'----------Estimate the number of bars extended from positive moment-----
'
    FOR I = 1 TO 9
      IF dbp1 = BAR.NUM(I) THEN
        Arep1 = BAR.AREA(I)
      END IF
      IF dbp2 = BAR.NUM(I) THEN
        Arep2 = BAR.AREA(I)
      END IF
    NEXT I
'
    IF (BLA1 + BLA2) > 2 THEN
      Asp = Asprov / 2              'assuming half of positive steel
      Nump1 = CINT(Asp / Arep1)
      IF Nump1 > BLA1 THEN
        Nump1 = BLA1
        Asrem = Asp - (BLA1 * Arep1)
        Nump2 = CINT(Asrem / Arep2)
        IF Nump2 > BLA2 THEN
          Nump2 = BLA2
        END IF
        IF Nump1 / 2 < > CINT(Nump1 / 2) AND Nump2 / 2 < > CINT(Nump2 / 2) THEN
          IF Nump2 < BLA2 THEN
            Nump2 = Nump2 + 1
          ELSE
            Nump1 = Nump1 - 1
```

```
        END IF
      END IF
    ELSEIF Nump1 < = BLA1 THEN
      IF Nump1 < = 1 AND BLA1 < 2 THEN
        Nump1 = 0
        Nump2 = CINT(Asp / Arep2)
        IF Nump2 > = BLA2 THEN
          Nump2 = BLA2
        END IF
      ELSEIF Nump1 < = 1 AND BLA1 > = 2 THEN
        Nump1 = 2
      END IF
    END IF
    IF BLA1 / 2 = CINT(BLA1 / 2) AND Nump1 > 0 THEN
      IF Nump1 / 2 < > CINT(Nump1 / 2) THEN
        Nump1 = Nump1 - 1
      END IF
    END IF
    IF BLA2 / 2 = CINT(BLA2 / 2) AND Nump2 > 0 THEN
      IF Nump2 / 2 < > CINT(Nump2 / 2) THEN
        Nump2 = Nump2 - 1
      END IF
    END IF
  ELSE
    Nump1 = BLA1
    Nump2 = 0
  END IF
'
  Asp = (Nump1 * Arep1) + (Nump2 * Arep2)      'real Asp
'
'----------Design for maximum negative moment----------------------------
'
  Mtotal = Mu + Musw
  Mn = Mtotal / .9                       'phi = 0.9
'
  dp = h - d
'
  Mn2 = Asp * Fy * (d - dp)         'assuming Asp yielded,i.e. fs=fy
  Mn1 = Mn - Mn2
'
  jd = .85 * d                  'moment arm
  Ast1 = Mn1 / (Fy * jd)
  Ast = Ast1 + Asp                 'Asp = Ast2
'
'------------SELECT REINFORCEMENT---------------------------------------
'
 FOR I = 1 TO 9
   db(I) = BAR.NUM(I)
   Num(I) = FIX(Ast / BAR.AREA(I))
'
   CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
'
   IF Status = 6 OR Status = 3 THEN
     Num(I) = Num(I) - 1
     CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
   ELSEIF Status = 4 OR Status = 7 OR Status = 8 THEN
```

```
            Num(I) = Num(I) + 1
            CALL CHECK(db(I), Num(I), Aspro(I), d2(I), Md(I), Layer(I), Status, BL1(I), TL1(I))
         END IF
'
         IF Status < > 0 THEN
            Num(I) = 0
         END IF
      NEXT I
'
'     get the provided minimum steel area
'
      FOR I = 1 TO 9
         IF Num(I) < > 0 THEN
            MinAspro = Aspro(I)
            EXIT FOR
         END IF
      NEXT I
      FOR I = 1 TO 9
         IF Num(I) < > 0 THEN
            IF Aspro(I) < = MinAspro THEN
               MinAspro = Aspro(I)
               db1 = db(I)
               Num1 = Num(I)
               Aspr = Aspro(I)
               dreal = d2(I)
               Mdes = Md(I)
               Nolayer = Layer(I)
               BLT1 = BL1(I)
               TLT1 = TL1(I)
            END IF
         END IF
      NEXT I
'
'-----------Return parameters to disk file-------------------------------
'
       OPEN "DATA.DAT" FOR OUTPUT AS #1
       PRINT #1, "13"
       PRINT #1, "N"; Num1
       PRINT #1, "N"; db1
       PRINT #1, "N"; Aspr
       PRINT #1, "N"; Ast
       PRINT #1, "N"; Mdes
       PRINT #1, "N"; dreal
       PRINT #1, "N"; dp
       PRINT #1, "N"; Nolayer
       PRINT #1, "N"; BLT1
       PRINT #1, "N"; TLT1
       PRINT #1, "N"; Nump1
       PRINT #1, "N"; Nump2
       PRINT #1, "N"; Asp
       CLOSE #1
'
END

DEFSNG I
'
```

```
'   Subroutine to check reinforcement adequacy
'
SUB CHECK (db1, Num1, Aspr, dreal, Md, Nolayer, Status, BL1, TL1) STATIC
'
    DEFINT I
    SHARED BAR.NUM(), BAR.DIA(), BAR.AREA(), d, dp, bw
    SHARED Fy, Fc, Mn, cover, Asp, Ast
'
    FOR I = 1 TO 9
      IF db1 = BAR.NUM(I) THEN
        dia1 = BAR.DIA(I)
        Area1 = BAR.AREA(I)
      END IF
    NEXT I
'
    Aspr = Num1 * Area1
'
    IF Status < > 0 THEN
      Status = 0
    END IF
'
'   check if there is two or more bars and less than 12 bars
'
    IF Num1 < 2 OR Num1 > 12 THEN
      Status = 1
      EXIT SUB
    END IF
'
'   check if the bar can be placed in one or two layer
'
    space = dia1
'
    IF space < = 1 THEN
      space = 1
    END IF
'
    breq = (Num1 * dia1) + (Num2 * dia2) + (cover + .5) * 2 + (Num1 - 1) * space
'
    IF breq > bw THEN
      Nolayer = (breq + ((cover + .5) * 2) - space) / bw
      IF Nolayer > 2 THEN
        Status = 3
        EXIT SUB
      ELSE
        Nolayer = 2
        BL1 = FIX(((bw - (cover + .5) * 2) + space) / (dia1 + space))
        TL1 = Num1 - BL1
        IF BL1 < TL1 THEN
          Status = 3
          EXIT SUB
        END IF
        IF BL1 < 2 OR TL1 < 2 THEN
          Status = 4
          EXIT SUB
        END IF
      END IF
```

```
    ELSE
      Nolayer = 1
      BL1 = Num1
      TL1 = 0
    END IF
'
'   check against minimum steel percentage
'
    IF Nolayer = 2 THEN                   'reduce d for two layer of steel
      dreal = d - ((dia1 / 2) + .5)
    ELSEIF Nolayer = 1 THEN
      dreal = d
    END IF
'
    IF Fc < 4 THEN
     Beta1 = .85
    ELSEIF Fc >= 4 AND Fc <= 8 THEN
     Beta1 = .85 - .05 * (Fc - 4)
    ELSE
     Beta1 = .65
    END IF
'
    RhoBal = Beta1 * .85 * Fc * 87 / (Fy * (87 + Fy))
    RhoMin = 200 / (Fy * 1000)
    Rho = Aspr / (bw * d)
    Rhop = Asp / (bw * d)
'
    IF Rho <= RhoMin THEN
     IF Aspr <= Ast / 3 THEN              'ACI 318-83 Sec. 10.5.2
       Status = 7
       EXIT SUB
     END IF
    END IF
'
'   check if compression steel had yielded as assumed
'
    RhoRhop = Beta1 * .85 * Fc * dp * 87 / (Fy * d * (87 - Fy))
'
    IF (Rho - Rhop) >= RhoRhop OR Rho = Rhop THEN
     Fsp = Fy                             'compression steel yielded
    ELSE
     Fsp = 87 * (1 - (.85 * Fc * Beta1 * dp / ((Rho - Rhop) * Fy * d)))
     IF Fsp > Fy THEN
       Fsp = Fy
     END IF
    END IF
'
'   check against maximum steel percentage
'
    RhoMax = (.75 * RhoBal) + (Rhop * Fsp / Fy)
    IF Rho > RhoMax THEN
     Status = 6
     EXIT SUB
    END IF
'
```

```
'     check capacity of section
'
      a = (Aspr * Fy - Asp * Fsp) / (.85 * Fc * bw)
      Md = .9 * (((Aspr * Fy - Asp * Fsp) * (dreal - (a / 2))) + (Asp * Fsp * (d - dp)))
'
      IF Md / .9 < Mn THEN
        Status = 8
      END IF
'
END SUB
```

C.8 Routine: ENLARGE.BAS

```
'     ************************************************************
'     *               ENLARGE.BAS                        *
'     *                                    *
'     *     This routine perform calculation to enlarge a       *
'     *     beam section.                             *
'     ************************************************************
'
'
      OPEN "DATA.DAT" FOR INPUT AS #1
      INPUT #1, Total
      INPUT #1, b$
      INPUT #1, bw$
      INPUT #1, d$
      INPUT #1, h$
      INPUT #1, hf$
      INPUT #1, L$
      INPUT #1, Beamspac$
      INPUT #1, hmin$
      INPUT #1, cover$
      CLOSE #1
'
      b = VAL(RIGHT$(b$, LEN(b$) - 1))
      bw = VAL(RIGHT$(bw$, LEN(bw$) - 1))
      d = VAL(RIGHT$(d$, LEN(d$) - 1))
      h = VAL(RIGHT$(h$, LEN(h$) - 1))
      hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
      L = VAL(RIGHT$(L$, LEN(L$) - 1))
      Beamspac = VAL(RIGHT$(Beamspac$, LEN(Beamspac$) - 1))
      hmin = VAL(RIGHT$(hmin$, LEN(hmin$) - 1))
      cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
'
      bw = bw + 1
'
'     choose a bw which is multiple of 2 or 3
'
      IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
        bw = bw
      ELSE
        bw = bw + 1
      END IF
'
      d = 2 * bw
'
      h = d + cover + 1.5
      IF h < > FIX(h) THEN
        h = FIX(h) + 1
      END IF
'
      IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
        h = h
      ELSE
        h = h - 1
```

```
END IF
IF h < hmin THEN
  h = FIX(hmin) + 1
  IF h / 2 = FIX(h / 2) OR h / 3 = FIX(h / 3) THEN
    h = h
  ELSE
    h = h + 1
  END IF
  bw = FIX(.5 * (h - cover - 1.5))          'adjust bw
  IF bw / 2 = FIX(bw / 2) OR bw / 3 = FIX(bw / 3) THEN
    bw = bw
  ELSE
    bw = bw + 1
  END IF
END IF
'
d = h - cover - 1.5
'
IF b < > 0 THEN
  b1 = L / 4
  b2 = bw + 16 * hf
  b3 = Beamspac
'
  IF b1 < = b2 THEN
    b = b1
  ELSE
    b = b2
  END IF
  IF b3 < > 0 AND b3 < = b THEN
    b = b3
  END IF
END IF
'
OPEN "DATA.DAT" FOR OUTPUT AS #1
PRINT #1, "4"
PRINT #1, "N"; b
PRINT #1, "N"; bw
PRINT #1, "N"; d
PRINT #1, "N"; h
CLOSE #1
'
END
```

C.9 Routine: BARCHAN.BAS

```
DECLARE SUB BARTABLE (BAR.NUM!(), BAR.DIA!(), BAR.AREA!())
'     ****************************************************************
'     *                    BARCHAN.BAS                         *
'     *                                               *
'     *     A routine to assist the expert system program          *
'     *     "BEAMDES" to let the user change reinforcement          *
'     *     if desired.                                  *
'     ****************************************************************
'
'---------Variable Descriptions----------------------------------------
'
'     Ast   = Area of tension steel required
'     Asprov = Area of tension steel provided
'     db1   = Bar size one
'     db2   = Bar size two
'     Num1  = Number of bar of size one
'     Num2  = Number of bar of size two
'
      DEFINT I
      DIM BAR.NUM(9), BAR.DIA(9), BAR.AREA(9)
'
'---------Read parameters from a disk file------------------------------
'
      OPEN "DATA.DAT" FOR INPUT AS #1
      INPUT #1, TOTAL
      INPUT #1, Asprov$
      INPUT #1, Ast$
      INPUT #1, Num1$
      INPUT #1, Num2$
      INPUT #1, db1$
      INPUT #1, db2$
      CLOSE #1
'
'     convert the string variables into real numbers
'
      Asprov = VAL(RIGHT$(Asprov$, LEN(Asprov$) - 1))
      Ast = VAL(RIGHT$(Ast$, LEN(Ast$) - 1))
      Num1 = VAL(RIGHT$(Num1$, LEN(Num1$) - 1))
      Num2 = VAL(RIGHT$(Num2$, LEN(Num2$) - 1))
      db1 = VAL(RIGHT$(db1$, LEN(db1$) - 1))
      db2 = VAL(RIGHT$(db2$, LEN(db2$) - 1))
'
'----------Reading Data for Bartable-------------------------------------
'
'
'       read bar sizes, bar diameters and bar areas
'
      FOR I = 1 TO 9
           READ BAR.NUM(I), BAR.DIA(I), BAR.AREA(I)
      NEXT I
'
```

```
'       data statements
'
      DATA 3,0.375,0.11
      DATA 4,0.500,0.20
      DATA 5,0.625,0.31
      DATA 6,0.750,0.44
      DATA 7,0.875,0.60
      DATA 8,1.000,0.79
      DATA 9,1.128,1.00
      DATA 10,1.270,1.27
      DATA 11,1.410,1.56
'
      COLOR 3: CLS
      LOCATE 5, 1
      PRINT "    Reinforcement selected is :"
      PRINT : PRINT "                              "; Num1; " # "; db1
      IF Num2 <> 0 THEN
      PRINT : PRINT "                              "; Num2; " # "; db2
      END IF
      PRINT : PRINT "                       Area  = "; Asprov; " sq. in"
      LOCATE 18, 1
      PRINT "    Strike Any Key TO Continue"
      ch$ = "": WHILE ch$ = "": ch$ = INKEY$: WEND
'
      CALL BARTABLE(BAR.NUM(), BAR.DIA(), BAR.AREA())
      COLOR 2
      PRINT "Area of steel required = "; Ast; " sq. in "
         PRINT
         INPUT "Enter Bar No. (combination of 2 sizes ?,?)  ", db1, db2
         INPUT "How many bar for each size.  (?,?)          ", Num1, Num2
         PRINT
'
      IF db1 < db2 THEN
        SWAP db1, db2
        SWAP Num1, Num2
      END IF
'
      FOR I = 1 TO 9
        IF db1 = BAR.NUM(I) THEN
          Area1 = BAR.AREA(I)
        END IF
        IF db2 = BAR.NUM(I) THEN
          Area2 = BAR.AREA(I)
        END IF
      NEXT I
'
      Asprov = Num1 * Area1 + Num2 * Area2
'
      OPEN "DATA.DAT" FOR OUTPUT AS #1
      PRINT #1, "5"
      PRINT #1, "N"; Num1
      PRINT #1, "N"; db1
      PRINT #1, "N"; Num2
      PRINT #1, "N"; db2
      PRINT #1, "N"; Asprov
```

```
    CLOSE #1
'
END

'
'
SUB BARTABLE (BAR.NUM(), BAR.DIA(), BAR.AREA()) STATIC
    '
    CLS
    COLOR 3
    LOCATE 2, 30
    PRINT "TABLE OF STEEL AREAS"
    PRINT STRING$(80, 45)
    LOCATE 4, 1
    PRINT "BAR"; TAB(7); "DIA."; TAB(41); "AREA OF BARS"
    LOCATE 5, 1
    PRINT "NO."; TAB(7); "(in)"; TAB(15); "1"; TAB(22); "2"; TAB(29); "3";
;TAB(36); "4"; TAB(43); "5"; TAB(50); "6"; TAB(57); "7"; TAB(64); "8";
;TAB(71); "9"; TAB(78); "10"
    PRINT STRING$(80, 45)
    FOR I = 1 TO 9
        PRINT USING "## "; BAR.NUM(I);
        PRINT USING " #.### "; BAR.DIA(I);
        FOR J = 1 TO 10
            PRINT USING "  ##.##"; J * BAR.AREA(I);
        NEXT J
        PRINT
    NEXT I
    PRINT STRING$(80, 45)
END SUB
```

C.10 Routine: BARCHECK.BAS

```
'     ***************************************************************
'     *                   BARCHECK.BAS .                    *
'     *                                         *
'     *      A routine to assist the expert system program      *
'     *      "BEAMDES" to check if the selected reinforcement      *
'     *      is adequate.                                *
'     ***************************************************************
'
'---------Variable Descriptions-----------------------------------------
'
'     Asprov = Area of steel provided
'     dreal = Revised d for two layer of steel
'     Md    = Moment capacity of the section
'     db1,db2 = Bar size one and two respectively
'     Num1,Num2 = Number of bar of size one and two respectively
'     BL1,BL2 = Number of bottom layer bar of size one and two respectively
'     TL1,TL2 = Number of top layer bar of size one and two respectively
'     Nolayer = Number layer the steel is placed into.
'
      DEFINT I
      DIM BAR.NUM(9), BAR.DIA(9), BAR.AREA(9)
'
'---------Read parameters from a disk file------------------------------
'
      OPEN "DATA.DAT" FOR INPUT AS #1
      INPUT #1, TOTAL
      INPUT #1, Fc$
      INPUT #1, Fy$
      INPUT #1, b$
      INPUT #1, bw$
      INPUT #1, d$
      INPUT #1, dp$
      INPUT #1, hf$
      INPUT #1, cover$
      INPUT #1, Mu$
      INPUT #1, Musw$
      INPUT #1, Asprov$
      INPUT #1, Asp$
      INPUT #1, Ast$
      INPUT #1, Num1$
      INPUT #1, Num2$
      INPUT #1, db1$
      INPUT #1, db2$
      INPUT #1, Section$
      CLOSE #1
'
'     convert the string variables into real numbers
'
      Fc = VAL(RIGHT$(Fc$, LEN(Fc$) - 1))
      Fy = VAL(RIGHT$(Fy$, LEN(Fy$) - 1))
      b = VAL(RIGHT$(b$, LEN(b$) - 1))
      bw = VAL(RIGHT$(bw$, LEN(bw$) - 1))
```

```
d = VAL(RIGHT$(d$, LEN(d$) - 1))
dp = VAL(RIGHT$(dp$, LEN(dp$) - 1))
hf = VAL(RIGHT$(hf$, LEN(hf$) - 1))
cover = VAL(RIGHT$(cover$, LEN(cover$) - 1))
Mu = VAL(RIGHT$(Mu$, LEN(Mu$) - 1))
Musw = VAL(RIGHT$(Musw$, LEN(Musw$) - 1))
Asprov = VAL(RIGHT$(Asprov$, LEN(Asprov$) - 1))
Asp = VAL(RIGHT$(Asp$, LEN(Asp$) - 1))
Ast = VAL(RIGHT$(Ast$, LEN(Ast$) - 1))
Num1 = VAL(RIGHT$(Num1$, LEN(Num1$) - 1))
Num2 = VAL(RIGHT$(Num2$, LEN(Num2$) - 1))
db1 = VAL(RIGHT$(db1$, LEN(db1$) - 1))
db2 = VAL(RIGHT$(db2$, LEN(db2$) - 1))
Section$ = RIGHT$(Section$, LEN(Section$) - 1)
'
'----------Reading Data for Bar variables--------------------------------
'
'
'     read bar sizes, bar diameters and bar areas
'
    FOR I = 1 TO 9
        READ BAR.NUM(I), BAR.DIA(I), BAR.AREA(I)
    NEXT I
'
'     data statements
'
    DATA 3,0.375,0.11
    DATA 4,0.500,0.20
    DATA 5,0.625,0.31
    DATA 6,0.750,0.44
    DATA 7,0.875,0.60
    DATA 8,1.000,0.79
    DATA 9,1.128,1.00
    DATA 10,1.270,1.27
    DATA 11,1.410,1.56
'
'----------Calculate RhoBal--------------------------------------------
'
    IF Fc < 4 THEN
       Beta1 = .85
    ELSEIF Fc > = 4 AND Fc < = 8 THEN
       Beta1 = .85 - .05 * (Fc - 4)
    ELSE
       Beta1 = .65
    END IF
'
    RhoBal = (Beta1 * .85 * Fc / Fy) * (87 / (87 + Fy))
'
   FOR I = 1 TO 9
     IF db1 = BAR.NUM(I) THEN
       dia1 = BAR.DIA(I)
       Area1 = BAR.AREA(I)
     END IF
     IF db2 = BAR.NUM(I) THEN
       dia2 = BAR.DIA(I)
       Area2 = BAR.AREA(I)
```

```
      END IF
    NEXT I
'
    Asprov = Num1 * Area1 + Num2 * Area2
'
'   check if there is two or more bars and less than 12 bars
'
    IF (Num1 + Num2) < 2 OR (Num1 + Num2) > 12 THEN
      Status% = 1
      GOTO SENDATA
    END IF
'
'   check if the difference in bar sizes is greater than 2
'
    IF Num1 < > 0 AND Num2 < > 0 THEN
      IF ABS(db1 - db2) > 2 THEN
        Status% = 2
        GOTO SENDATA
      END IF
    END IF
'
'   check if the bar can be placed symmetrically in one or two layer
'
    space = dia1
'
    IF space < = 1 THEN
      space = 1
    END IF
'
    Num = Num1 + Num2
    breq = (Num1 * dia1) + (Num2 * dia2) + (cover + .5) * 2 + (Num - 1) * space
'
    IF breq > bw THEN
      Nolayer = (breq + ((cover + .5) * 2) - space) / bw
      IF Nolayer > 2 THEN
        Status% = 3
        GOTO SENDATA
      ELSE
        Nolayer = 2
        BL1 = FIX(((bw - (cover + .5) * 2) + space) / (dia1 + space))
        IF Num1 > = BL1 THEN
          BL2 = 0
          TL1 = Num1 - BL1
          TL2 = Num2
        ELSE
          BL1 = Num1
          brem = bw - ((BL1 * dia1) + ((cover + .5) * 2) + ((BL1 - 1) * space))
          BL2 = FIX(brem / (dia2 + space))
          TL1 = 0
          TL2 = Num2 - BL2
        END IF
        IF BL1 < TL1 THEN
          Status% = 3
          GOTO SENDATA
        END IF
        IF (BL1 + BL2) < 2 OR (TL1 + TL2) < 2 THEN
```

```
        Status% = 4
        GOTO SENDATA
      END IF
      IF BL1 / 2 < > CINT(BL1 / 2) AND BL2 / 2 < > CINT(BL2 / 2) THEN
        Status% = 4
        GOTO SENDATA
      END IF
      IF TL1 / 2 < > CINT(TL1 / 2) AND TL2 / 2 < > CINT(TL2 / 2) THEN
        Status% = 4
        GOTO SENDATA
      END IF
    END IF
  ELSE
    Nolayer = 1
    BL1 = Num1
    BL2 = Num2
    TL1 = 0
    TL2 = 0
    IF Num1 / 2 < > CINT(Num1 / 2) AND Num2 / 2 < > CINT(Num2 / 2) THEN
      Status% = 5
      GOTO SENDATA
    END IF
  END IF
'
'  check steel percentage
'
  IF Nolayer = 2 THEN                'reduce d for two layer of steel
    dreal = d - ((dia1 / 2) + .5)
  ELSEIF Nolayer = 1 THEN
    dreal = d
  END IF
'
  RhoMin = 200 / (Fy * 1000)
'
  IF Section$ < > "DOUBLE" THEN            'singly reinforced section
   IF b = 0 THEN                      'rectangular section
     Rho = Asprov / (bw * dreal)
     RhoMax = .75 * RhoBal
'
     IF Rho < = RhoMin THEN
       IF Asprov < = Ast / 3 THEN     'ACI 318 - 83 Sec. 10.5.2
         Status% = 7
         GOTO SENDATA
       END IF
     END IF
   ELSE                               'tee section
     Rhof = .85 * Fc * (b - bw) * hf / (Fy * bw * dreal)
     RhoBal1 = (bw / b) * (RhoBal + Rhof)
     Rho = Asprov / (b * dreal)
     Rhow = Asprov / (bw * dreal)
     RhoMax = .75 * RhoBal1
'
     IF Rhow < = RhoMin THEN
       IF Asprov < = Ast / 3 THEN     'ACI 318 - 83 Sec. 10.5.2
         Status% = 7
         GOTO SENDATA
```

```
        END IF
      END IF
    END IF
'
    IF Rho > RhoMax THEN
      Status% = 6
      GOTO SENDATA
    END IF
  ELSE          'doubly reinforced section
'
'   check against minimum steel percentage
'
    Rho = Asprov / (bw * d)
    Rhop = Asp / (bw * d)
'
    IF Rho < = RhoMin THEN
      IF Asprov < = Ast / 3 THEN          'ACI 318 - 83 Sec. 10.5.2
        Status% = 7
        GOTO SENDATA
      END IF
    END IF
'
'   check if compression steel had yielded as assumed
'
    RhoRhop = Beta1 * .85 * Fc * dp * 87 / (Fy * d * (87 - Fy))
'
    IF (Rho - Rhop) > = RhoRhop THEN
      Fsp = Fy                    'compression steel yielded
    ELSE
      Fsp = 87 * (1 - (.85 * Fc * Beta1 * dp / ((Rho - Rhop) * Fy * d)))
      IF Fsp > Fy THEN
        Fsp = Fy
      END IF
    END IF
'
'   check against maximum steel percentage
'
    RhoMax = (.75 * RhoBal) + (Rhop * Fsp / Fy)
    IF Rho > RhoMax THEN
      Status% = 6
      GOTO SENDATA
    END IF
  END IF
'
'   check capacity of section
'
  IF Section$ < > "DOUBLE" THEN           'singly reinforced section
    IF b = 0 THEN                      'rectangular section
      a = Asprov * Fy / (.85 * Fc * bw)
      Md = .9 * Asprov * Fy * (dreal - (a / 2))
    ELSE                               'tee section
      w = Asprov * Fy / (b * dreal * Fc)
      a = 1.18 * w * dreal
'
      IF a < hf THEN                  'analyze as rectangular section
        a = Asprov * Fy / (.85 * Fc * b)
```

```
          Md = .9 * Asprov * Fy * (dreal - (a / 2))
        ELSE                              'analyze as tee section
          Asf = .85 * Fc * (b - bw) * hf / Fy
          a = (Asprov - Asf) * Fy / (.85 * Fc * bw)
          Mn1 = (Asprov - Asf) * Fy * (dreal - (a / 2))
          Mn2 = Asf * Fy * (dreal - (hf / 2))
          Md = .9 * (Mn1 + Mn2)
        END IF
      END IF
    ELSE              'doubly reinforced section
     a = (Asprov * Fy - Asp * Fsp) / (.85 * Fc * bw)
     Md = .9 * (((Asprov * Fy - Asp * Fsp) * (dreal - (a / 2))) + (Asp * Fsp * (d - dp)))
    END IF
'
    Mtotal = Mu + Musw
    IF Md < Mtotal THEN
      Status% = 8
      GOTO SENDATA
    END IF
'
'---------Return parameters to disk file---------------------------------
'
SENDATA:
'
    OPEN "DATA.DAT" FOR OUTPUT AS #1
    PRINT #1, "8"
    PRINT #1, "N"; Md
    PRINT #1, "N"; dreal
    PRINT #1, "N"; Nolayer
    PRINT #1, "N"; BL1
    PRINT #1, "N"; BL2
    PRINT #1, "N"; TL1
    PRINT #1, "N"; TL2
    IF Status% = 0 THEN
      PRINT #1, "C OK"
    ELSE
      PRINT #1, "C"; Status%
    END IF
    CLOSE #1
'
    END
```

The vita has been removed from the scanned document