ARTIFICIAL INTELLIGENCE BASED VISUAL ASSESSMENT: AN EXPLANATION SYSTEM FOR LANDSCAPE AESTHETIC

by

DAN ZHOU

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

MASTER OF SCIENCE

in

Forestry

APPROVED:

Gregory J. Buhyoff, Chairman

Patrick A. Miller

John W. Roach

July, 1992
Blacksburg, VA

ARTIFICIAL INTELLIGENCE BASED VISUAL ASSESSMENT: AN EXPLANATION SYSTEM FOR LANDSCAPE AESTHETICS

(ABSTRACT)

A rule-based expert system was developed to evaluate landscape quality from the perspective of a landscape designer/architect. The rationale for the development of such a system is that it can provide explanations for final assessments and retain this information during the reasoning process. The system also provides for systematic consideration of a broad range of variables with complex interrelationships. The entire system is composed of four subsystems programmed in VPI PROLOG. These subsystems are each separate expert systems for the assessment of man-made features, natural features, spatial organization and visual composition. The prototype of the system has been preliminarily tested and the results of the evaluation of selected landscape photographs is encouraging. The visual quality determination along with the explicit explanations, which are the translation and interpretation of the expert rules used in the reasoning process, were shown to be important in achieving a better understanding and evaluation of the landscape visual quality.

ACKNOWLEDGEMENTS

First and foremost, thanks go to my graduate committee members, Drs. Gregory J. Buhyoff, Patrick A. Miller and John W. Roach, for their thoughtful ideas and constructive comments. Dr. Miller spent countless hours elaborating and developing decision rules as the focus of the knowledge engineering process. His dedication is deeply appreciated. Dr. Roach provided computer science expertise, guidance and dealt with many programming and development queries. Finally, I can not thank Dr. Gregory J. Buhyoff and his "red pens" enough. His intelligent direction and encouragement have supported me through all my graduate life at Virginia Polytechnic Institute and State University.

This research was supported by the United States Forest Service, North Central Forest Experiment station, Chicago, Illinois through Cooperative Research Agreements 23-89-15 and 23-89-17. Their support is gratefully acknowledged. In particular, I would like to give special thanks to Dr. Herbert Schroeder whose interest and moral support of this research was a critical element in its accomplishment.

My deepest dept is to my parents. Ten thousand miles away across the Pacific ocean, I can always feel their everpresent love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

There has been an increasing public concern for preserving the beauty of natural environments in the United States. Several laws and regulations, such as The National Environmental Policy Act of 1969, have been passed in the last three decades which have decreed that visual attributes of natural environments must be considered in environmental protection efforts. These concerns and legislation have spurred the development of methods to evaluate scenic visual quality, which is an important step toward a better understanding and management of the visual attributes of landscapes.

The psychophysical modeling approach has been a major research direction for attempting to evaluate scenic visual quality. However, since scenic visual quality is neither inherent in the landscape nor "made up" of purely subjective factors, psychophysical models may not be able to explain all the variability inherent in the process of human perception of scenic visual quality.

Lacking an explanation for a scenic quality evaluation sometimes can greatly weaken the power of the evaluation of a landscape. For example, a

conclusion in a statistical evaluation system with only a number as the result can often make users feel difficult to understand the real meaning of that number. In a rule based system, without knowing how a conclusion is generated and where it comes from, users may also be frustrated facing a computer generated complex conclusion. Adding explanations to a conclusion of the evaluation can potentially improve the "acceptance" of a scenic quality assessment method.

This research deals with the problem of scenic visual quality evaluation by way of artificial intelligence applications in order to improve the explanatory ability of scenic quality assessments. Specifically, this investigation focuses on developing a rule based system that can be used to collect visual quality data, process scenic visual quality assessment information and generate evaluations of scenic visual quality from the standpoint of an expert landscape designer. This research explores the potential for artificial intelligence as a methodological basis for developing better and more comprehensive scenic visual quality evaluations.

The second chapter reviews the previous work done so far. The design assessment approaches, psychophysical models, expert type assessments and potential computer science applications are the main areas reviewed. Chapter

3 discusses the methodologies in the research which includes the rule based system, explanation mechanism, explanation visual management system and so on. Chapter 4 presents and discusses the results of this research. It describes the six program components and four subsystems. Finally, it talks about the preliminary subsystem testing and gives the results of the testing. Chapter 5 concludes the entire thesis by discussing the advantages and potential disadvantages of the system. At the end, recommendations for the further research are given.

CHAPTER II

PREVIOUS WORK

The major impetus for systematic analysis and study of scenic visual quality occurred during the decades of the 1960's and 1970's. Legislation was enacted during this time that directed attention to the identification and management of scenic resources. Policy directives and mandates during these two decades, together with an increasing scientific interest in perception and management of scenic visual quality, provided stimulus for considerable research and study about landscape values and methods for assessing values.

Public concerns also require that scenic visual quality and other intangible products of public land be considered. Scenic visual quality is an important natural resource. Among the many natural resources, scenic visual quality has proven one of the most difficult to manage since visual quality is defined by both characteristics of the environment and human judgement. It is therefore a subjective attribute which, if possible, must be dealt with objectively.

As the chief of United States Forest Service, John McGuire (1979) pointed out:" I believe that by managing the visual resource in integration with

other resources on the National forest, there is a broad range of forest management practices which can be made visually acceptable to the public, and which may in several instances actually enhance both the quantity and quality of the resource values produced" (pp 18).

Correct and efficient measurement or assessment of scenic quality is vital to management of landscapes. Daniel (1976) notes three reasons for determining relative landscape scenic quality:

1) Better integration with other resources and products;
2) Better justification for land use decisions;
3) Restoration of the client-architect relationship.

To manage landscapes efficiently, it is important to have a comprehensive and correct evaluation of landscapes. It is not easy to have a comprehensive evaluation of landscapes because there are too many various factors affecting the evaluation of landscapes. Nor is it easy to have a correct evaluation of landscapes since the perception of landscape scenic quality is a function of the interaction of humans and the landscape (Zube et al., 1975; Zube et al., 1982).

The human concepts involved in visual landscape appreciation include the implicit and explicit assumptions about the nature of humans, or some features of humans that play a role in a particular interaction with a landscape. Past experience, knowledge, expectations and the cultural context of individuals and groups are all considered part of the human component of landscape appreciation or scenic visual quality judgement. Landscape properties on the other hand refer to the tangible and intangible elements of the landscape themselves which are influential in the human visual assessment process. Landscape components are the physical elements and features* of a scene which taken together make up the landscape entity or whole. Interaction outcome is the product that arises from the human and landscape perceptual interaction, which may be both tangible or intangible.

Design Assessment Approaches

The most widely used and probably the most successful descriptive inventory methods have been developed by landscape architects. The pioneering work of Litton (1972) may be credited for much of the impetus in

---

* Elements are usually line form, color and texture and so on. Features are mountains, trees and so on.

the development of this approach. Litton based his method primarily on the identification and evaluation of landscape types. Litton proposed that the managerial objectives of landscapes are preservation, protection (maintenance), enhancement and restoration. Some similar managerial objectives were also proposed by the United States Forest Service. They are preservation, retention, partial retention, modification and maximum modification (USDA, Forest Service, 1974)

One of the most basic problems in assessing landscape scenic visual quality is understanding what to assess and understanding what environmental attributes cause variation in visual quality judgements. Litton (1972) identified six factors that he felt were basic to the human judgement process in regarding appreciation. These six recognition factors are form, space, time variability, observing position, observing distance and observing sequence. The first three are called primary factors which are generally beyond the capacity of the observer to change while the last three describe relationships between observer and landscapes. Furthermore, according to these recognition factors and their combinations, it becomes possible to identify the following six landscape types:

1) panoramic landscape;

2) feature landscape;

3) enclosed landscape;

4) focal landscape;

5) forest landscape, and

6) detailed landscape.

In the light of these factors and landscape types, it is theoretically possible for landscape managers to come to some agreement about what landscape resources consist of and to document and evaluate landscape resources in a more rational way. Both primary and secondary groups of recognition factors reinforce one another and offer us greater choice in how we wish to approach aesthetic analysis of the landscape.

Psychological Predictors

Another way to characterize the landscape visual environment may be through dimensional analyses of people's preference for different landscapes (Kaplan et al., 1982). Based on their studies and the studies of their students the Kaplans have developed a theory about landscapes that people prefer. They believe people react to the content and the spatial organization of the landscape. In terms of spatial organization, "complexity", "mystery", "legibility" and "coherence" are important attributes. They use dimensional analyses of

landscape preferences to identify interpretable groupings of landscape characteristics that isolated meaningful content domains. Generally, they have found that coherence does not play a major role in the prediction of visual preference. On the other hand, mystery has been found to be a strong positive predictor of scenic preference.

Of course, many other concepts related to visual field have also been used to define landscape attributes. These other elements might include variety, contrast, texture and harmony; among others of these, variety has received much attention. Variety has many facets: complexity, change, uniqueness, number of edges and so on. All of these are probably dimensions of variety. Conceptually, complexity is related to variety in that the more complex an area is, the greater variety the area has. (Arthur et al., 1977).

Studies have demonstrated that variety can have both positive and negative effects. Its positive valuation can be explained in psychological theory (Wohlwill, 1966). When landscapes are divided into urban and rural categories, preference within categories generally increases with complexity (Kaplan et al., 1972). Gratzer and McDowell (1971) have also noticed that observers tend to pay attention to areas of change in landscapes. Too much change or variety however can result in negative responses by evaluators.

On the other hand, some other studies have failed to show a systematic relationship between variety and scenic visual quality preference (Wohlwill, 1968; Zube et al, 1974). This suggests that variety as an exclusive concept for landscape analysis may be insufficient. Further identification and analysis of the context of variety and elements comprising it may be necessary. Kaplan et al. (1972) have shown that complexity by itself does not precisely predict landscape preference. Nevertheless, complexity can predicate visual preferences quite well if landscapes are first divided into urban and rural categories.

Jacques (1980) commented on attempts to try and make objective predictions of landscape scenic visual quality for different social groups. He concluded that landscape evaluation was entirely subjective and therefore the first step of landscape evaluation should start from an identification of cultural groups and their tastes. In his studies, Jacques emphasized subjective variables such as history, culture, personal experience and taste. He felt that landscape observers use their own values as the source of beauty, not the "intrinsic qualities" of the landscape.

Daniel and Boster (1976) found that cattle and range interest groups prefer open and grassy landscape to densely forested area. On the other hand,

university students apparently found that the opposite is more beautiful. Peterson and Neumann (1969) identified two cohesive sub-samples having distinctly different scenic preferences and Zube et al. (1974) observed that systematic differences of scenic visual quality evaluation existed between inner-city residents and other respondents.

In contrast to these findings, substantial agreement is often noted when quantitative scenic visual quality assessment methods are employed and mean ratings are calculated for large groups of respondents. Craik (1972) reported that high agreement in landscape scenic visual quality evaluations was found between professionals and non-professionals. Uniformity of land-managers' and users' attitudes has also been demonstrated by some verbal surveys (Clark et al. 1971). Daniel and Boster (1976) found a high degree of consensus among 27 groups on evaluation of many types of forest silvicultural treatments. Other research has also indicated that various socioeconomic, age, geographical groups have not shown large differences between groups in the evaluation on landscape visual quality. For example, high correlations between scenic quality metrics exist for college students, landscape architect graduate students, black youths and middle aged white males ($r = .92$)(Wellman et al., 1980); college students and civic groups ($r = .91$)(Schroeder et al., 1983).

Psychophysical Models

In an attempt to determine the relationships between physical characteristics of landscapes and perceptual judgements of human observers, psychophysical models and methods have been developed. Originally, psychophysical research was used to "measure" the perception of stimulus attributes such as size, weight and brightness. It was then recognized that scenic visual quality evaluations might be somewhat similar to these perceptual processes (Arthur et al., 1977). So, psychophysical methods were used in an attempt to predict scenic quality. With these methods observers indicate their preferences for various visual stimuli of landscapes by assigning numerical ratings to the stimuli, or by selecting the preferred stimulus from a pair or group of stimuli. After each area is assigned a numerical index of preference, these indices are then standardized, adjusted to remove scale weighing biases and then manipulated to obtain means and variances for each stimulus (Daniel et al., 1976). Tangible variables of forest stand such as Diameter at Breast Height (DBH), Basal Area (BA), Tree Per Acre (TPA) and so on. which can be directly measured are then statistically related to the scenic quality judgements. Multiple linear regression has been the most used technique to determine these relationships. Studies (Buhyoff et al., 1986) have shown that psychophysical models are quite reliable and sensitive to subtle landscape variations. Also,

these psychophysical models have been proven to provide good assessments of public perceptions of the relative scenic quality differences between landscapes within a given set (Daniel et al., 1984).

One of the most popular psychophysical methods is the Scenic Beauty Estimation Method (SBE)(Daniel and Boster, 1976). In the SBE method, an observer may express his or her appreciation of scenic visual quality by a numerical rating, e.g. 1 (very low) to 10 (very high). Different observers may use the rating scales differently. Differences in ratings may indicate true differences in perceived scenic visual quality or may not. In SBE, the ambiguity of differences in observers' criterion values can be eliminated by providing measures of scenic visual quality independent of observer appraisal scale. Actually, scenic visual quality is determined neither entirely from characteristics of landscape, nor by stated preferences of observers' ratings. An SBE, then, is a quantitative index of the perceived scenic visual quality adjusted for the different rating scales used by different subjects.

Another widely used and reliable psychophysical scenic visual quality assessment method is based on the Law of Comparative Judgement (Buhyoff et al., 1978; Buhyoff et al. 1980). The Law of Comparative Judgement (LCJ) asks a respondent to make comparative judgements among all possible pairs of

landscapes to determine the degree of confusion which is represented by the proportion of times one landscape is selected over the other. Discrimination between landscapes will be easily confused if they are very similar. There are two methods of applying comparative judgements in LCJ, i.e. pair comparison and rank ordering. There is no major difference between them and the results are similar (Hull et al., 1981).

These two most widely used and validated psychophysical scenic visual quality assessment approaches (SBE and LCJ) have been carefully and thoroughly compared from theoretical and practical points of view. It has been demonstrated that their results are identical for the most part (Hull et al., 1984).

Many psychophysical models have been developed to estimate, evaluate and predict landscape scenic visual quality. Some statistical models that predict scenic visual quality from an in-stand perspective were developed for southern pine stands. Results indicate that stand age, average stand diameter, and stand stocking density are positively related to scenic quality (Buhyoff, et al. 1986). In other research, Hull and Buhyoff (1986) developed a method to simulate and evaluate temporal distributions of scenic visual quality, which represent the level of scenic visual quality at each year during a stand planning horizon. In

general, they found that decreasing stand density increases scenic visual quality, less productive sites have more scenic visual quality, and increasing stand age increases scenic visual quality. Other psychophysical scenic quality models (Lien et al., 1986) were formulated for a random sample of residential streets in Ann Arbor, Michigan and indicated that an increasing average tree diameter increases visual quality in logarithmic fashion.

As is evidenced by the numerous studies in the literature, predicting perceived scenic visual quality is possible. Broad issues, however, remain unresolved: 1) identifying the "correct" predictor variables of preference is problematic since there is no underlying theory to identify logical predictors; 2) the models provide no explicit information as to "why" they are good models; 3) surrogate measures are used as predictors for underlying psychological constructs which are likely the "real" causes of visual preferences; 4) the models are based upon measurement of public preferences and do not consider expert design assessment rules and concepts; and 5) these models do little in providing important theoretical insights.

Psychophysical models most often use physical measures of landscape features to predict scenic quality. Statistical methods are often used to develop regression equations which are mathematically complex and defy simple, logical

explanation as to "why" they may be valid prediction models. Thus, they can be viewed as "black-box" formulations with no underlying rational for their behavior. Since most psychophysical models utilize surrogate measures for predictors (as explained above) they are theoretically further from identifying true cause/effect variables and are then more difficult to rationalize to potential users of such models. Compounding the problem of "explaining" the foundation and behavior of such formulations is the desirability to keep such models simple (i.e., use as few variables as possible) in an attempt to avoid the fitting of spurious relationships. Therefore, inherently, the process of developing psychophysical models limits their potential for being comprehensive landscape quality prediction and assessment devices. On the average, these models generally account for only 50 - 70 percent of the variability in preference. This is indicative, then, of the presence of other, important underlying causes of personal visual preferences. The structure of the models is therefore a limiting factor to their potential power and explanatory value.

In summary, scenic quality is a complex psychophysical/perception process, and traditional statistical psychophysical models cannot provide much insight into the complexity of these perceptions since underlying theoretical explanation is absent, or minimally, extremely difficult to determine.

Expert Type Assessments

The other broad approach to the evaluation/assessment of visual resources is the expert assessment type of evaluation. This method is typified by the U.S. Forest Service's Visual Management System (VMS used here to classify a general set of approaches exemplified by the U.S.F.S (1974) VMS). There are certain advantages to such an approach over a statistical modeling method. VMS's are straightforward systems which use intuitive constructs and obvious physical landscape attributes to arrive at landscape classification decisions. Therefore, they may have a certain appeal since the "language" used by the system is nonmathematical and more readily explainable to the non-expert. In addition, the problems of sampling, model construction, and metric development and evaluation which characterize psychophysical approaches are non-existent with VMS approaches. Finally, and likely one of the most important positive qualities of these methods, is that these systems utilize classifications (i.e., visual retention, partial modification, modification, etc.) which provide important, if only minimal, management criteria and guidelines which are at least somewhat related to real policy stipulations. However, VMS's are actually *inventory systems and not strictly assessment systems*. The assessment part of the process which involves the underlying reasoning for determining landscape classifications is lost in the process itself and as

information is collapsed into smaller and smaller ranges of landscape attribute categories. There are other inherent problems or limitations to this VMS type approach. Such systems assume that expert can speak for public concerns when some research indicates that they may carry their personal biases into the evaluation process (Buhyoff et al., 1978). Unfortunately, the intended user of VMS's are the experts themselves since the systems are based in the lexicon developed in formal landscape design training. Therefore, in the absence of an expert to use the system, a VMS may not actually be useful to other non-experts. Traditionally, just as with psychophysical models, there is no explanatory ability to the conclusions. The underlying explanations are lost in the process of narrowing decision categories, and the multiple and complex expert determinations and decisions are not easily recaptured to provide reasoning and rationale for final classification decisions. Finally, mostly visual managements are done as part of managing other resources. Most of the time, these management decisions require more than a scenic value rating. The explanation can be helpful in managing the other resource without doing unintentional harm to scenic resources.

At this time, however, there is no single model that can deal with all types and classes of landscapes. For different applications, one method may be more appropriate than others. Briggs and France (1980) compared four

different methods of evaluating landscapes: arithmetic aggregation, multiple regression analysis, statistical classification and intuitive classification. They found that

1) arithmetic aggregation is not a satisfactory method to assess landscape visual quality;
2) the need for a preliminary survey of landscape features seems to limit the applicability of multiple regression technique;
3) statistical classifications have the advantage of flexibility and provide a useful service by identifying areas of similar character as well as quality; and,
4) intuitive classifications are likely to be effective when a simple, quick and cheap method is required.

Research into the assessment and prediction of landscape scenic visual quality over the last three decades has often been aimed at very specific applications. In reality, landscape management is a problem where the solution probably rests in the approaches and methods of many sciences such as forestry, landscape architecture, psychology, statistics and computer science. Also, landscapes vary widely and since the process of landscape perception is partially related to the observers' personal experiences, culture heritage and

taste as well as a complex interaction of the physical landscape elements, it is not surprising that at this time there is still no complete set of satisfactory, comprehensive and foundational theoretical frameworks in this field. Therefore, a different, more comprehensive method for integrating the various approaches must be examined.

Potential Computer Science Applications

Computer science may offer some methods that will permit better, more comprehensive and "more realistic" approaches to landscape assessment. A branch of computer science called artificial intelligence (AI) may provide the technology to accomplish this goal. Traditional evaluation methods and statistical models have limitations; it is possible that AI technology can provide a basis for combining assessment methods and thus provide a better way to evaluate and manage landscape resources.

Expert Systems

Much AI work involves development of expert systems that use symbolic knowledge to simulate the behavior of human experts. A common expert

system is composed of a user interface, an inference engine, and a domain specific knowledge base (Stock, 1987).

The user interface is an essential part of an expert system. It is the task of the interface to handle all the communication between the user and the expert system. The user's impression of the expert system usually depends on the nature of the interface. The way that information is presented to the user should conform to the user's model of the task and expectations.

To arrive at conclusions, an expert system needs to relate pieces of knowledge by performing inference or deduction. The part of an expert system that performs inference is called an inference engine. The task of the inference engine is to take the knowledge in the knowledge base and carry out a set of actions that utilize the knowledge in finding a solution to the problem.

A knowledge base contains facts, structures and heuristic rules that represent expert knowledge about the domains of expertise. An expert system searches through the knowledge base to find a solution to a problem.

Providing high quality explanations is one of the most important research fields in artificial intelligence. Generating explanations is actually a problem of

producing machine generated descriptions of the operation of a computer system --- what it does, how it works, and why its actions are appropriate (Swartout, 1987). Generally, trust in a system results from not only the high quality of its results, but also the clear description of how they were derived. So, a successful system should be able to explain what it is doing and justify why it is doing it.

It may be possible, then, for an expert system to serve as the tool for integrating a large body of rules which are used to develop assessments of landscape quality. Additionally, such a system can provide coherent, logical explanations of the complex interactions and rules which give rise to such assessments. Thus, such a system may have more intuitive appeal while actually taking into account many more causal or correlated predictions of an illusive criterion like landscape quality.

CHAPTER III

METHODS

Rule Based Systems

Rule based systems (RBS) are the best means available today for capturing and encoding the problem-solving know-how of human experts. Generally, experts can express most of their problem-solving techniques as a set of situation-action rules. This makes RBS's the ideal method of choice for building knowledge intensive expert systems (Hayes-Roth,1987).

Rule based systems may be defined as modularized "know-how" systems. Know-how refers to practical problem-solving knowledge. It uses various kinds of personal information, including inferences that follow from observations, abstractions, generalizations and categorizations of given data; necessary and sufficient conditions for achieving some goal; preferred strategies for eliminating uncertainty or minimizing other risks and so on. The RBS's can directly incorporate rules that emulate the effective special-case reasoning characteristic of highly experienced professionals. Since each rule approximates an independent part of know-how, RBS development has two key

characteristic features:

1. A RBS system can improve its performance and solve more complex problem as the system builders refine its existing rules and add new knowledge.
2. A RBS system has the ability to explain its reasoning which makes its processing and conclusions more convincing and reliable.

RBS's have been considered the most practical way to build expert systems that incorporate large amounts of judgmental, heuristic and experimental know-how because many human experts generally find it easy to express methods for solving problems in their application areas using rule formation.

Basic Expert System Principles

A simplified form of a rule based system consists of storage and processing elements, i.e. knowledge base and inference engine. The basic cycle of a RBS consists of select and execute phases. In the selection phase the system determines which rules can apply and chooses one in particular to execute. In the execution phase the system interprets the selected rule to draw

inferences that alter the dynamic storage of system. A knowledge base stores rules and facts. Rules always express a conditional, with an antecedent and a consequent component. The interpretation of a rule is that if the antecedent condition can be satisfied, the consequent may also. The consequent can be either an action or a conclusion.

Rules use symbolic descriptions to characterize relevant situations and corresponding actions. Rules can be very precise or gross and the intermediate partial solutions may be abstract or detailed. On the other hand, facts constitute the other kind of data in a knowledge base. Facts express assertions about properties, relations, propositions and so on.

Generally, there are two ways to proceed toward an inference; backward chaining and forward chaining. In backward chaining, the program starts with the goal configuration. It finds one or more general rules whose conclusion matches the goal to some extent and formulates as a new goal the problems of satisfying all the conditions for applying the general fact. If there are several rules that match some part of the current goal, then the program, must choose one of them to use (Henschen, 1987). In forward chaining, the specific facts are matched with the conditions of the rules rather than goal parts matching with conclusions. Thus, the program starts with the initial situation rather than

the desired end state. The intermediate steps are situations that can be reached from the starting state rather than secondary goals that must be achieved in order to solve the original problem. When one of these reachable states matches the goal, the problem is solved (Henschen, 1987).

Forward chaining tends to generate many more intermediate states than backward chaining for problems in which the goal is fairly well defined since in backward chaining the effort is more focused on the given problem. However, forward chaining in some cases is more appropriate than backward chaining. Forward chaining is used in this research because there is no goal to configured when the reasoning starts. The reasoning starts from the root of the decision tree, it reaches a final conclusion gradually by consulting the dynamic information and reasoning over the production rules.

Explanation Concepts

Usually, one of two major approaches have been taken to provide programs with an explanatory capability (Swartout, 1983). The first approach uses previously prepared text to provide explanations (also called the canned text approach) and the second produces explanations directly from computer program code.

One problem with the canned text approach is that it is difficult to maintain consistency between what the program really does and what it claims to do because the program code and the explanatory text strings are independent. Another problem is that all the possible questions must be anticipated in advance and the corresponding answers must be provided by the programmer. This might be able to make do in a very small system, but in a large system, this is almost impossible. Thus, it would be difficult to use such an approach in a system dealing with visual quality assessment since this type of expert system is quite complicated and users can, very likely, develop various and unanticipated questions.

However, with an explanation routine that examines the executed program, a system can produce explanations by performing transformations on the program code. This method overcomes the problems of the canned text approach and has been used in other expert systems (Winograd T., 1971; Shortliffe E.H., 1976).

In the system developed at MIT for advising physicians regarding digitalis therapy, an English explanation can be generated for both the methods it uses and how those methods were applied during a particular session. It demonstrates a simple way to generate explanations based on the code of the

program and a trace of its execution. One limitation of this type of system is that the specificity of the explanations it produces is fixed when the program is written. For example, a program of this type may be capable of producing very general or very specific explanations, but it may not be able to produce explanations of intermediate specificity. More recently, some work has been done by using a mixture of the canned text approach and direct-from-computer-code approach to improve its ability of providing explanations with better quality (Kastner et al., 1982).

Despite the advantages of the direct-from-computer-code approach, it still can suffer the problem of lacking an ability to give adequate justifications for its actions. That is, although this method can explain what it does or did, it has only very limited ability to explain why it did what it did. Although Swartout (1983) pointed out that the Digitalis Therapy Advisor System (Swartout, 1977) could adequately explain program behavior, it could not provide justifications for that behavior. The reason causing the problem is that the knowledge required to provide these justifications is used to produce the program but is itself not recorded as part of the code, and hence is unavailable. In the new explanatory system XPLAIN, Swartout improved major portions of the Digitalis Therapy Advisor and provided superior explanations of its behavior. XPLAIN uses an automatic programmer to generate a performance program. As

the program is created, a refinement structure is created to give the explanation routines access to decisions made during the creation of the program. By examining the refinement structure, XPLAIN provides justifications of the code and thus it can provide justifications for its behavior.

The Explanation Visual Management System

The intention of this research was to develop an Explanation Visual Management System (EVMS) that uses subjective decision rules of design experts such as landscape architects and the general public to evaluate landscape visual quality with detailed explanations and reasoning logic. Additionally the resulting assessments (expert opinions) of EVMS were then compared with the result of a PMES (Psychophysical Model Explanation Subsystem).

The subsystem of EVMS consists four separate programs or components:

1. man-made feature component;
2. natural feature component;
3. spatial organization component; and a
4. visual composition component.

These four program components deal with different aspects of the landscape. The first two parts are mainly concerned with the characteristics of the landscape content, on the other hand, the last two parts are mainly concerned with the dimensional composition of the landscape. Conceptually, these four parts are independent with each other, e.g. man-made feature part is focused on the impact of man-made features whereas spatial organization part is focused on the visual composition of the landscape. In reality, however, these four parts are all closely related with each other and a change in one part may affect another part or even all the other parts. Thus, to have a comprehensive and unbiased evaluation of the landscape, the interaction among those four parts should also be carefully considered.

The four parts of the EVMS can run either separately or jointly. Separate running of each program component generates a separate result for that component. In other words, an independent design assessment can be generated for each of the components. A combined result can be achieved by jointly running all four components and assessing the interaction among the parts.

In creating the EVMS, two different kinds of knowledge were considered. The first, called the *domain-dependent* knowledge, contains the descriptive

facts of the domain. This domain-dependent knowledge is acquired from the answers of many user queries. This domain-dependent knowledge is then stored in a dynamic knowledge base. The dynamic knowledge base is problem-specific oriented. That is, when a different landscape is evaluated, different domain-dependent knowledge is collected for that particular landscape. The dynamic knowledge base can be modified at any time during the run of a system component to address changes in judgements being made.

The second kind of knowledge utilized in developing the EVMS is *domain-independent* which can be further classified as factual or judgmental knowledge. Factual knowledge refers to facts that are valid, by definition and with certainty, independent of a particular case. Judgmental knowledge, on the other hand, is composed of the rules acquired from experts. Unlike a domain-dependent knowledge base, both factual and judgmental knowledge is in the form of a static knowledge base and is not subject to change during system runs. Any change to a static knowledge base may affect the whole system's operation. In other words, a change in an inference rule may change a valid conclusion into an invalid one. Therefore, the static knowledge base must be developed and maintained very carefully.

Judgmental knowledge in the EVMS is built in a structure called a

decision tree. This decision tree reflects the decision directed approach. Each node in the tree represents a decision and contains information about how the system makes its decision to go to next node. Inside each node, in addition to an inference rule, there are some other actions that help the system accomplish its reasoning process. For example, reasoning process starts from the root of the decision tree. By checking the dynamic knowledge base, a piece of domain-dependent information is selected and compared with the premise of an inference rule. Therefore, a branch of the decision tree can be chosen based on the different result of the comparison between inference rule and dynamic information. At the same time, when a node is chosen, some information about the inference rule is also accrued along with the process of reasoning. When reasoning is over, a conclusion as well as a set of accrued decisions are also dynamically generated.

According to the dynamically generated decision set, explanations are generated by matching the decisions with the appropriate entries of explanations. The explanation set was stored in the system when the system was developed. Every entry of the explanation corresponds to an inference rule and it is independent of any specific landscape. As a result, every landscape has its corresponding dynamically generated decision set and every decision set has its corresponding explanations.

Since the explanations were stored in the static knowledge base in advance and not subject to change during the operation of the system, they are canned explanations in this sense. However, every explanation is actually a kind of translation and specification of a corresponding decision rule that is dynamically chosen. In other words, the set of the final explanations is also dynamically formulated in this sense. Therefore, the explanations generated by the EVMS are "semi-dynamic".

Programming Environment

The EVMS was written in VT PROLOG, version 1.005. The host system was a MICROVAX II minicomputer running on the VMS 5.3 operating system. The interface between the host system and the EVMS implementor was a TEKTRONICS 4105A (VT 220 compatible).

Knowledge Engineering

Knowledge engineering for the four EVMS components was accomplished during a two year period from June 1990 - June 1992. A professional landscape architect and faculty member at Virginia Tech was the

subject of this engineering effort. This expert's career focus has been on both professional application and theoretical research aspects of landscape quality evaluation and assessment.

CHAPTER IV

RESULTS AND DISCUSSION

The EVMS is written in VT PROLOG and is made up of over 7000 lines of code (Appendices A - D). The entire system consists of four separate expert subsystems which generate scenic assessments for man-made features, natural features, spatial organization and visual composition. Each of these expert systems is comprised of six components that are logically independent of each other. The six program components are:

1. user interface and miscellaneous routines;
2. dynamic database;
3. static database;
4. production rule processor;
5. explanation generator; and
6. dynamic database modifier.

Program Components

User Interface and Miscellaneous Routines

Landscape visual quality assessments are generated from user responses to inquiries about various landscape characteristics. The user interface is responsible for asking various questions about the landscape characteristics and collecting the answers to be used as input for the EVMS. All inquiries in the system fit one of two categories; 1) questions which reference landscape content as part of the natural and man-made features subsystems and, 2) questions which reference landscape dimensions in the spatial organization and visual composition subsystems.

User responses to each inquiry are checked for validity by the *user interface*. If a valid answer is input, the user interface stores it into a dynamic database for later use. The user interface also takes care of screen arrangement, various prompt information and the communication between the program and the operating system.

An important type of predicate defined in the user interface is the input

checker. Whenever an input is read, it is checked by an input checker that is specific for that inquiry. When an illegal input is detected, the input checker is responsible for displaying some prompt information to tell the user what error he/she has made and the previous question is printed out again to let the user input a new answer. This process is repeated until a legal answer is read by the system. In this way, any inappropriate information input by accident can be avoided and the information stored in the dynamic database is guaranteed to be a legal input.

Miscellaneous routines contain a number of predicates that serve as public tools. For example, the predicate "append" puts two PROLOG lists into one list. The predicate "find_high" searches a list of elements and finds an element with the highest level of some property. The "page_control" predicate controls the print format that counts the lines of information having printed on screen so far and makes 20 lines a page on screen. Some of these public tools can be shared by all four subsystems whereas some of these miscellaneous routines are only used in a specific subsystem.

Dynamic Database

Each subsystem has a *dynamic database*. At initialization of each

subsystem, there are no contents in its respective dynamic database. The dynamic database is generated and developed as inquiry input is made. Generally, either the name of an identified feature or a piece of the description of the feature or some indicator that stands for some level of a property is stored in the dynamic database. The information in the dynamic database is domain specific. In other words, the dynamic database content will be different for each landscape and is developed as the responses to inquires about a particular landscape are determined and input. For example, a user may identify four man-made alterations or built structures for a particular landscape scene, let's call it scene "A". For another landscape scene, call it "B", the user may identify only one man-made alteration or built structure or even none of them. In addition to this, different landscape scenes usually have different scene characteristics such as "historical", "human scale", "apparentness" and so on.

One property of the dynamic database is that it contains a complete set of information concerning a specific landscape scene. This complete set of information is often redundant because not all the information may be used later traversing a decision tree and developing the decision list. The reason for this redundancy is that it is not known which decision path through the tree will be chosen for a particular landscape evaluation. Therefore, it is not known which piece of information in the dynamic database will be used in a specific

traversing process. Generally, in traversing through a decision tree and reasoning over a set of decision rules, all information in the dynamic database has a chance to be used. However, in a specific traversing process, some of the information may not be used at all.

Static Database

The *static database* was designed and generated by the programmer when the system was built. It does not change with the input and processing of the visual quality information in response to inquiries for any specific landscape scene. Since all four subsystems share the same set of conclusions, the conclusions in the static database are accessible to all subsystems. Every explanation in the static database is a stored translation and specification of the corresponding production rule. Every subsystem has its own set of explanations which are not used by the other subsystems since each subsystem has its own set of production rules. Each entry in the static database is stored in the form of a list written in PROLOG and is easy to manipulate via a PROLOG matching mechanism.

Production Rule Processor

The *production rule processor* is a software mechanism that traverses through a decision tree to select a valid path. A forward chaining strategy is used to reason over a set of production rules which are the base of the domain specific decision tree for each subsystem. The reasoning process starts from the root of the decision tree and then reads heuristic information from the dynamic database and chooses an appropriate child as its next node by consulting the heuristic information in the dynamic database and then reasoning over production rules. Each rule and piece of heuristic information from the dynamic database is accrued into a decision list. The entire reasoning process ends up at a leaf node that stands for a possible conclusion the system can draw. When the reasoning process is complete, a path is selected and its corresponding decision list is then generated. Since different scenes have different dynamic databases, this usually results in different paths being selected and different paths usually have different decision lists. Therefore, dynamically produced decision lists that contain a group of rules and landscape specific heuristic information can be translated into a corresponding explanation set.

Explanation Generator

The explanation generator uses a specific decision list as its input. It processes the decision list sequentially beginning at the top. Since the translation of each rule is already stored in the static database, a selected rule's name can be matched with the corresponding entry in the static database. The translation of the rule can then be processed and printed out. Since an explanation is a direct translation of the production rule used in the reasoning process, it can be determined which rule has been used and why it has been used.

Dynamic Database Modifier

After a dynamic database has been created, it can only be changed at certain points and by certain means. Modification of the dynamic database can be done only by the *dynamic database modifier*. After each subsystem has been executed and an assessment output generated, the "user interface" provides the user the choice of modifying the dynamic database.

The dynamic database modifier component searches through the dynamic database and prints its contents -- that is, all the answers input during

the last execution. For each entry in the dynamic database, a number and a name that describes a particular characteristic of that specific landscape are also given. The user can then choose one of the corresponding numbers to change the associated entry. The dynamic database modifier searches through the dynamic database by using the selected number as the index and deletes the appropriate old entry in the dynamic database. The modifier then prints out the corresponding inquiry so the user may input new information for that inquiry in the dynamic database. After a user has changed the contents of the dynamic database, he/she can either choose to change another entry in the dynamic database or run the program with the new information.

After modification of the dynamic database, the program will traverse through the entire decision tree and consult the modified dynamic database to generate a new decision list and a new set of explanations. Partial traversing through the decision tree from the point where the dynamic database has been modified is not permissible even though only a single entry of the dynamic database may have been changed. Even though this would result in a time saving, any change in a dynamic database entry may well affect the entire reasoning process and the heuristic information selected for use. Therefore, traversing of the decision tree must start from the root of the tree.

The Subsystems

The four subsystems are logically independent, although they have similar structures, components and design. These subsystems are logically and physically independent in most aspects. Some routines are shared by more than one subsystem. However, each subsystem has its own dynamic database, static database, dynamic database modifier and production rule processor. Only parts of the user interface and explanation generator are shared by more than one subsystem.

Man-made feature Subsystem

The *man-made feature subsystem* is the largest of the four subsystems. It has about 3500 lines of code (Figure 1A - 1I). Since more than one man-made alteration or built structure may be identified, a type of loop programming is used. In other words, all 15 inquiries are generated for each of the identified alterations or built structures. The answers to the inquiries range from simply "yes" or "no" to providing a description or an indicator which stands for a certain degree of some property. Some example inquiries follow:

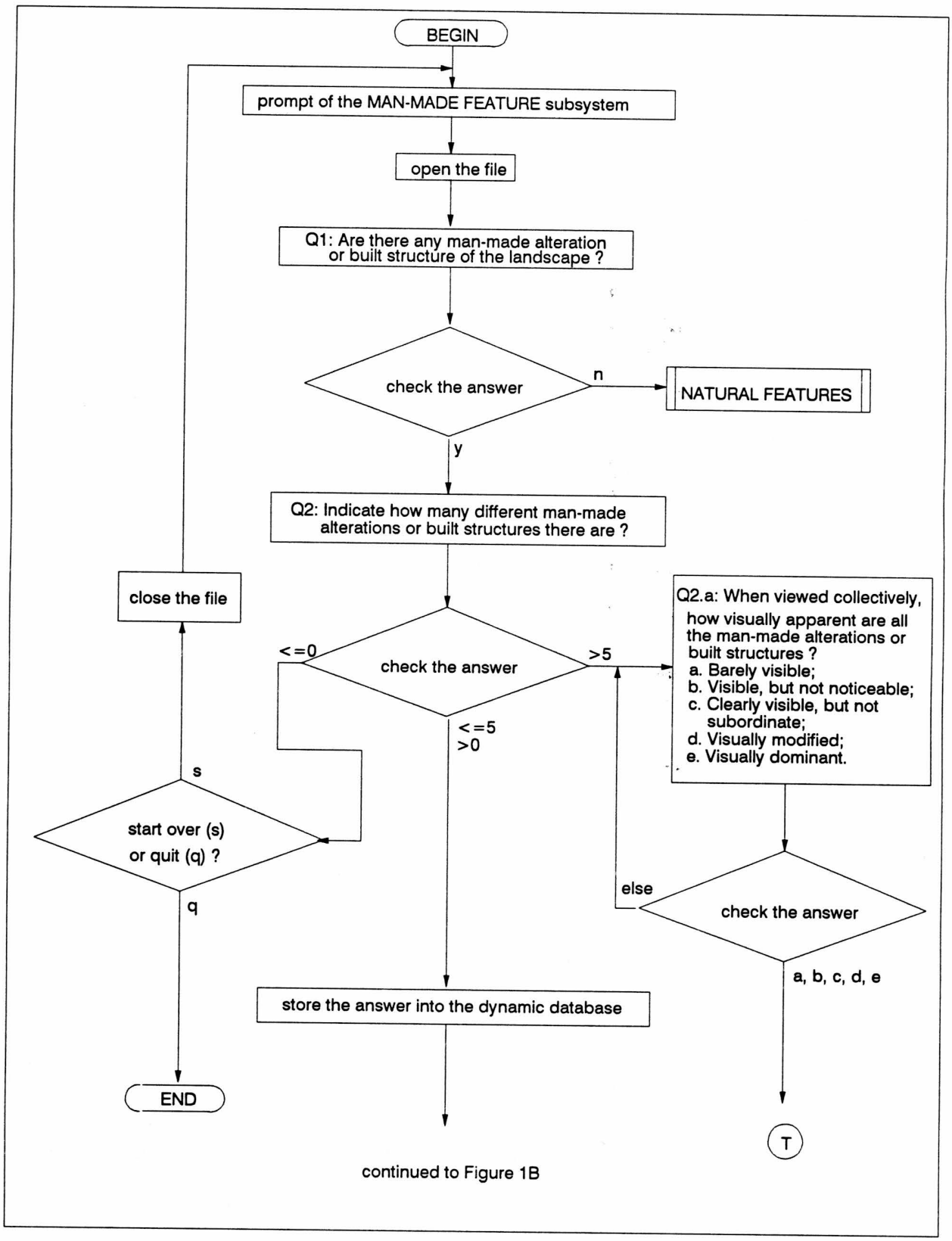


Figure 1A MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

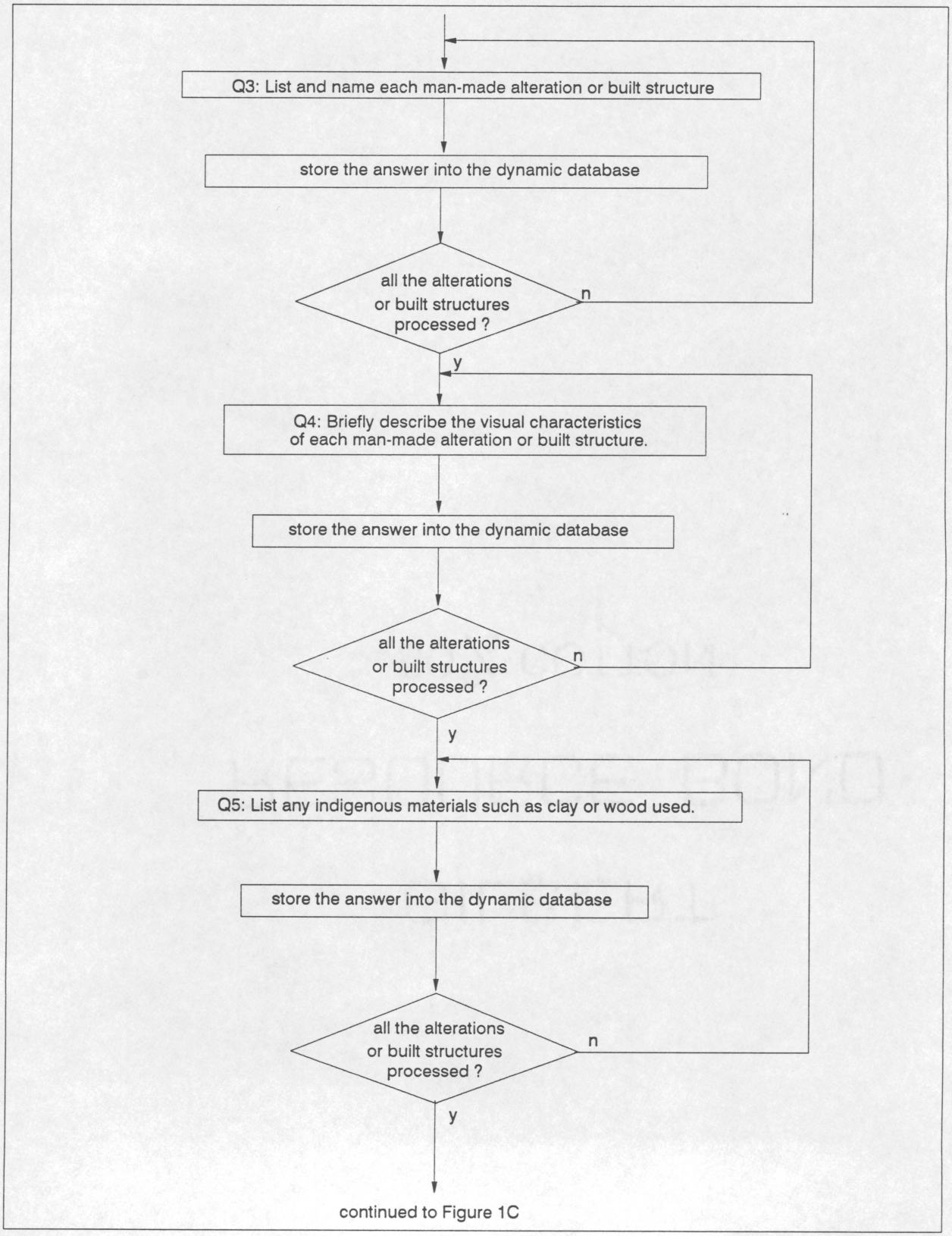


Figure 1B MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

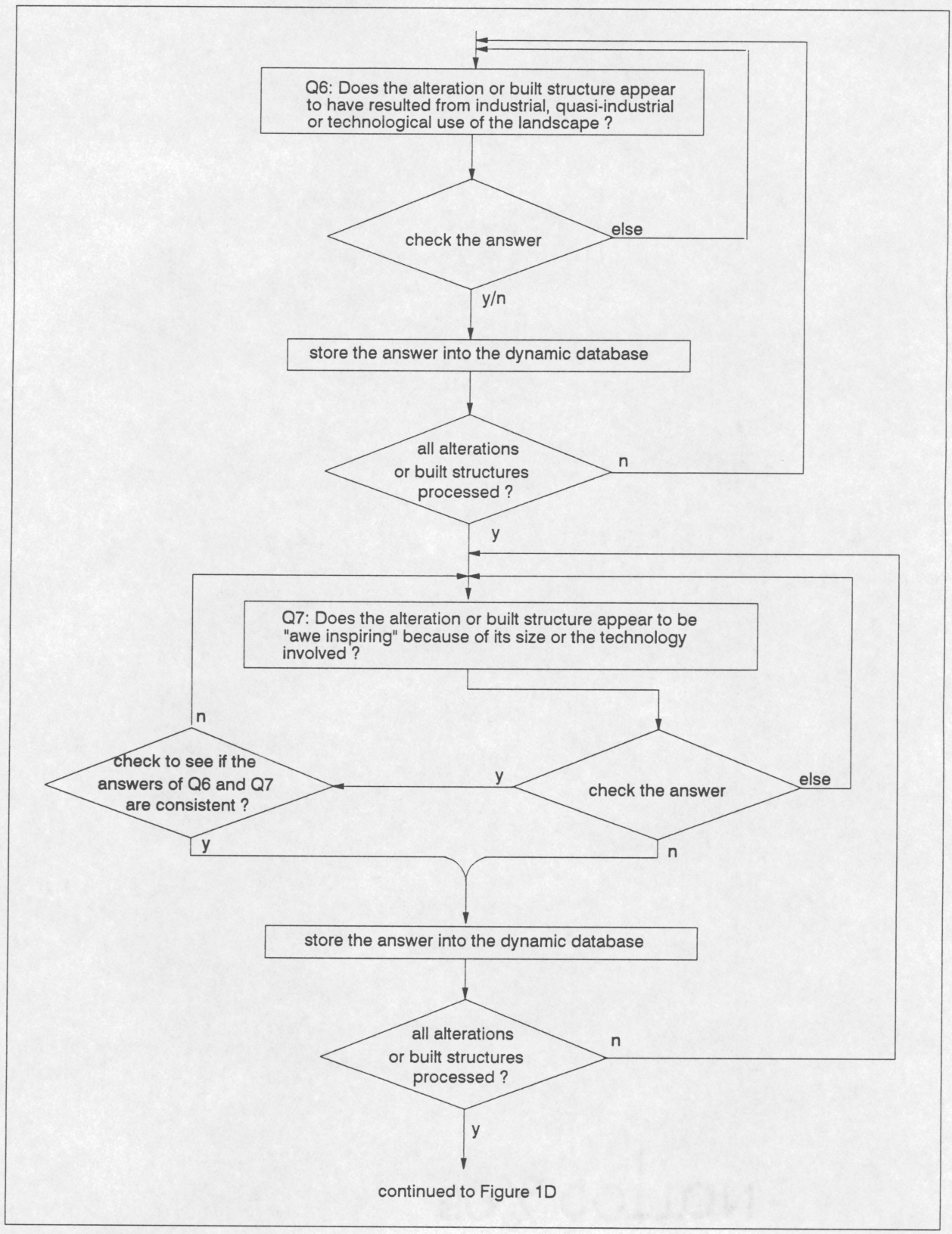


Figure 1C MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

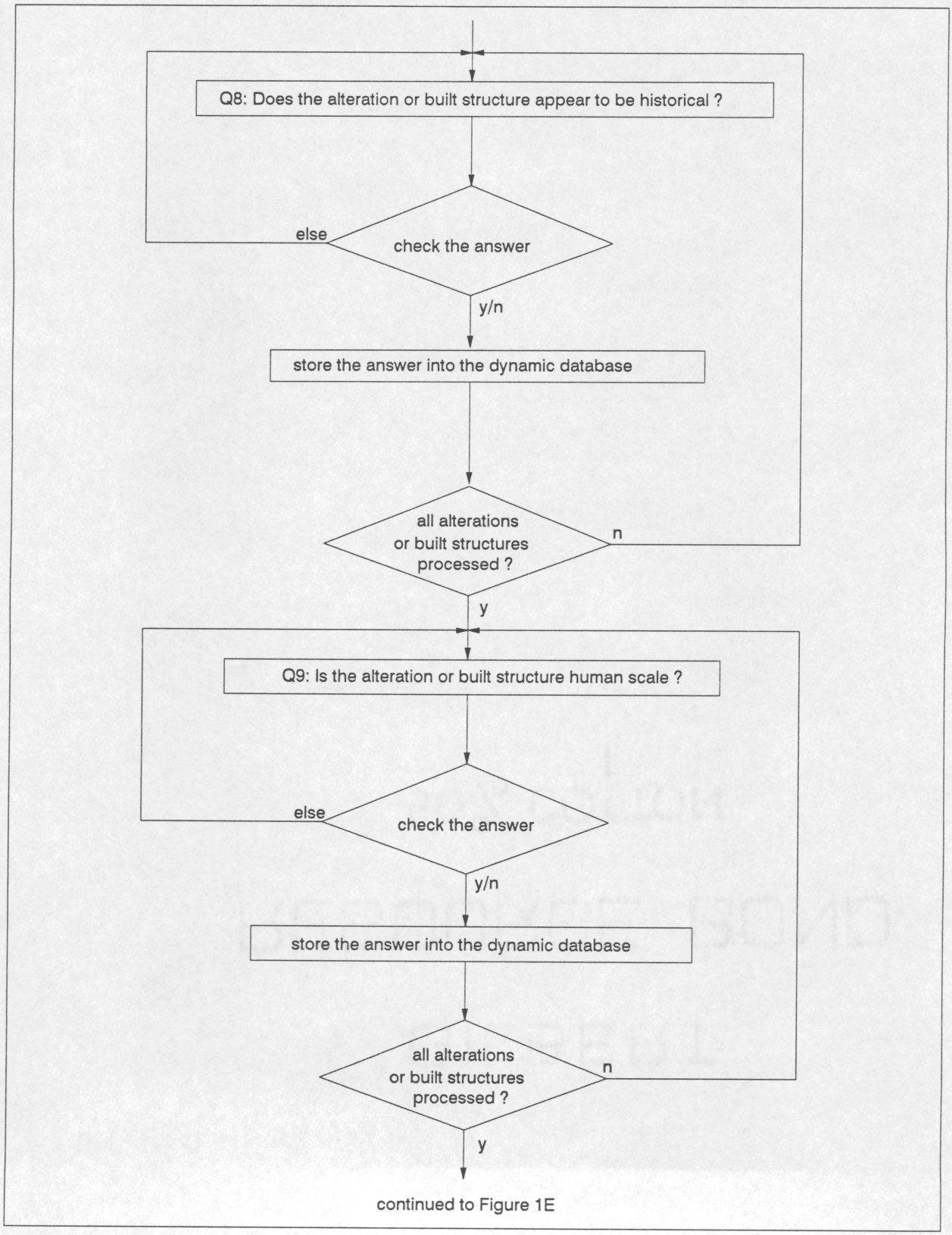


Figure 1D MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

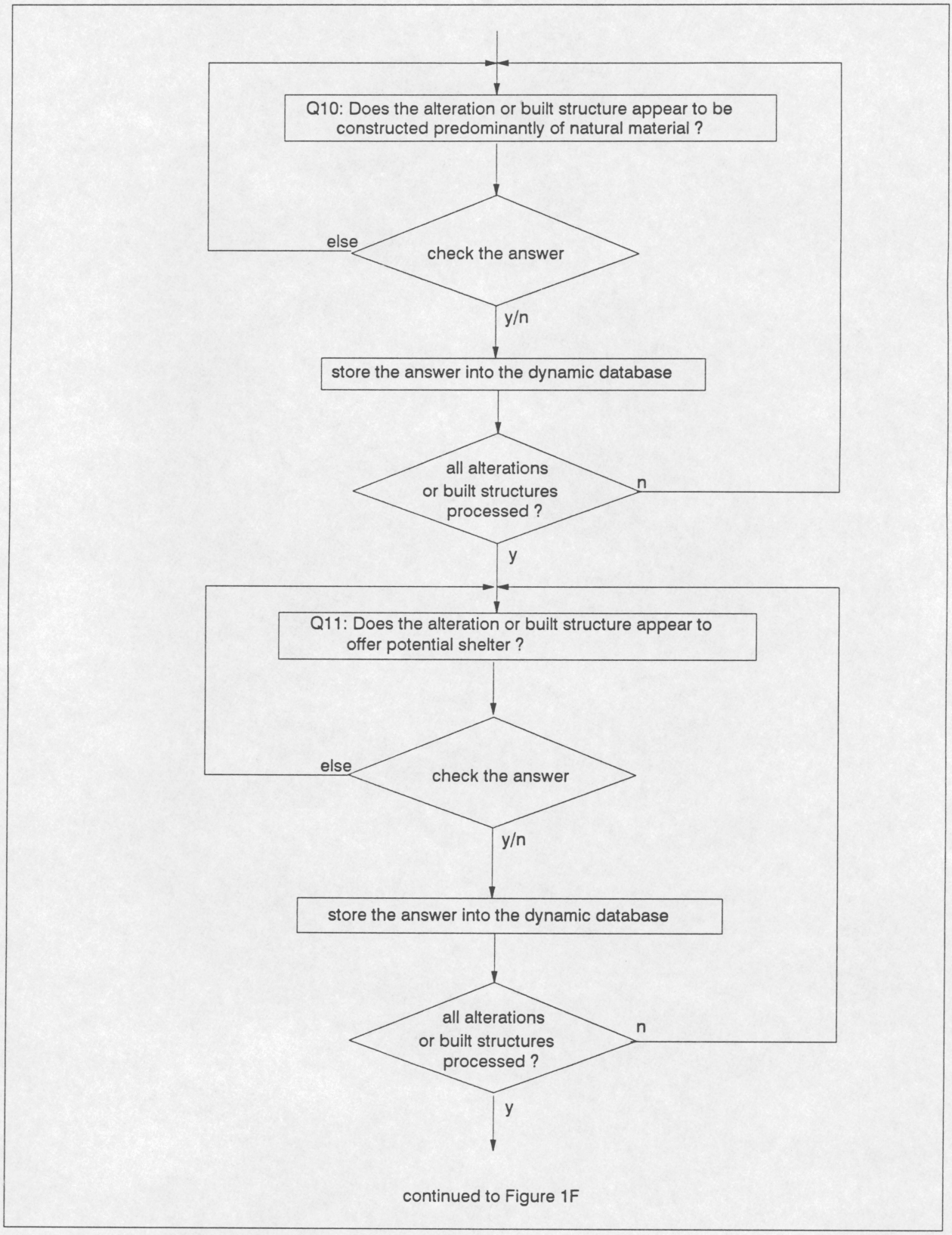


Figure 1E MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

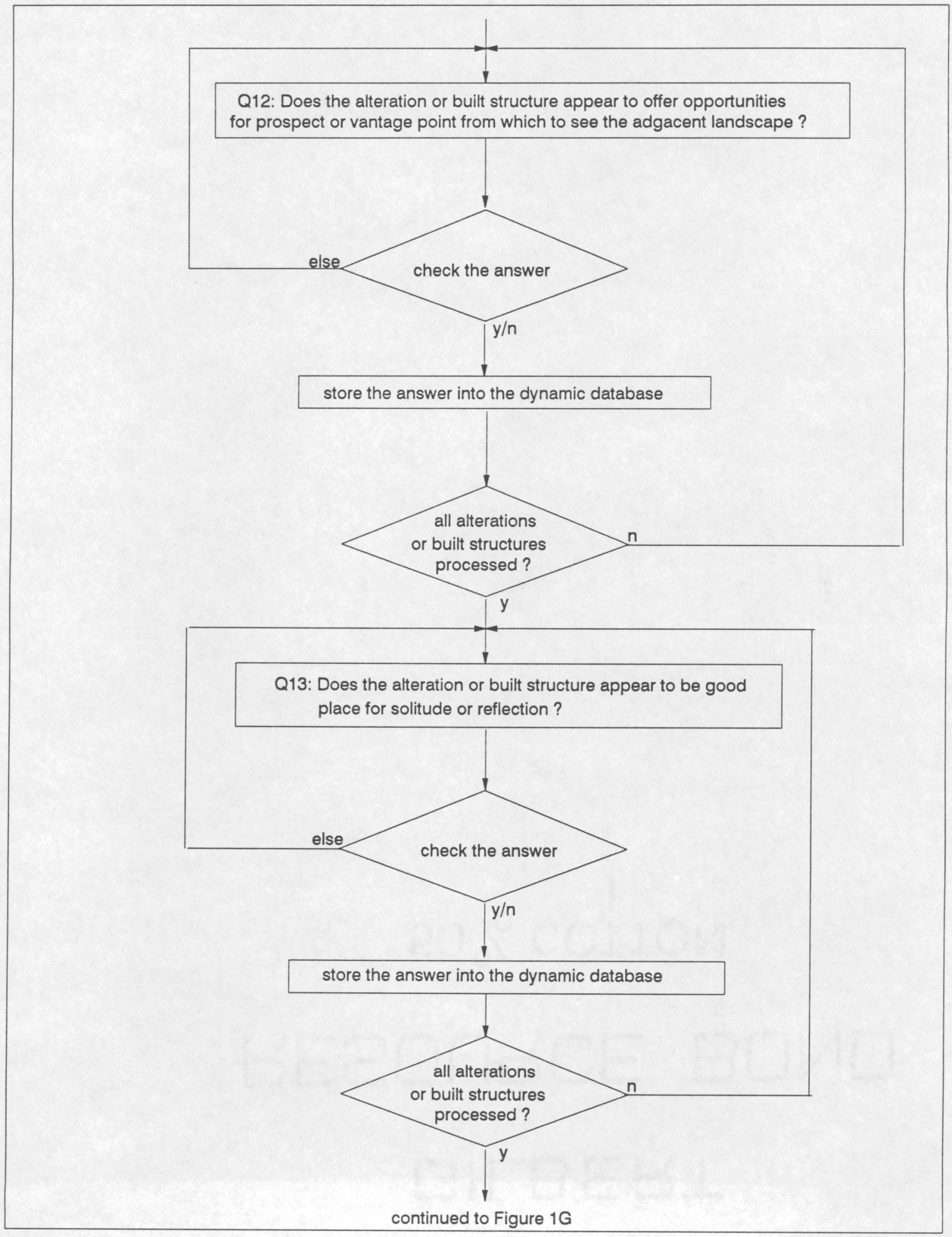


Figure 1F MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

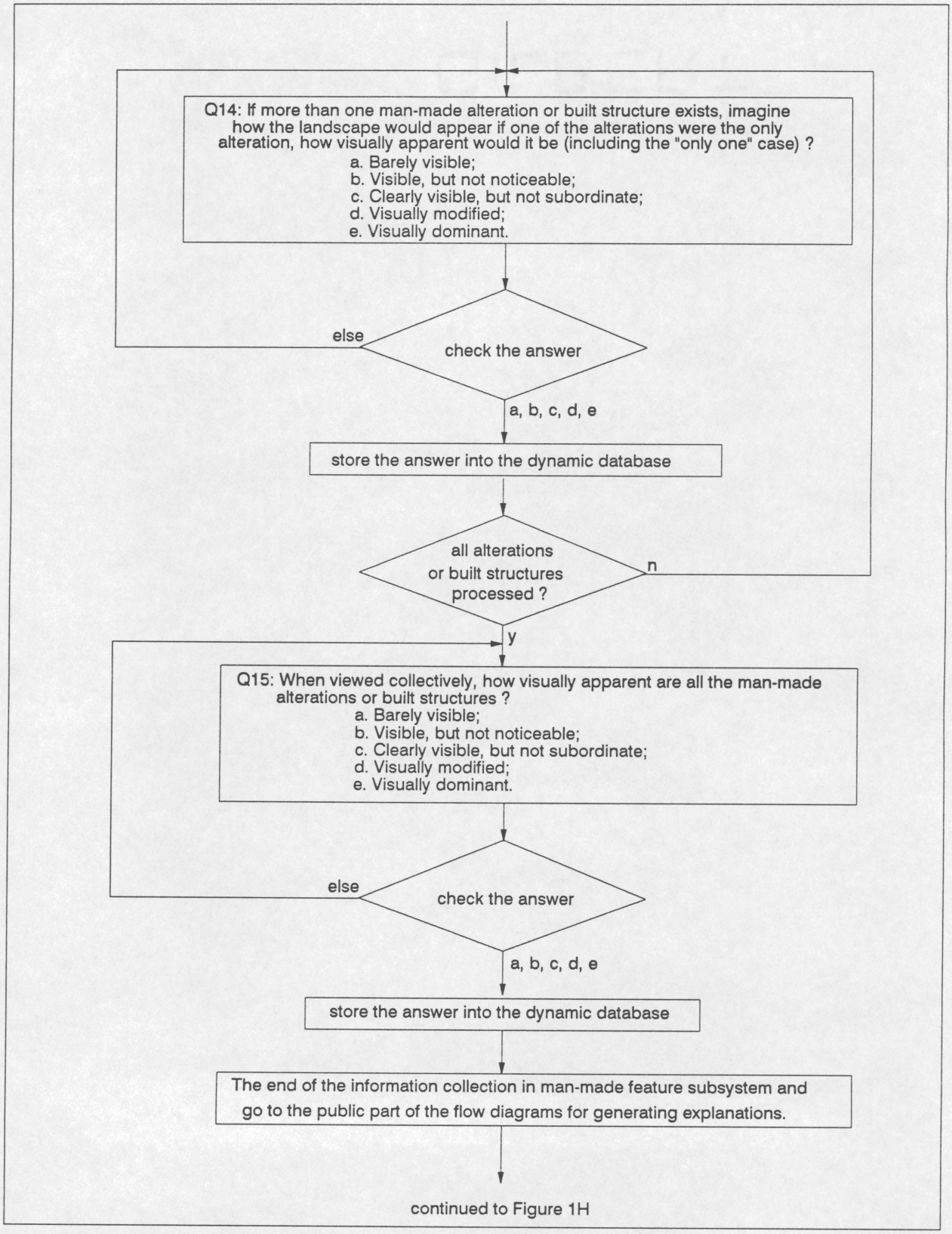


Figure 1G MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

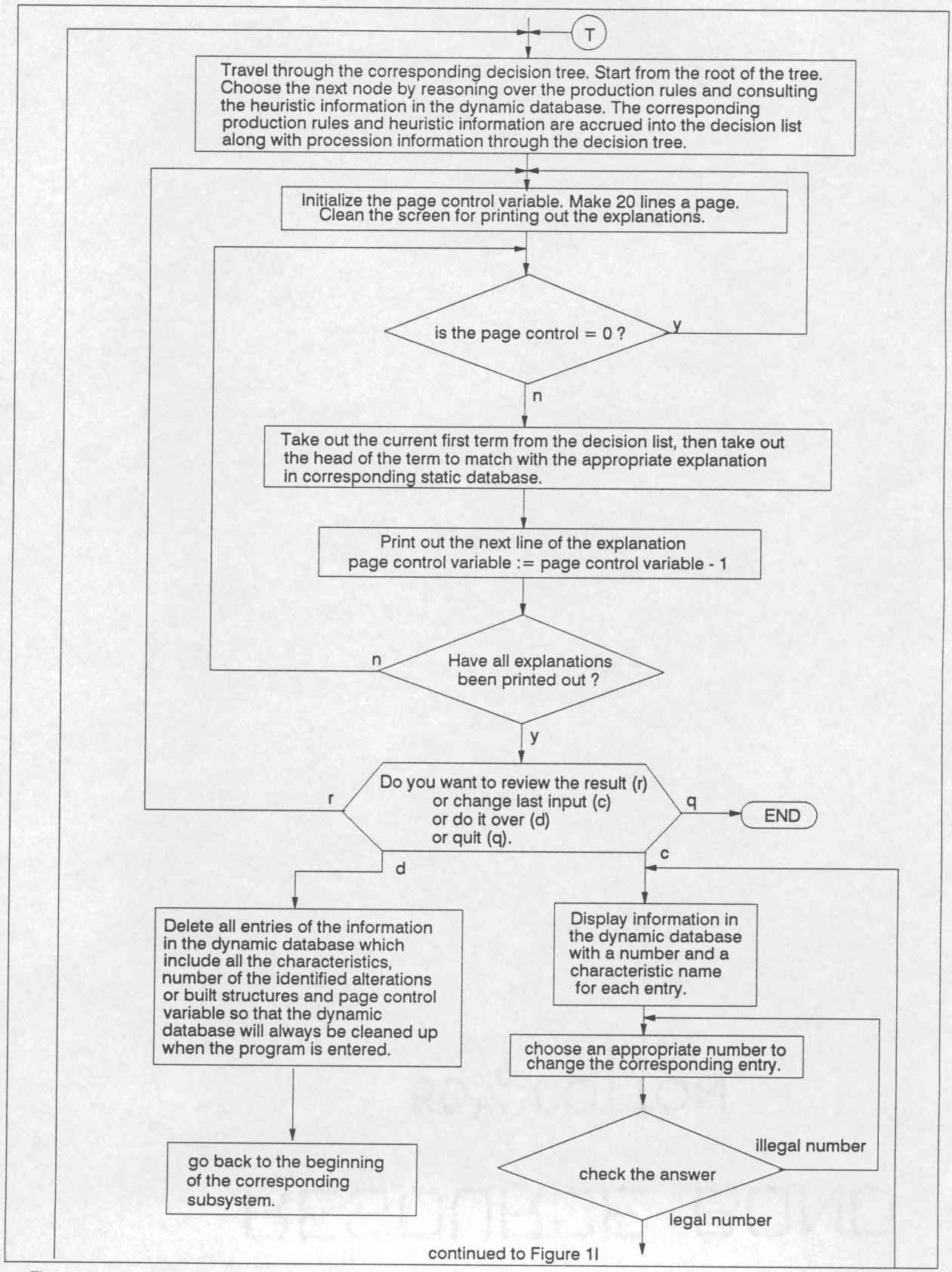


Figure 1H MAN-MADE COMPOSITION SUBSYSTEM FLOW DIAGRAM

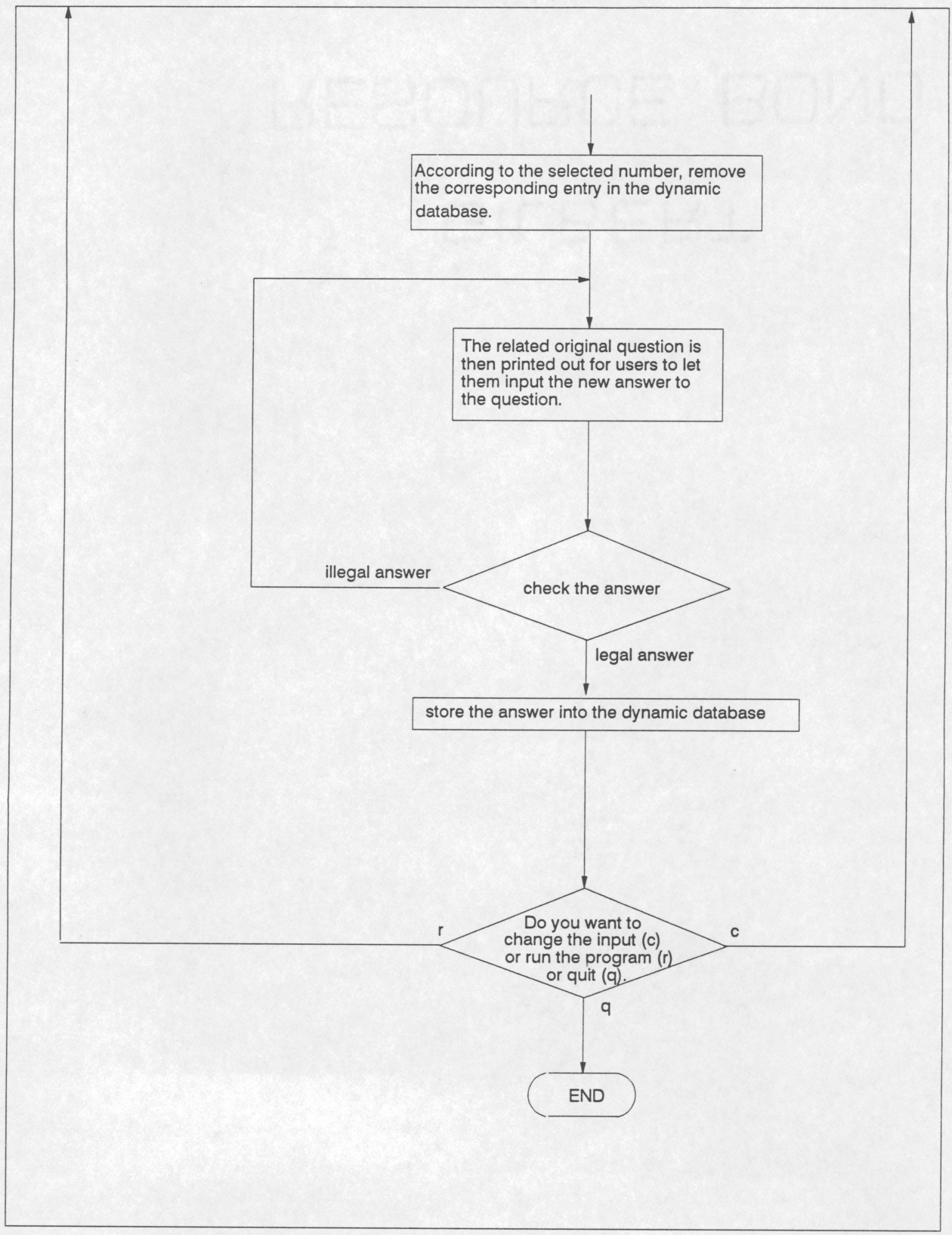


Figure 1l MAN-MADE FEATURE SUBSYSTEM FLOW DIAGRAM

Q3: Please list and name each man-made alteration or built structure ? (Figure 1B)

Q4: Please briefly describe the visual characteristics of each man-made alteration or built structure. The description should include the size, color, position in the landscape and any other noteworthy visual aspects of the man-made alteration or built structure ? (Figure 1B)

Q6: Does the alteration or built structure appear to have resulted from industrial, quasi-industrial or technological use of the landscape ? (Figure 1C)

Q7: Does the alteration or built structure appear to be awe inspiring because of its size or the technology involved ? (Figure 1C)

Generally, the user inquiries are independent of one another. In other words, the answer to one question usually has nothing to do with the answer

to another question. However, in this subsystem there is some relationship between question 6 and question 7 (see above). To be more specific, if the answer to question 6 is "no" for an identified alteration or built structure, then the answer to the question 7 cannot be "yes" for that identified alteration or built structure, because industrial, quasi-industrial or technological use has already been rejected in question 6. Therefore, the answers to questions 6 and 7 are checked for discrepancy. If disagreement is found, then question 7 is repeated to permit the user to reconsider the inquiry. This process will continue until no contradiction exists.

When the number of identified man-made alterations or built structures in a particular landscape under evaluation are more than 5, only an inquiry about their "collective apparentness" is generated (Figure 1A). In the case of a large number (i.e., more than 5) of man-made alterations or built structures being identified, it is not appropriate nor necessary to evaluate characteristics of individual alterations such as "apparentness", "human scale" and "potential for prospect" for each identifiable man-made alterations or built structures since the collective effect of all alterations will out weigh effect of individual alterations. With so many identified alterations or built structures, only "collective apparentness" has a decisive effect on visual quality. When the identified man-made alterations or built structures are less than or equal to 5,

inquiries concerning the detailed characteristics of the landscape are asked about each of the identified alterations or built structures (Figure 1A - 1G) since the viewer's attention will focus on each of those detailed characteristics of the landscape and thus each of these man-made details has an effect on overall visual quality.

After all inquiries are made and answers given, the system begins traversing the decision tree of the *man-made feature subsystem*. The decision tree (Figure 2) lists all the possible paths and conclusions for the *man-made feature subsystem*. There are 263 main nodes and 191 leaf nodes in the decision tree. The associated explanatory rules for this tree are in Appendix E.

There are several points at which the user can quit the *man-made feature subsystem* and exit to the operating system. While running the *man-made feature subsystem* the user can execute the program repeatedly without having to exit the subsystem. The dynamic database can also be modified and the subsystem can be executed after each modification. When the *man-made feature subsystem* is first entered, the user is asked to indicate if any man-made alterations or built structures are identifiable. If the answer is "yes", then the program continues. Otherwise, the program will prompt the user to go to the next part, the *natural feature subsystem*.

As an example, the entire process of explanations generated by the man-made subsystem follows. Suppose, for a landscape scene, three man-made features or structure alterations have been identified, i.e. *road, camp ground* and *hiking trail*. Since the number of the identified man-made features or structure alteration is less than five and larger than zero, then 15 questions (Figure 1A-1G) about various landscape characteristics are asked for the identified *road, camp ground* and *hiking trail*. After all answers to the questions have been processed and stored in the dynamic database, the content of the dynamic database for this specific landscape scene may look like this:

(name 1 "road"), (name 2 "camp ground"), (name 3 "hiking trail")

These three entries record the names of all three identified man-made features or structure alterations. The following six entries record all the descriptions about the visual characteristics of the identified man-made features or structure alterations.

(desc 1 "desc1"), (desc 2 "desc2"), (desc 3 "desc3")

(indi 1 "indi1"), (indi 2 "indi2"), (indi 3 "indi3")

These descriptions include the size, color, position in the landscape and

any other noteworthy visual aspects of the man-made alteration or built structure. Any indigenous materials such as rock, clay or wood used in the construction of built structures are also listed in the last three entries. Note that the entries for "desc1" to "desc3" and "indi1" to "indi3" are descriptions and these can be either short (nothing) or long (several paragraphs).

Entries have either "yes" or "no" answers to all questions:

(tech 1 "no"), (tech 2 "no"), (tech 3 "no")

(awe 1 "no"), (awe 2 "no"), (awe 3 "no")

(hist 1 "no"), (hist 2 "yes"), (hist 3 "yes")

(huma 1 "no"), (huma 2 "no"), (huma 3 "no")

(natu 1 "no"), (natu 2 "yes"), (natu 3 "yes")

(shel 1 "no"), (shel 2 "no"), (shel 3 "no")

(pros 1 "no"), (pros 2 "yes"), (pros 3 "no")

(refl 1 "no"), (refl 2 "yes"), (refl 3 "no")

The entries for "tech" indicate that none of the identified features or structure alterations resulted from industrial, quasi-industrial or technological uses of the landscape. Also, none of the entries for "awe" indicate that any of the identified features or structure alterations are "awe-inspiring" because of

their size or the technology involved. Quite similarly, the entries for "hist", "huma", "natu", "shel", "pros" and "refl" store all the answers to the inquires about the characteristics of the landscape scene for "being historic", "being human scale", "being a place for shelter", "having opportunities for prospect" and "having opportunities for solitude and reflection" respectively.

The last two groups of the entries in the dynamic database are:

(visb 1 "c"), (visb 2 "d"), (visb 3 "a")

(coll "d")

The first three entries above store the visual apparentness for the three identified man-made features or structure alterations. In other words, the visual apparentness of the *road* is "clearly visible, but subordinate (c)", the visual apparentness of the *camp ground* is "visually modified (d)" and the visual apparentness of the *hiking trail* is "barely visible (a)". Finally, the collective apparentness for all the three man-made features or structure alterations is "visually modified (d)".

With all the answers stored in the dynamic database in the form shown above, the *production rule processor* can start to traverse through the decision

tree of *man-made feature subsystem* (Figure 2). By consulting the content in the dynamic database for this specific landscape scene, a unique path is selected. Along with the process of reasoning over the production rules, a decision list is gradually accrued and the specific decision list for this landscape scene looks like:

((alter 3) (tech n) (huma n) (natu y) (shel n) (hist y) (resolution 2))

This decision list records all the nodes being passed in the decision tree and all the decisions being made along the reasoning over the production rules. The *explanation generator* then takes this decision list as its input. It processes the first term of the list, i.e. (alter 3), to get an appropriate explanation. Since (alter 3) involves a random number (3) in the term, a special process is needed. However, a typical process of a term in the decision list to get the corresponding explanation is to put the entry name and related answer, e.g. "tech" and "n", into the term (expla tech n ?expla) where ?expla is a variable. And then, the *explanation generator* brings the term (expla tech n ?expla) into the static database and tries to match it up with the translation and interpretation of the production rule. Inside the static database, there is also a list that has the form:

((expla tech n ((explanation.....)

(explanation.....)

.....

)))

As soon as the matched list is found, ?expla locates the appropriate explanation and it is printed out on the screen according to a certain format. This process continues until all the terms in the decision list are processed.

The explanations corresponding to the term (alter 3), (tech n) and (hist y) appear as follows:

"Visually evident man-made features or structure alterations in the natural landscape tend to have a negative influence on visual quality of this type of landscape unless the man-made features or structure alterations are awe-inspiring, related to historic use of the landscape or are pleasant places for people. The extent of the influence on visual quality is dependent upon how many man-made features are present. This landscape was found to contain 3 features or structure alterations. The extent of negative or positive influence on visual quality of the landscape depends on

the individual and collective characteristics of the man-made features or structure alterations.

Industrial, quasi-industrial or technological alterations and/or structures in the natural landscape tend to have a negative influence on the visual quality of the landscape unless they are a source of visual interest due to their size (awe-inspiring) or historic significance. Industrial, quasi-industrial or technological alterations and/or structures tend to have a negative influence on visual quality because they seem out of place in the natural landscape and contrast with the natural character of the landscape. This landscape was not found to contain industrial, quasi-industrial or technological alterations or structures.

Alterations or built structures that "appear" to be old or historic enhance the visual quality of the landscape. They symbolize past cultures and times and tend to fascinate people, thus they contribute to people's appreciation of the landscape. If the landscape contains more than one man-made alteration or built structure then the alteration or built structure which is most visually apparent dominates and is responsible for the effect of

man-made alterations or built structures on the visual quality of the landscape. The most visually apparent man-made alteration or built structure in this landscape appears to be historic. Therefore, this alteration or structure has a positive influence on visual quality."

Finally, by processing the term (resolution 2) in the decision list, the *explanation generator* determines that the final conclusion or assessment is "high visual quality".

Natural Feature Subsystem

The *natural feature subsystem* is comprised of approximately 1600 lines of code. Like the previous subsystem, this program also requires the user to identify the number of visually significant features but in this case, natural features. If more than three visually significant natural features are identified (Figure 3A), then they are not considered to be "visually striking" since many natural features tend to scatter a viewer's attention. If the user indicates that more than three visually striking landscape features exist, the program will request the user to carefully reconsider whether or not there are not in fact actually three or less. This subsystem only executes if there are three or less

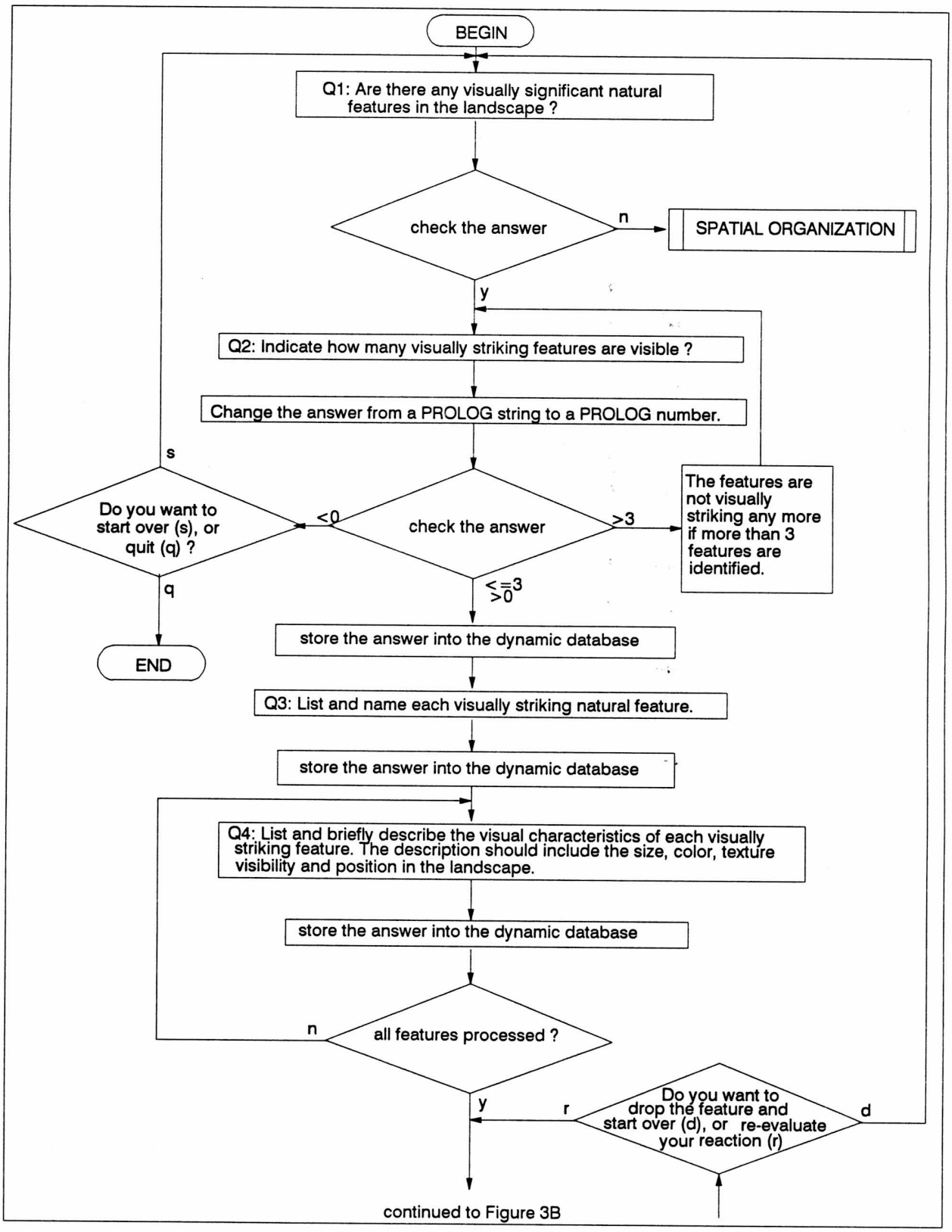


Figure 3A NATURAL FEATURE SUBSYSTEM FLOW DIAGARM

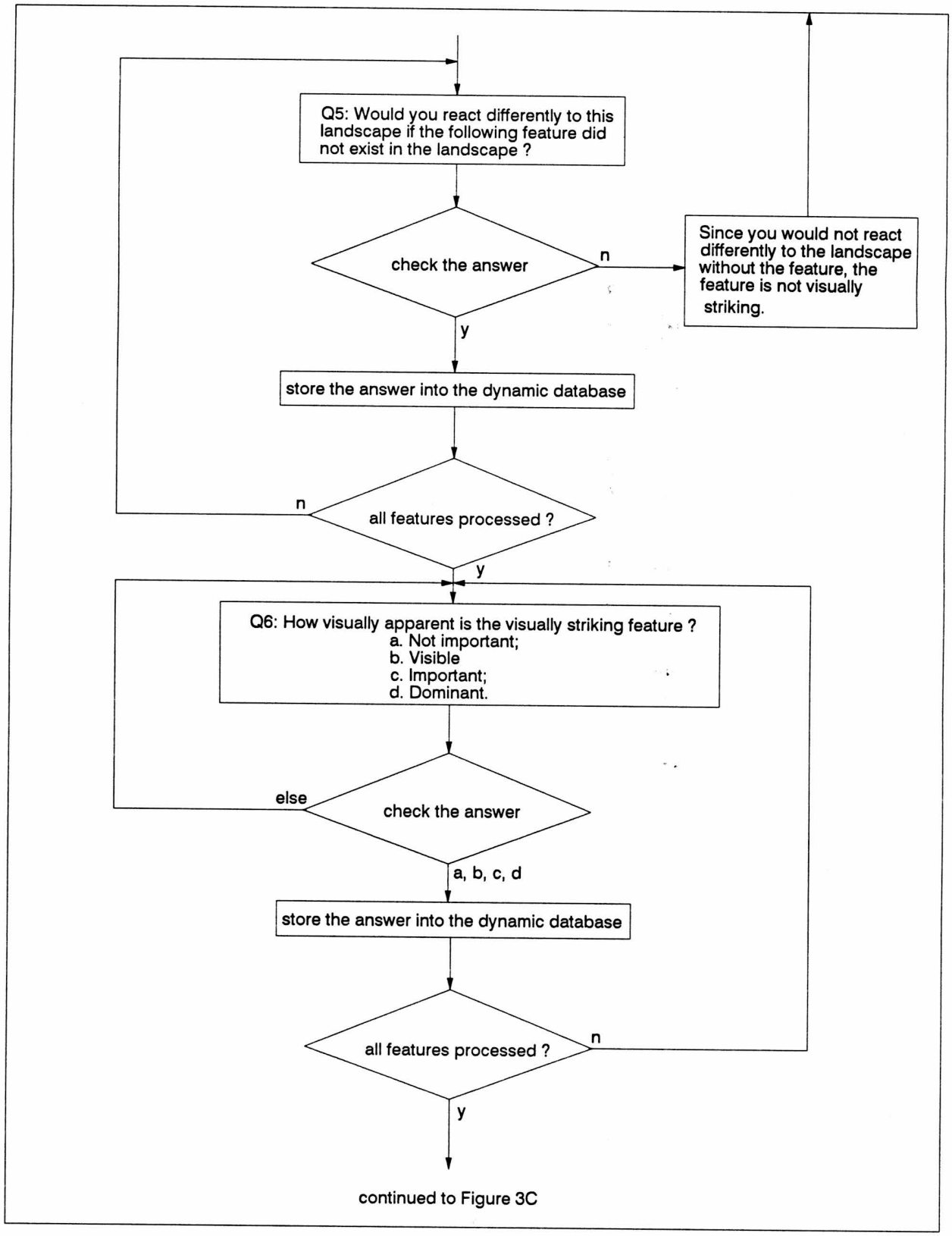
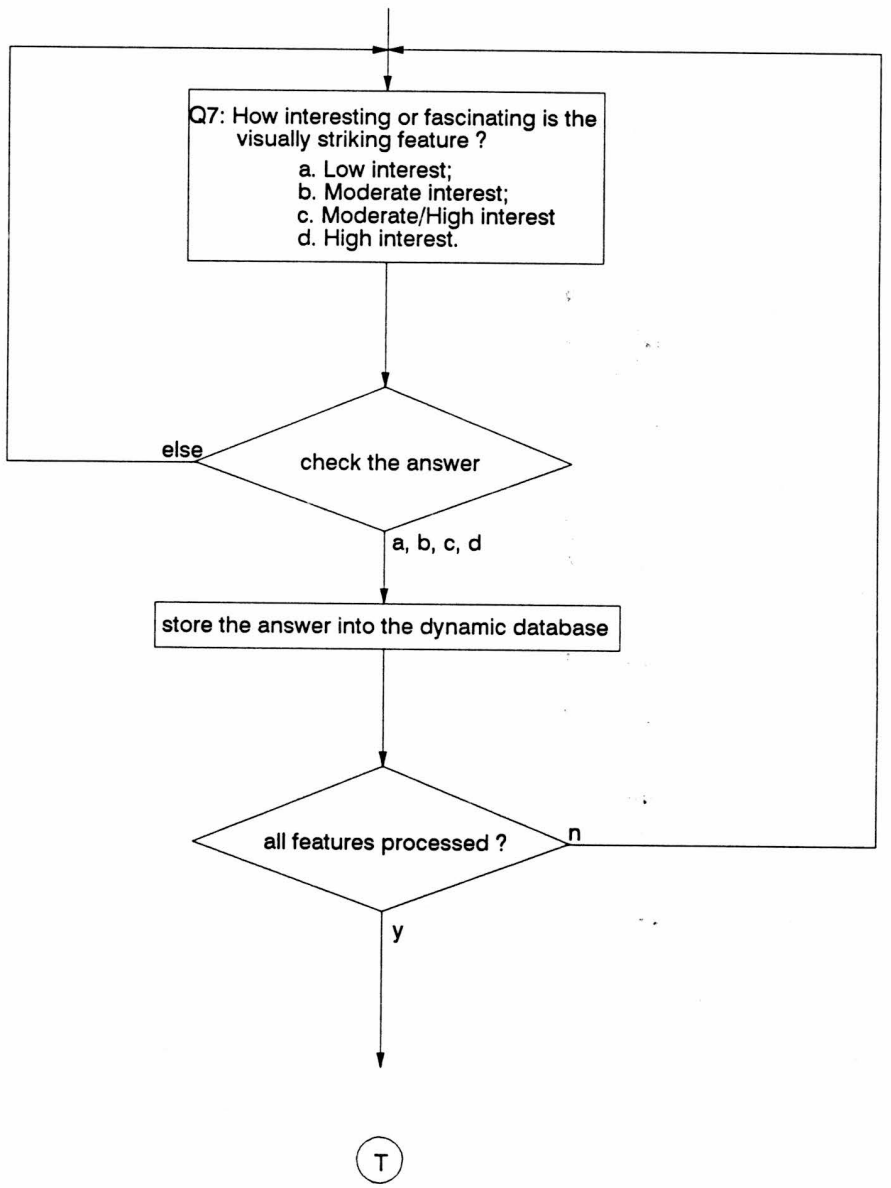


Figure 3B NATURAL FEATURE SUBSYSTEM FLOW DIAGRAM



continued to common program segment, see Figure 1H

Figure 3C NATURAL FEATURE SUBSYSTEM FLOW DIAGRAM

visually significant natural features. Since more than one natural feature may be identified as visually significant, loop programming is used for each of the identified natural features.

There are 7 inquiries in the *natural feature subsystem* (Figures 3A - 3C). Some typical inquiries follow:

Q3: List and briefly describe the visual characteristics of each visually striking natural feature, the description should include the size, color, texture visibility and position in the landscape ? (Figure 3A)

Q5: Would you react differently to this landscape if the following feature did not exist in the landscape ? (Figure 3B)

Q7: How interesting or fascinating is the following visually striking feature ? (Figure 3C)

Question 5 is another check of the identified striking natural features. If the removal of a previously identified visually striking natural feature would not

make the user react differently to the landscape, then the previously identified visually striking natural feature may actually not be visually significant in the landscape. Reconsideration of the user's initial reaction or dropping of the previously identified visually striking natural feature are two choices provided for the user.

Since the number of questions or characteristics to be identified in the *natural feature subsystem* is only half of the number of questions in the *man-made subsystem* the decision tree and the traversing of it is much simpler. Figure 4 shows the decision tree for this subsystem with its 54 main and 39 leaf nodes. The explanatory rules for this decision tree are shown in Appendix F.

Spatial Organization Subsystem

The *spatial organization subsystem* is the second largest program with a total of 2000 lines code. Unlike the previous two subsystems, this program does not require the user to identify characteristics of landscape content and no loop programming is needed. This "straight" programming structure makes the *spatial organization subsystem* logically simpler than the previous two subsystems.

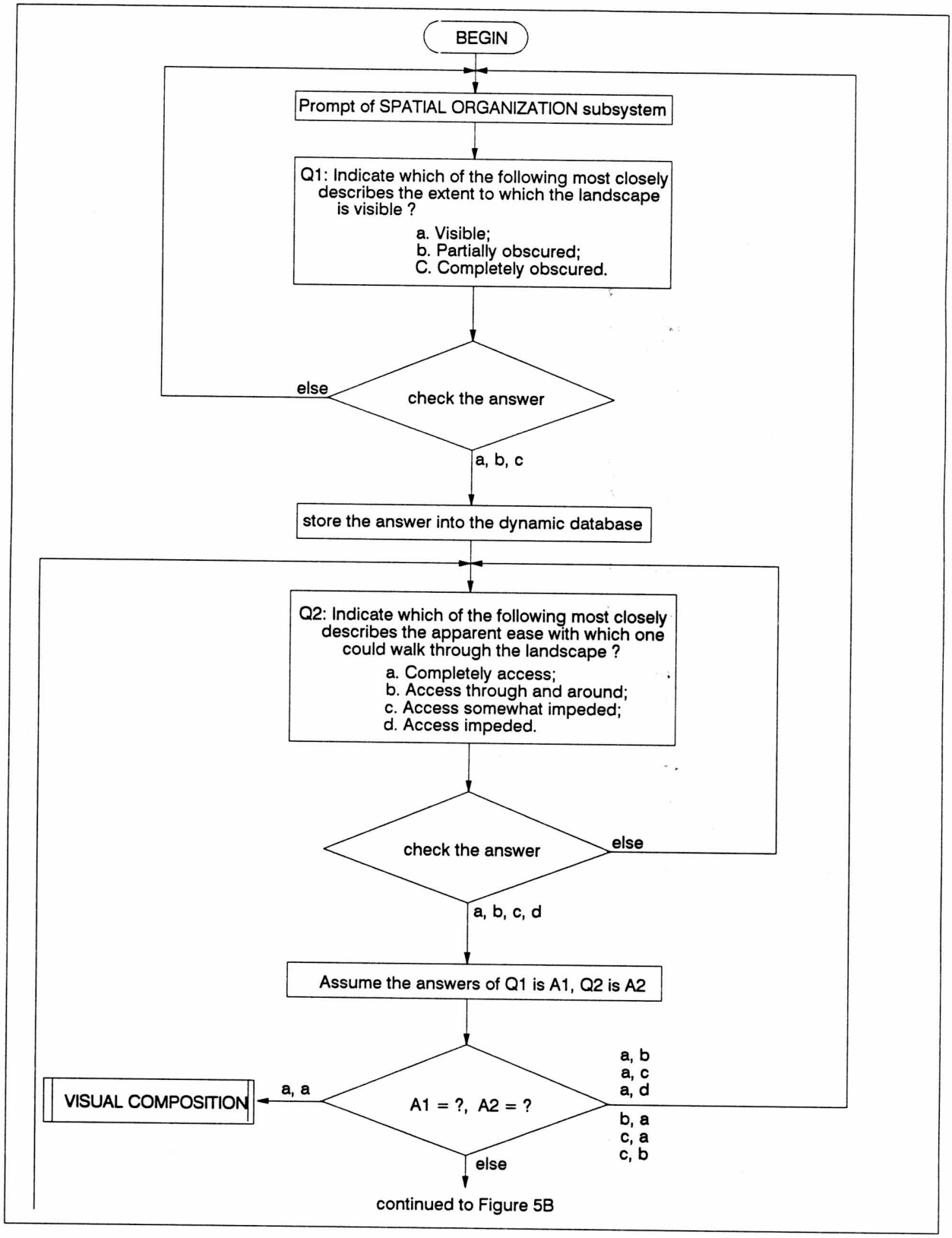


Figure 5A SPATIAL ORGANIZATION SUBSYSTEM FLOW DIAGRAM

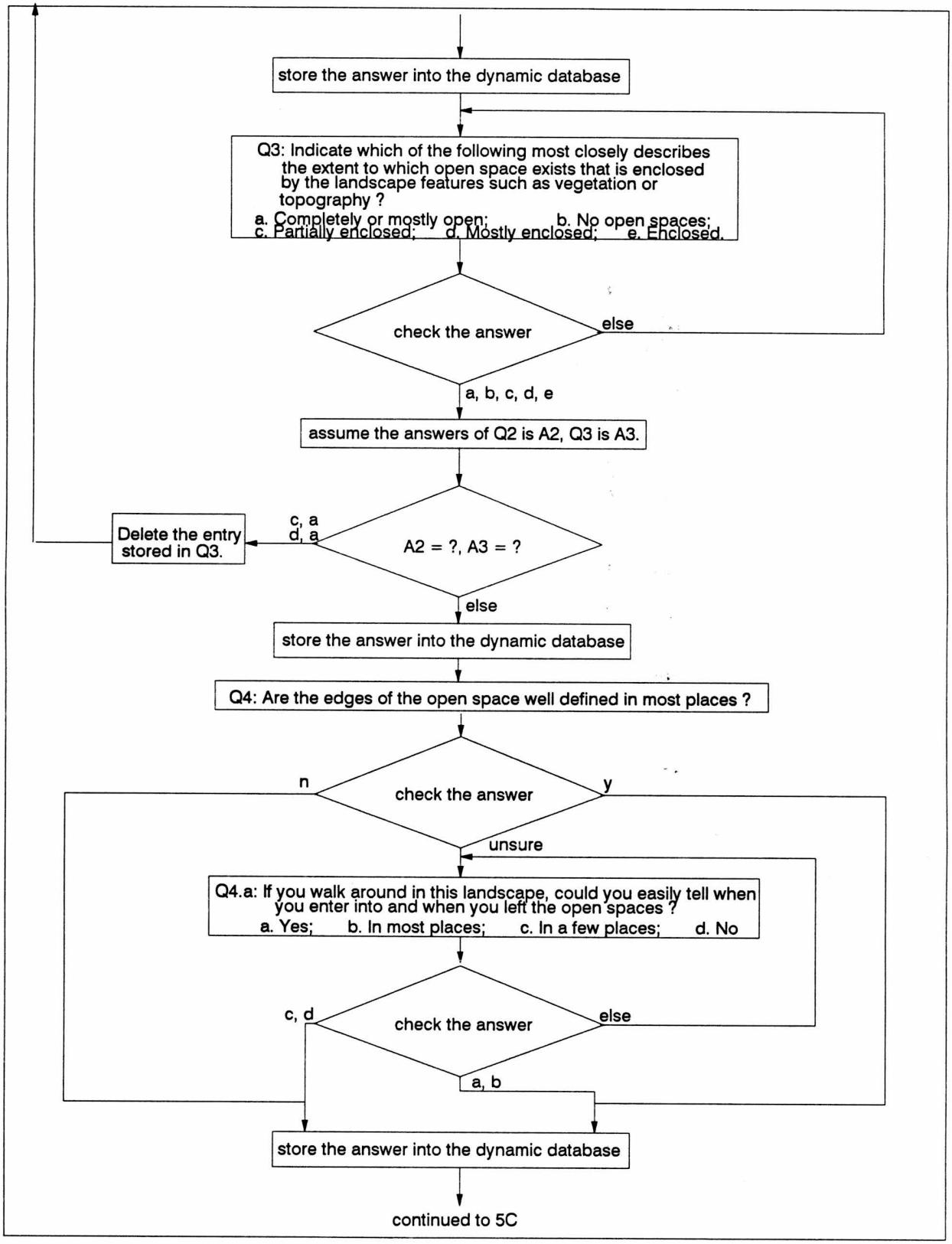


Figure 5B SPATIAL ORGANIZATION SUBSYSTEM FLOW DIAGRAM

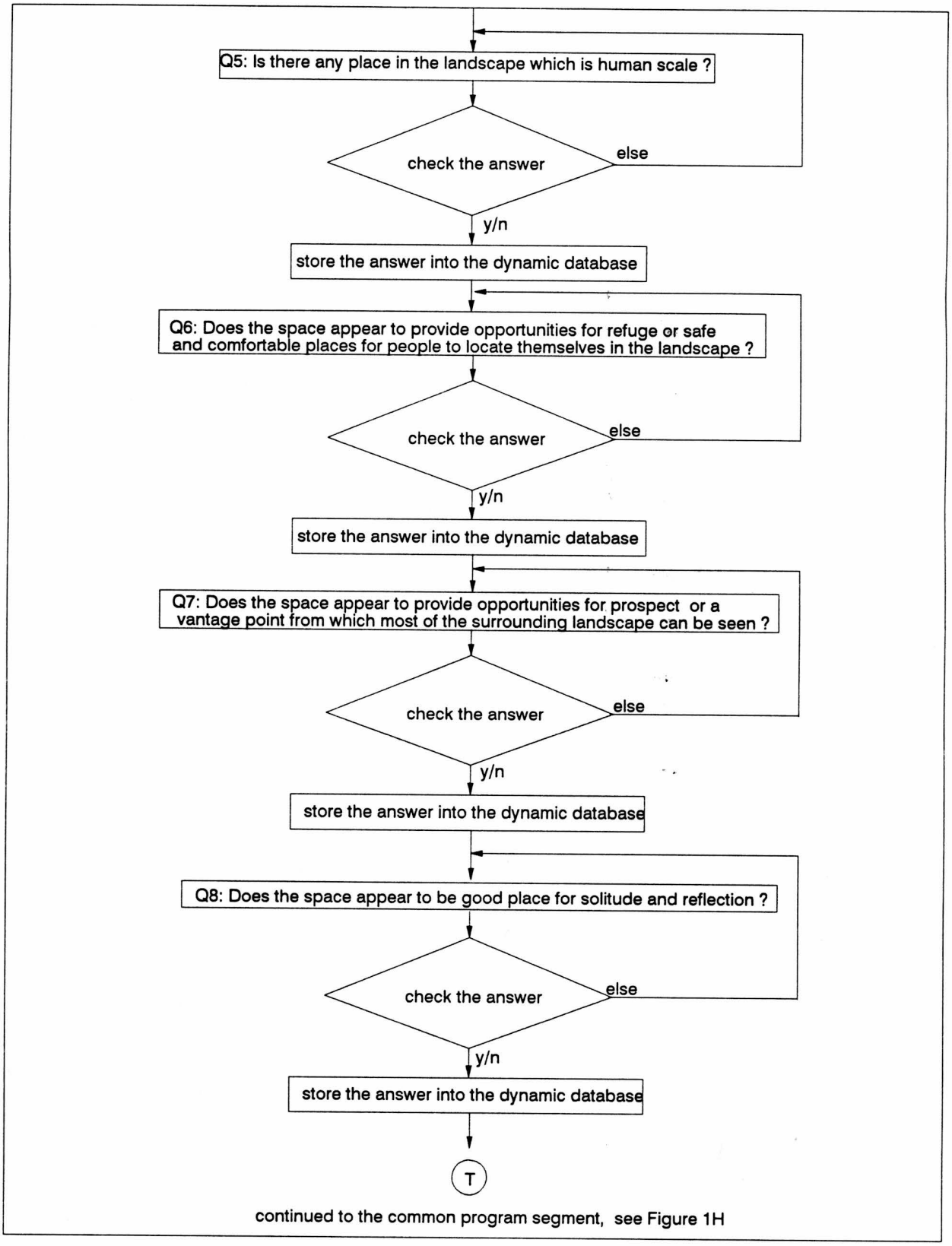


Figure 5C SPATIAL ORGANIZATION SUBSYSTEM FLOW DIAGRAM

There are, however, potential interactions between answers to questions 1, 2 and 3 in this subsystem.

Q1: Indicate which of the following most closely describe the extent to which the landscape is visible ?

a. Visible;

b. Partially obscured; and

c. Completely obscured. (Figure 5A)

Q2: Indicate which of the following most closely describe the apparent ease with which one could walk through the landscape ?

a. Completely access;

b. Access through and around;

c. Access somewhat impeded; and

d. Access impeded. (Figure 5A)

Q3: Indicate which of the following most closely describes the extent to which open space exists that is enclosed by the landscape features such as vegetation or topography ?

a. Completely or mostly open;

b. No open spaces;

c. Partially enclosed;

d. Mostly enclosed; and

e. Enclosed. (Figure 5B)

Assume A1, A2 and A3 stand for the answers to question 1, 2 and 3 respectively.

If

A1 = a and A2 = a

then

the spatial organization program will not have any effect on overall landscape visual quality. Therefore, the user needs to go to the visual composition program to investigate other aspects of the landscape which affect visual quality.

If, however;

A1 = a and A2 = b or A2 = c or A2 = d or

If

A1 = b and A2 = a or

If

A1 = c and A2 = a or A2 = b or

If

A2 = c and A3 = a or

If

A2 = d and A3 = a

then:

the answers are not consistent in the sense of analyzing the landscape dimensions. For example, a landscape that is identified as "completely visible" for visibility cannot also be identified as "impeded" for accessibility. Whenever such an inconsistency is found, the user is asked to go back to reconsider his/her previous answer or quit to the operating system.

Question 4 (Figure 5B) in this subsystem, "Are the edges of the open space well defined in most places ?" is unique in that there are 3 possible answers. In addition to "yes" or "no" answers, "unsure" is the third choice.

When "unsure" is selected as the answer, another question that is more specific about the setting is then asked to help the user resolve the uncertainty.

There are only eight questions in the spatial organization program (Figures 5A - 5C). However, its decision tree (Figure 6) is comparatively large (157 main and 85 leaf nodes) because many characteristics have roughly equal importance and therefore a greater number of combinations of potential answers are evaluated and more paths can be selected. The explanatory rules used in this decision tree are in Appendix G.

Visual Composition Subsystem

The *visual composition subsystem* is logically the simplest and physically the smallest program and has about 1000 lines of code. Since this program deals with landscape dimensions rather than the landscape content, loop programming is not required and the whole program is comparatively simple. Several questions are asked of the user. One inquiry requests the user to identify any visual pattern which may be present. If a visual pattern is identified, then an additional question is asked regarding the description of the identified visual pattern. The user is also asked to identify the appropriate level of visual complexity, vividness and striking visual pattern of the landscape

being accessed (Figures 7A - 7B). All the answers to these questions are independent of each other, and, therefore, the entire structure of the program is straight forward. Figure 8 presents the decision tree for this subsystem, which is comprised of 66 main and 52 leaf nodes. The explanatory rules used in this decision tree are in Appendix H.

Preliminary Subsystem Testing

Three subjects used the EVMS to perform landscape assessments for several landscape photographs. The three subjects were faculty members at Virginia Tech. One of the subjects (subject 2) was a professional landscape architect who was also the person who was knowledge engineered for the development of the subsystems. The other two subjects were forestry faculty members who had no formal design training. Three landscape photographs were selected for preliminary testing. They were chosen based on variability in landscape content. Figures 9 and 10 show the three landscapes which were selected. Two of the photographs represent in-stand landscape views (Photo numbers 1 and 2 in Figure 9) and one represents a scenic vista (Photo number 3 in Figure 10). The photographs were selected purposely for their heterogeneity in order to preliminarily evaluate how well the subsystems

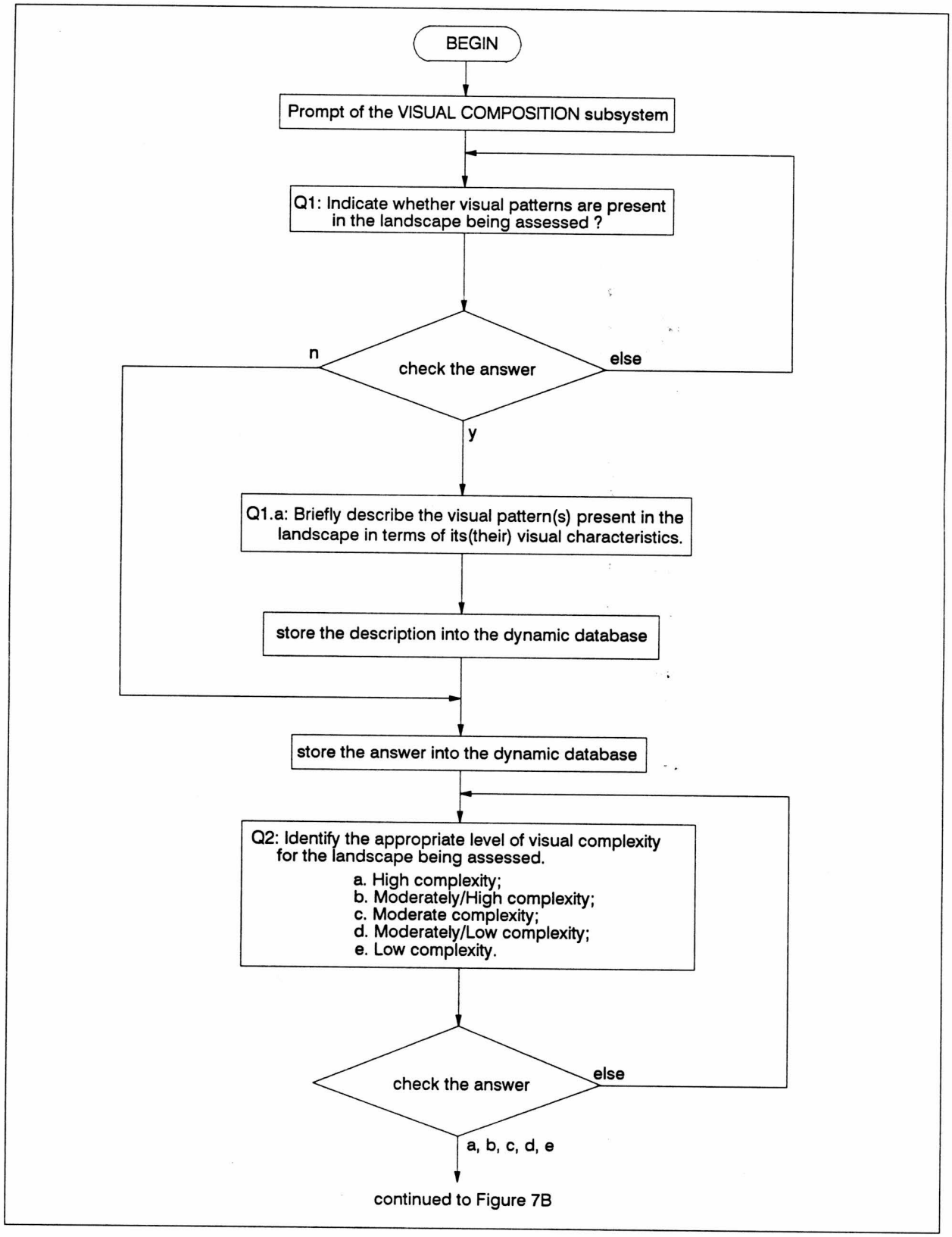


Figure 7A VISUAL COMPOSITION SUBSYSTEM FLOW DIAGRAM

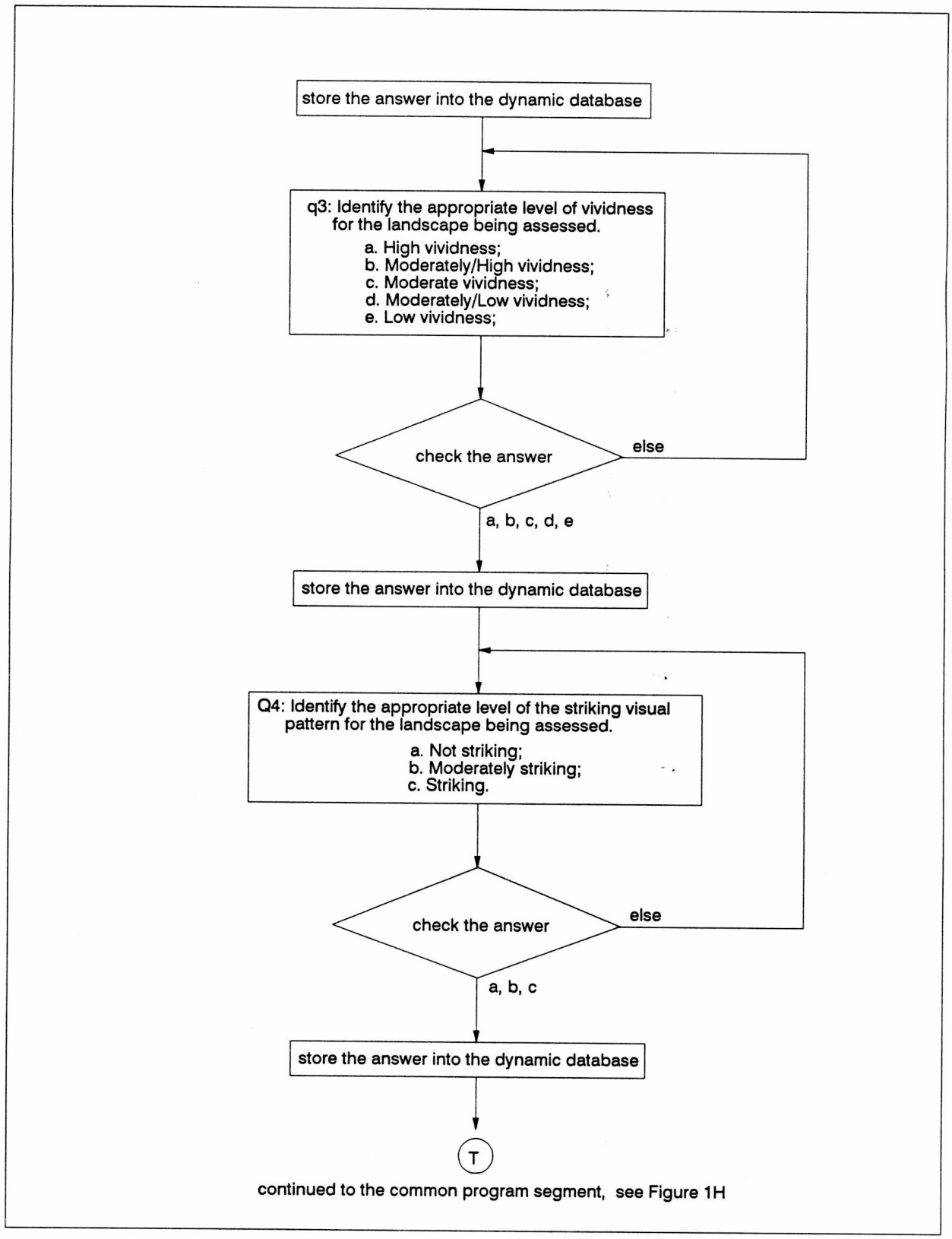


Figure 7B VISUAL COMPOSITION SUBSYSTEM FLOW DIAGRAM

photograph 1

photograph 2

Figure 9 Landscape photographs 1 and 2 for preliminary testing.

photograph 3

Figure 10 Landscape photograph 3 for preliminary testing.

handled a wide range of landscape views and content. All three subjects performed the landscape assessments utilizing the EVMS for all three photographs. The results of this preliminary testing are shown in the Table 1.

Overall, the results from the three subjects are remarkably similar. In most cases, even where there are some differences in the absolute assessments, these differences are minor in relative terms. For example, the difference between an assessment of "moderately low" and "slightly positive" (Table 2) is not dramatic since both of these evaluations signifies that the visual quality is about "middle", not greatly positive or negative. In fact six of the ten assessments are absolutely or relatively
identical. These include the *natural feature* assessments for photograph 3; the *spatial organization* assessments for all three landscapes; the *visual composition* assessment for landscape 1; and, the *man-made feature* assessment for photograph 3. Note that the *man-made feature subsystem* only applied to landscape photograph 3 because of the presence of a roadway in that photograph.

The subjects noted that while using the system that they were unsure of the definition of certain concepts such as "striking visual pattern" in the *visual composition subsystem*. Because of these problems in understanding

TABLE 1. RESULTANT VISUAL ASSESSMENTS FOR THREE USERS USING THE FOUR EVMS SUBSYSTEMS FOR THREE DIFFERENT LANDSCAPE PHOTOGRAPHS

NATURAL FEATURE SUBSYSTEM			
LANDSCAPE PHOTO	**SUBJECT 1**	**SUBJECT 2**	**SUBJECT 3**
1	Moderately high visual quality	None*	Moderately high visual quality
2	No effect on visual quality	None	No effect on visual quality
3	Outstanding visual quality	Outstanding visual quality	Outstanding visual quality
SPATIAL ORGANIZATION SUBSYSTEM			
1	Moderately low visual quality	Slightly positive effect on visual quality	Slightly positive effect on visual quality
2	No effect on visual quality	No effect on visual quality	No effect on visual quality
3	High visual quality	High visual quality	High visual quality
VISUAL COMPOSITION SUBSYSTEM			
1	Slightly positive effect on visual quality	Slightly positive effect on visual quality	Slightly positive effect on visual quality
2	Moderate visual quality	Moderately low visual quality	Low visual quality
3	Moderately low visual quality	Moderate visual quality	High visual quality
MAN-MADE COMPOSITION SUBSYSTEM			
3	No effect on visual quality	No effect on visual quality	No effect on visual quality

* Subject 2 did not find any visually significant natural feature in the landscape.

TABLE 2. THE DEFINITION OF ALL CONCLUSIONS IN EVMS.

CONCLUSION	DEFINITIONS
OUTSTANDING VISUAL QUALITY	This rating is reserved for landscapes with exceptionally high visual quality. These landscape would be regionally and or nationally significant. They usually contain interesting "natural features" that contribute to this rating. They would be what we think of when we think of "picture post card" landscapes. People would be attracted to these landscapes to be able to view them. typically these landscapes are managed in a manner which preserves the visual quality of the landscape.
HIGH VISUAL QUALITY	This rating is for those landscape have high quality scenic value. This may be due to "man-made or natural features" contained in the landscape or to the "arrangement of spaces" in the landscape that causes the landscape to be visually interesting or a particularly comfortable place for people. These are often landscapes which have high potential recreational activities in which the visual experience is important and are typically managed in a manner that gives priority or preserving the visual quality of the landscape.
MODERATELY HIGH VISUAL QUALITY	This rating is for landscapes which have above average scenic value, but are not of high scenic value. The scenic value of these landscapes may be due to "man-made or natural features" contained in the landscape, to the "arrangement of spaces" in the landscape or to the two dimensional visual attributes of the landscape. These landscapes often have considerable recreational potential and visual quality is an important management concern.
MODERATE VISUAL QUALITY	These are landscapes which have average scenic value. They usually lack significant "man-made and natural features." Their scenic value is primarily a result of the "arrangement of spaces" contained in the landscape and two dimensional visual attributes of the landscape. These landscapes also often have considerable recreation potential and visual quality is a management consideration.
MODERATELY LOW VISUAL QUALITY	These are landscapes which have below average scenic value, but not low scenic value. They may contain visually discordant man-made alterations, but the landscape is not dominated by these features. They often lack "spatial arrangements" which provide comfortable places for people and provide little interest in terms of two dimensional visual attributes of the landscape. These landscapes often have limited recreation potential or provide opportunities for recreational activities in which the visual experience in less important. Management concerns for visual quality are usually limited minimizing adverse visual impacts of various resources management activities.

TABLE 2. (Continued) THE DEFINITION OF ALL CONCLUSIONS IN EVMS.

LOW VISUAL QUALITY	These are landscape which have low scenic value. The landscape is often dominated by visually discordant man-made alterations; or they are landscape with "spatial arrangements" which do not provide comfortable places for people and lack interest in terms of two dimensional visual attributes. These landscapes often have little recreation potential. Management concerns for visual quality either address rehabilitation of visually discordant man-made alterations or are limited minimizing adverse visual impacts of various resources management activities.
SLIGHT POSITIVE EFFECT ON VISUAL QUALITY	This is the result when one component of the visually quality determination system has an effect on visual quality, but not a primary effect. In this case the primary determinant of visual quality is determined by another component of the system and the component producing this result has a "slight positive" modifying effect. For example, the primary visual quality determination produced by another component of the system may be "moderate visual quality" and the modifying effect of this component would raise the scenic quality of the landscape being evaluated to "moderately high" scenic quality.
NEGATIVE EFFECT ON VISUAL QUALITY	This is the result when one component of the visually quality determination system has an effect on visual quality, but not a primary effect. In this case the primary determinant of visual quality is determined by another component of the system and the component producing this result has a "slight negative" modifying effect. For example, the primary visual quality determination produced by another component of the system may be "moderate visual quality" and the modifying effect of this component would lower the scenic quality of the landscape being evaluated to "moderately low" scenic quality.
NO EFFECT ON VISUAL QUALITY	This is the result when one component of the visually quality determination system has no effect on visual quality. In this case the determination of visual quality is made by another component of the system. For example, there may be no "man-made" alterations in the landscape being evaluated. In this case the visual quality of the landscape will be determined by one or more of the other components.

what certain concepts and words meant relative to the inherent system definitions, there may have been some variance in how the subjects responded. Therefore, this may have caused problematic variations in the final visual assessment of some of the landscape photographs for some subsystems.

The largest difference in final assessments showed up in the use of the *visual composition subsystem* for photograph 3. The results range from "moderately low visual quality" to "high visual quality". Again, it was observed during the testing session that there may have been some problems in understanding what the definitions of certain evaluation terms were.

CHAPTER V

CONCLUSIONS

Advantages of the EVMS

The EVMS is a quite different visual assessment system from a traditional VMS assessment or psychophysical model prediction. Two prominent characteristics of the EVMS are that it provides explanations for visual assessments and retains this information during the reasoning procession. A traditional VMS or psychophysical model collapses information. The traditional VMS, as used, for example, by the United States Forest Service collapses information on evaluation criteria into a few cells of decision matrices. As the steps toward a final evaluation are made, information is inherently "lost" in terms of the relevant and complex decisions which were made for each classifications of a landscape into a particular category for an evaluation criterion (for example, visual sensitivity, variety class and so on). Likewise a traditional psychophysical model, while not "classifying" landscapes, is developed to make predictions of scenic preference or visual quality from variables which are often selected from a vast pool for their predictive rather than "genuine" explanatory ability. The resulting mathematical expressions may

predict well but often explanations for the complex variable interactions are not understood or cannot be explained.

The EVMS can be easily modified. For example, a reasoning rule can be easily modified or added in a subsystem without affecting the other subsystems. This independence is not only important in developing, maintaining and testing the entire system, but also provides the user great convenience for updating the system. On the contrary, if a traditional VMS system step or function is changed, all of the system matrices used to arrive at a final assessment should also be modified correspondingly since the decision or matrix leads to evaluation matrices are dependent. Psychophysical models also suffer from a similar problem. Changing a variable in a psychophysical model would necessarily result in an entirely new model with possibly very different predictive power.

It is possible that the EVMS can be used both on and off site. The user can use the system to evaluate landscapes by using the photographs of landscapes. It may also be possible to use the system while actually on-site via the use of a small lap-top computer. On-site use of the system might provide "comprehensive" assessments since the evaluation would be influenced by the total environment rather than just a visual representation of it, despite the fact

that the present EVMS intentionally focuses on visual stimuli as a basis for the assessments.

The EVMS is more comprehensive than a traditional VMS or psychophysical model. The system is comprised of a large number of decision rules which are logically linked and provide explanations for a visual assessment. The rules and explanations consider both psychological rationale stemming from an expert's experience and knowledge of "facts" which may have come from a variety of sources including a multitude of research results and interactions with public clientele, and from his knowledge of physical variable interactions which give rise to degrees of scenic quality. The traditional VMS or statistical prediction models are narrower, in that many fewer variable, conditions and influences are considered or used in "arriving at a conclusion". If anything, scenic quality is a complex criterion and evaluation or prediction of it cannot be inherently simple.

Potential Disadvantages of the EVMS

Since the entire system is developed to represent an expert designer's point of view, it is not known how well this represents general public evaluations of landscapes. Also, evaluations performed by the EVMS are more

subjective compared with a statistical prediction method such as a psychophysical model. For example, the EVMS system does not use objective variables such as "basal area", "trees per acre" and so on. Instead, the EVMS system requires the user to make subjective assessments such as answering the question "Do you think the natural pattern is visually striking?". Therefore, the system is not relatable to objective measures of landscape attributes and therefore may not be simply integrated directly into larger computerized forest planning and management systems which rely on mathematical model representations of forest processes and product output.

All four subsystems act independently and no interaction among them has been taken into account. This inevitably needs to be addressed since the interaction among the four subsystems is an important factor in the overall evaluation of the landscape.

Recommendations

The EVMS is currently in a prototype version. Its performance, based upon preliminary testing, seems to be quite good with regard to generating consistent evaluations across users. However, more work is required to improve precision and to test the external validity of the EVMS. First, the inconsistency

in assessments generated by different users may well be substantially rectified by the insertion of a set of archived definitions of concepts and terms used by the system. These definitions should cause greater consistency of understanding between users and therefore aid immeasurably in system use. It also should be possible to integrate visual images which could serve as examples or standards for difficult-to-understand concepts such as "mystery" or other psychological constructs.

The system needs several levels of validation and comparative studies. On-site versus off-site performance of the EVMS should be evaluated. Assessments generated by the EVMS should be compared to those offered by a sample of landscape design professionals. A comparison of EVMS assessments to assessments or predictions made by other methods would also likely be beneficial. For example, the results of EVMS assessments could be compared and contrasted to the results of other assessment systems such as the USFS VMS or to prediction methods such as psychophysical models.

Finally, the four current subsystems must be integrated so as to provide a consolidated assessment. The interactions among the four subsystems must be discovered through further knowledge engineering.

LITERATURE CITED

Arthur, L.M., Daniel T.C. and Boster R.S. 1977. Scenic Assessment: An overview. Landscape Planning. 4:109-129.

Briggs, D.J. and France, J. 1980. Landscape Evaluation: A Comparative Study. Journal of Environmental Management. 10: 263-275.

Buhyoff, G.J. and Leuschner, W.A. 1978. Estimating Psychological Disutility From Damaged Forest Stands. Forest Science. 24:424-432.

Buhyoff, G.J. and Hull, R.B. 1980. Computation of Law of Comparative Judgement Scales on a Microcomputer. Behavior Research Methods and Instrumentation. 12(4):465.

Buhyoff, G.J., Hull, R.B., Lien, J.N. and Cordell, H.K. 1986. Prediction of Scenic Quality for Southern Pine Stands. Forest Science. 32(3):769-778.

Clark, R.N., Hendee, J.C. and Campbell, F.L. 1971. Values, Behavior and Conflict in Modern Camping Culture. Journal of Leisure Research. 3(3):143-159.

Craik, K.H. 1972. Psychological Factors in Landscape Appraisal. Environment and Behavior. 4(3):255-266.

Daniel, T.C., Boster, R.S. 1976. Measuring Landscape Esthetics: The Scenic Beauty Estimation Method. USDA Forest Service Research Paper RM-167. 66 pgs.

Daniel, T.C., Vining, J. 1984. Methodological Issues in the Assessment of Landscape Quality. in Human Behavior and Environment. Altman, I. and Wohlwill, J.F. (editor). Plenum Press: NY. pgs 39-84.

Gratzer, M.A. and McDowell, R.D. 1971. Adaption of an Eye Movement Recorded to Esthetic Environmental Mensuration. Storrs Agricultural Experiment Station Research Report. No. 36. 29 pgs.

Hayes-Ross, F. 1987. Rule-based System. in Encyclopedia of Artificial Intelligence. Shapiro, S.C. (editor). pp:963-973.

Henschen, L. 1987. Reasoning. in Encyclopedia of Artificial Intelligence. Shapiro, S.C. (editor). pp: 822-827.

Hull R.B., Buhyoff, G.J and Daniel, T.C. 1984. Measurement of Scenic Beauty: The Law of Comparative Judgement and Scenic Beauty Estimation Procedures. Forest Science. 30(4): 1084-1096.

Hull, R.B and Buhyoff, G.J. 1981. On the Law of Comparative Judgement: Scaling With Intransitive Observers and Multidimensional Stimuli. Educational and Psychological Measurement. 41:1083-1089.

Hull, R.B. and Buhyoff, G.J. 1986. The Scenic Beauty Temporal Distribution Method: An attempt to Make Scenic Beauty Assessments Compatible With Forest Planning Efforts. Forest Science. 32(2):271-286.

Jackues, D.L. 1980. Landscape Appraisal: The Case for a Subjective Theory. Journal of Environmental Management. 10: 107-113.

Kaplan, S. and Wendt, J.S. 1972. Preference and the Visual Environment: Complexity and Some Alternatives. Proceedings of the 3rd Environmental Design Research Association Conference.

Kaplan, S. and Kaplan, R. 1982. Cognition and Environment: Functioning in an Uncertain World. Praeger Publishers, New York, New York. PP 287.

Kastner, J.R., Weiss, S.M. and Kulikowski, C.A. 1982. Treatment Selection and Explanation in Expert Medical Consultation Application to a Model of Ocular Herpes Simplex. Proceedings of MEDCOMP82. 420-427.

Lien, J.N. and Buhyoff, G.J. 1986. Extension of Visual Quality Models for Urban Forests. Journal of Environmental Management. 22:245-254.

Litton, R.B. Jr. 1972. Aesthetic Dimensions of the Landscape. in Natural Environments: Studies in Theoretical and Applied Analysis. Krutilla J.V. (editor). John Hopkins Univ. Press, Baltimore.

McGuire, J.R. 1979. Managing the Forest Landscape for Public Expectations. in Proceedings of Our National Landscape. A Conference on Applied Techniques for Analysis and Management of the Visual Resource. Elsner, G. H. and Smardon R.C. (editor). pp:16-19.

Peterson, G.S. and Neumann, E.S. 1969. Modeling and Predicting Human Response to the Visual Recreation Environment. Journal of Leisure Research. 1(3):219-237.

Roach, J.W. and Deighan J. 1990. VPI Prolog User's Manual. Virginia Polytechnic Institute and State University, Department of Computer Science, Blacksburg, VA.

Schroeder, H.W. and Cannon, W.N. Jr. 1983. The Esthetic Contribution of Trees to Residential Streets in Ohio Towns. Journal of Arboriculture. 9(9):237-243.

Shortliffe, E.H. 1976. Computer Based Medical Consultations: MYCIN. American Elsevier Publishing Co., Inc. 264pgs.

Stock, M. 1987. AI and Expert System: An Overview. AI Applications in Natural Resource Management. 1(1): 9-17.

Swartout, W.R. 1977. A Digitalis Therapy Advisor With Explanations. Proceedings of the Fifth International Conference on Artificial Intelligence. IJCAI, Cambridge, MA. pp:819-825.

Swartout, W.R. 1983. Xplain: A System for Creating and Explaining Expert consulting Programs. Artificial Intelligence. 21(3):285-325.

Swartout, W.R. 1987. Explanation. in Encyclopedia of Artificial Intelligence. Shapiro S.C. (editor). pp:298-300.

USDA Forest Service. 1974. National Forest Landscape Management System. Vol. 1, Ch 2. Agricultural Handbook 462. 47 pgs.

USDA Forest Service. 1974. National Forest Landscape Management System. Vol. 1. Agricultural Handbook 434. 76p.

Wellman, J.D. and Buhyoff, G.J. 1980. Effects of Regional Familiarity on Landscape Preferences. Journal of Environmental Management. 11:1-6.

Winograd, T. 1971. A Computer Program for Understanding Natural Language. MIT Artificial Intelligence Laboratory. TR-17. Cambridge, MA.

Wohlwill, J.F. 1966. The Physical Environment: A Problem for A Psychology of Stimilation. Journal Social Issues. 22(4):29-38.

Wohlwill, J.F. 1968. Amount of Stimulus Exploration and Preference as Differential functions of Stimulus Complexity. Perception and Psychophysics. 4:307-312.

Zube, E.H., Pitt, D.G. and Anderson, T.W. 1974. Perception and Measure of Scenic Resources in the Southern Connecticut River Valley. Institute for Man and His Environment. Publication No. R-74-1, Amherst, Mass. 191 pgs.

Zube, E.H., Brush, R.O. and Fabos, J. 1975. Landscape Assessment: Values, Perceptions And Resources. Dowden, Hutchinson and Ross, Stroudsburg, PA. 367 pgs.

Zube, E.H., Sell, J.L. and Taylor, J.G. 1982. Landscape Perception: Research, Application and Theory. Landscape Planning. 9:1-34.

APPENDIX A

CODE OF MAN-MADE FEATURE SUBSYSTEM

```
(assert
  ((doit) if
   (man)
   (go_exp)
  )
)


(assert
  ((man) if
   (system cls)
   (println "*****************************************************")
   (println "*                                                   *")
   (println "*      Visual Quality Component I: Landscape Content *")
   (println "*             Part A -- Man-made Feature            *")
   (println "*                                                   *")
   (println "*****************************************************")
   (nl)(nl)

   (println "To evaluate man-made features, you need to answer the following questions.")
   (nl)(nl)
   (open stream "TXB1:" read)

   (println "Q1: Are any man-made alterations of the landscape or built")
   (println "structures present? Include only those alterations or built")
   (println "structures that would be recognizable by the public as caused")
   (println "by man.")
   (nl)(nl)
   (println "          PLEASE ANSWER Y/N")
   (getline stream ?alteration)
   (continue ?alteration)
   (nl)(nl)

   (system cls)
   (println "Q2: Please indicate how many different types of man-made ")
   (println "alterations of the landscape or different types of built ")
   (println "structures are visible.")
   (nl)(nl)
   (println "      PLEASE INPUT THE NUMBER OF ALTERATIONS")
   (getline stream ?no_of_alter)
   (:= ?no_alter (strtonum ?no_of_alter))
   (assert ((no_alter ?no_alter)))
   (nl)(nl)
   (whether ?no_alter)
  )
)
```

```
(assert
  ((continue ?c) if
   (or (= ?c n) (= ?c N))
   (println "You did not identify any man-made alteration or built structure, the visual")
   (println "quality is therefore not being affected.")
   (nl)(nl)
   (println "Do you want to QUIT or START again, input (q/s).")
   (getline stream ?a)
   (again ?a)
  )

  ((continue ?c) if
   ( or (= ?c y) (= ?c Y))
  )
)


(assert
  ((again q) if
   (quit)
  )

  ((again s) if
   (close stream)
   (doit)
  )

  ((again ?a) if
   (!= ?a q)
   (!= ?a s)
   (println "Do you want to QUIT or START again, input (q/s).")
   (getline stream ?b)
   (again ?b)
  )
)


(assert
  ((whether ?count) if
   (> ?count 5)
   (more_t_five)
  )

  ((whether ?count) if
   (<= ?count 0)
   (system cls)
   (expla man a ?expla)
   (assert ((page 20)))
   (print_exp ?expla)
```

```
  (nl)(nl)
  (println "FINAL CONCLUSION:  No effect on visual quality.")
  (retract ((page ?)))
  (nl)(nl)
  (println "Do you want to QUIT or START again?   Input (q/s)")
  (getline stream ?again)
  (do_over ?again)
 )

 ((whether ?count) if
  (> ?count 0)
  (< = ?count 5)
  (pre_name ?count)
  (pre_desc ?count)
  (pre_indi ?count)
  (pre_tech ?count)
  (pre_awe ?count)
  (pre_hist ?count)
  (pre_huma ?count)
  (pre_natu ?count)
  (pre_shel ?count)
  (pre_pros ?count)
  (pre_refl ?count)
  (pre_visb ?count)
  (larger ?count)
 )
)


(assert
 ((do_over q) if
  (quit)
 )

 ((do_over s) if
  (retract ((no_alter ?)))
  (close stream)
  (doit)
 )

 ((do_over ?a) if
  (! = ?a q)
  (! = ?a s)
  (nl)(nl)(nl)(nl)
  (println "Your input is not appropriate, input (q/s) only.")
  (getline stream ?b)
  (do_over ?b)
 )
)
```

```
(assert
  ((larger 1) if
   (visb 1 ?collect)
   (y_or_n_2 ?collect)
  )

  ((larger ?multy) if
   (> ?multy 1)
   (system cls)
   (println "Q15: When viewed collectively, how visually apparent are all the man-made")
   (println "alterations and built structure?")
   (nl)
   (paragra)
   (nl)
   (getline stream ?collect)
   (y_or_n_2 ?collect)
  )
)


(assert
  ((more_t_five) if
   (system cls)
   (println "Q3: When viewed collectively, how visually apparent are all the man-made")
   (println "alterations and built structure?")
   (nl)
   (paragra)
   (nl)
   (getline stream ?group)
   (y_or_n_2 ?group)
  )
)


(assert
  ((show_common ?num) if
   (println "You have identified the following man-made alterations or built")
   (println "structures being present in the landscape being assessed.")
   (nl)
   (show_names 1 ?num)
  )
)


(assert
  ((show_names ? 0) if
   (nl)
  )
```

```
  ((show_names ?up ?down) if
   (print "        ")
   (print ?up)
   (print ". ")
   (name ?up ?name)
   (println ?name)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_names ?new_up ?new_down)
  )
)


(assert
  ((pre_name ?end_cond) if
   (system cls)
   (println "Q3: Please list and name EACH man-made alteration.")
   (nl)(nl)
   (name_loop 1 ?end_cond)
  )
)


(assert
  ((pre_desc ?end_cond) if
   (system cls)
   (println "Q4: Please briefly describe the visual characteristics of each")
   (println "man-made alteration. The description should include the size,")
   (println "color, postion in the landscape and any other noteworthy visual")
   (println "aspects of the man-made alteration or built structure.")
   (nl)(nl)
   (desc_loop 1 ?end_cond)
  )
)


(assert
  ((pre_indi ?end_cond) if
   (system cls)
   (println "Q5: Any indigenous materials such as rock, clay or wood used in")
   (println "the construction of built structures should also be listed.")
   (nl)(nl)
   (indi_loop 1 ?end_cond)
  )
)


(assert
```

```
  ((pre_tech ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q6: Is the alteration or built structure that appear to have")
   (println "resulted from INDUSTRIAL, QUASI-INDUSTRIAL or TECHNOLOGICAL use")
   (println "of the landscape?")
   (nl)(nl)
   (tech_loop 1 ?end_cond)
  )
)


(assert
  ((pre_awe ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q7: Is the alteration or built structure that appear to be AWE")
   (println "INSPIRING because of its size or the technology involved?")
   (nl)(nl)
   (awe_loop 1 ?end_cond)
  )
)


(assert
  ((pre_hist ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q8: Is the alteration or built structure that appear to be")
   (println "HISTORICAL or resulted from PAST CULTURES with different")
   (println "technologies than exist today?")
   (nl)(nl)
   (hist_loop 1 ?end_cond)
  )
)


(assert
  ((pre_huma ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q9: Is the alteration or built structure HUMAN SCALE?")
   (nl)(nl)
   (huma_loop 1 ?end_cond)
  )
)


(assert
```

```
  ((pre_natu ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q10: Is the alteration or built structure that appear to be")
   (println "constructed predominantly by NATURAL MATERIALS such as wood,")
   (println "rock or adobe?")
   (nl)(nl)
   (natu_loop 1 ?end_cond)
  )
)


(assert
  ((pre_shel ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q11: Is the alteration or built structure that appear to offer")
   (println "POTENTIAL FOR SHELTER")
   (nl)(nl)
   (shel_loop 1 ?end_cond)
  )
)


(assert
  ((pre_pros ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q12: Does the alteration or built structure that appear to ")
   (println "provide OPPORTUNITIES FOR PROSPECT or a VANTAGE POINT(S) from")
   (println "which to see the adjacent landscape?")
   (nl)(nl)
   (pros_loop 1 ?end_cond)
  )
)


(assert
  ((pre_refl ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q13: Does the alteration or built structure that appear to be")
   (println "good place for solitude and reflection?")
   (nl)(nl)
   (refl_loop 1 ?end_cond)
  )
)
```

```
(assert
  ((pre_visb ?end_cond) if
   (system cls)
   (println "Q14: If more than one man-made alteration exists, imagine how the landscape")
   (println "would appear if one of alterations were the only alteration, how visually")
   (println "apparent would it be(including only one identified alteration case)?")
   (nl)
   (visb_loop 1 ?end_cond)
  )
)



(assert
  ((name_loop ? 0))
  ((name_loop ?up ?down) if
   (print "Please input the name for the alteration NO. ")
   (print ?up)
   (println ".")
   (getline stream ?name)
   (nl)(nl)(nl)(nl)
   (assert ((name ?up ?name)))
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (name_loop ?new_up ?new_down)
  )
)


(assert
  ((desc_loop ? 0))
  ((desc_loop ?up ?down) if
   (print "INPUT FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (nl)
   (println "BE SURE TO BRACKET YOUR DESCRIPTION BY '(' & ')'.")
   (println "IF YOU DO NOT HAVE ANYTHING TO INPUT, TYPE IN () FOR A EMPTY INPUT.")
   (println "BE SURE NOT TO USE ANY '.' SIGN INSIDE THE BRACKET.")
   (read ?desc)
   (nl)(nl)(nl)(nl)
   (assert ((desc ?up ?desc)))
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (desc_loop ?new_up ?new_down)
  )
)
```

```
(assert
  ((indi_loop ? 0))
  ((indi_loop ?up ?down) if
   (print "INPUT FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (nl)
   (println "BE SURE TO BRACKET YOUR DESCRIPTION BY '(' & ')'.")
   (println "IF YOU DO NOT HAVE ANYTHING TO INPUT, TYPE IN () FOR A EMPTY INPUT.")
   (println "BE SURE NOT TO USE ANY '.' SIGN INSIDE THE BRACKET.")
   (read ?indi)
   (nl)(nl)(nl)(nl)
   (assert ((indi ?up ?indi)))
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (indi_loop ?new_up ?new_down)
  )
)


(assert
  ((tech_loop ? 0))
  ((tech_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?tech)
   (nl)(nl)(nl)(nl)
   (y_or_n ?up tech ?tech)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (tech_loop ?new_up ?new_down)
  )
)


(assert
  ((awe_loop ? 0))
  ((awe_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?awe)
   (nl)(nl)(nl)(nl)
   (tech ?up ?tech)
   (check ?tech ?awe ?feed_back)
```

```
  (awe_1_loop ?up ?down ?awe ?feed_back)
 )
)


(assert
 ((check n y n) if
  (cut)
 )

 ((check ? n y) if
  (cut)
 )

 ((check y y y) if
  (cut)
 )

 ((check ? ? wrong))
)


(assert
 ((awe_1_loop ?up ?down ?awe y) if
  (assert ((awe ?up ?awe)))
  (:= ?new_up (+ ?up 1))
  (:= ?new_down (- ?down 1))
  (awe_loop ?new_up ?new_down)
 )

 ((awe_1_loop ?up ?down ? n) if
  (system cls)
  (println "The alteration or structure you identified as AWE INSPIRING is")
  (println "not even being identifed as INDUSTRIAL or QUASI-INDUSTRIAL or")
  (println "TECHNOLOGICAL. So your answer is not consistent.")
  (nl)
  (println "To help your input, here is the inconsistent you made:")
  (name ?up ?name)
  (print "    ALTERATION NAME  :  ")
  (println ?name)
  (print "    INDUSTRIAL       :  ")
  (println "n")
  (print "    AWE-INSPIRING    :  ")
  (println "y")
  (nl)
  (println "NOTE: You need to do it over for ALL the objects !")
  (nl)(nl)(nl)(nl)(nl)
  (println "          **********   hit RETURN to continue   **********")
  (getline stream ?any)
```

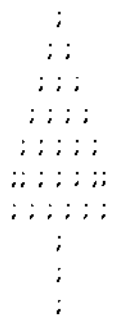
```
  (:= ?t (+ ?up ?down))
  (:= ?s (- ?t 1)) ; ?up + ? down -1 = ?no_alter. Always valid.
  (pre_awe ?s)
 )

 ((awe_1_loop ?up ?down ? wrong) if
  (println "Your input is wrong, INPUT ONLY 'y or n'. Now, do it over !")
  (getline stream ?awe)
  (tech ?up ?tech)
  (check ?tech ?awe ?feed_back)
  (awe_1_loop ?up ?down ?awe ?feed_back)
 )
)


(assert
 ((hist_loop ? 0))
 ((hist_loop ?up ?down) if
  (print "PLEASE ANSWER y/n FOR ALTERATION: ")
  (name ?up ?name)
  (print ?name)
  (println ".")
  (getline stream ?hist)
  (nl)(nl)(nl)(nl)
  (y_or_n ?up hist ?hist)
  (:= ?new_up (+ ?up 1))
  (:= ?new_down (- ?down 1))
  (hist_loop ?new_up ?new_down)
 )
)


(assert
 ((huma_loop ? 0))
 ((huma_loop ?up ?down) if
  (print "PLEASE ANSWER y/n FOR ALTERATION: ")
  (name ?up ?name)
  (print ?name)
  (println ".")
  (getline stream ?huma)
  (nl)(nl)(nl)(nl)
  (y_or_n ?up huma ?huma)
  (:= ?new_up (+ ?up 1))
  (:= ?new_down (- ?down 1))
  (huma_loop ?new_up ?new_down)
 )
)
```

```
(assert
  ((natu_loop ? 0))
  ((natu_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?natu)
   (nl)(nl)(nl)(nl)
   (y_or_n ?up natu ?natu)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (natu_loop ?new_up ?new_down)
  )
)


(assert
  ((shel_loop ? 0))
  ((shel_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?shel)
   (nl)(nl)(nl)(nl)
   (y_or_n ?up shel ?shel)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (shel_loop ?new_up ?new_down)
  )
)


(assert
  ((pros_loop ? 0))
  ((pros_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?pros)
   (nl)(nl)(nl)(nl)
   (y_or_n ?up pros ?pros)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (pros_loop ?new_up ?new_down)
  )
)
```

```
(assert
  ((refl_loop ? 0))
  ((refl_loop ?up ?down) if
   (print "PLEASE ANSWER  y/n  FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?refl)
   (nl)(nl)(nl)(nl)
   (y_or_n ?up refl ?refl)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (refl_loop ?new_up ?new_down)
  )
)


(assert
  ((visb_loop ? 0))
  ((visb_loop ?up ?down) if
   (paragra)
   (print "   FOR:   ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?visb)
   (y_or_n_1 ?up visb ?visb)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (nl)(nl)(nl)(nl)(nl)
   (visb_loop ?new_up ?new_down)
  )
)



(assert
  ((y_or_n ?up ?pred y) if
   (assert ((?pred ?up y)))
   (nl)
   (cut)
  )

  ((y_or_n ?up ?pred n) if
   (assert ((?pred ?up n)))
   (nl)
   (cut)
  )
```

```
  ((y_or_n ?up ?pred ?) if
   (println "Your input is not correct, please do it over!")
   (nl)
   (print "PLEASE ANSWER y/n FOR ALTERATION: ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?repeat)
   (y_or_n ?up ?pred ?repeat)
  )
)



(assert
  ((y_or_n_1 ?up ?pred ?range) if
   (> = ?range a)
   (< = ?range e)
   (assert ((?pred ?up ?range)))
   (cut)
  )

  ((y_or_n_1 ?up ?pred ?) if
   (system cls)
   (println "Your input is not correct, please do it over!")
   (nl)
   (paragra)
   (print "   FOR:   ")
   (name ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?repeat)
   (y_or_n_1 ?up ?pred ?repeat)
  )
)



(assert
  ((y_or_n_2 ?range) if  ; The difference between y_or_n_1 and y_or_n_2 is
   (> = ?range a)        ; that y_or_n_2 does not print out any name.
   (< = ?range e)
   (assert ((coll 1 ?range)))
   (cut)
  )

  ((y_or_n_2 ?) if
   (system cls)
   (println "Your input is not correct, please do it over!")
   (nl)
```

```
   (paragra)
   (getline stream ?repeat)
   (y_or_n_2 ?repeat)
  )
)


(assert
  ((paragra) if
   (println "a. BARELY VISIBLE: The man-made alteration is barely visible, the natural visual")
   (println "   character of the landscape has been altered.")

   (println "b. VISIBLE, BUT NOT NOTICEABLE: The man-made alteration is visible but not")
   (println "   noticeable, and the natural visual character of the landscape is essentially")
   (println "   intact.")

   (println "c. CLEARLY VISIBLE, BUT SUBORDINATE: The man-made alteration structure is")
   (println "   clearly visible, but is still subordinate to the natural character of the")
   (println "   landscape. The natural character has been altered, but is still the")
   (println "   predominate character of the landscape.")

   (println "d. VISUALLY MODIFIED: The man-made alteration or structure is obvious and the")
   (println "   natural character of landscape no longer predominates but is still present.")

   (println "e. VISUALLY DOMINATE: The man-made alteration is visually dominant and little")
   (println "   or none of the natural character remains.")
   (nl)

   (print "CHOOSE A CHARACTER FROM: a, b, c, d, e")
  )
)
```



```
; ***** TREE starts here *****

(assert
  ((go_exp) if
   (no_alter ?no_alter)
   (travel ?no_alter ((alter ?no_alter)) ?return)
   (del_coll ?return ?short ?no_alter)
   (cut)
   (system cls)
   (assert ((page 20)))
(print "********** : flag ")(println ?return)
(print "---------- : here ")(println ?short)
   (give_exp ?short)
   (nl)
   (loop ?short ?no_alter)
  )
)


(assert
  ((del_coll ?return ?short 1) if
   (erase ?return ?short)
   (cut)
  )

  ((del_coll ?return ?short ?num) if
   (!= ?num 1)
   (== ?short ?return)
  )
)


(assert
  ((erase nil nil))
```

```
  ((erase ((coll.(?.nil)).?rest) ?return) if
   (erase ?rest ?return)
  )

  ((erase (?head.?tail) (?head.?return)) if
   (erase ?tail ?return)
  )
)


(assert
  ((loop ?return ?num) if
   (print "Do you want to REVIEW result (r) or CHANGE input (c) or REDO (d) or QUIT (q)")
   (getline stream ?look)
   (review ?return ?look ?num)
  )
)


(assert
  ((review ?return q ?) if
   (quit)
  )

  ((review ? d ?num) if
   (< = ?num 5)
   (retract ((page ?)))
   (retract ((coll 1 ?)))
   (retract ((no_alter ?)))
   (del_pred apptec)
   (del_pred visb_tech)
   (further ?num)
   (close stream)
   (doit)
  )

  ((review ? d ?num) if
   (> ?num 5)
   (retract ((no_alter ?)))
   (retract ((page ?)))
   (retract ((coll 1 ?)))
   (close stream)
   (doit)
  )


  ((review ?return c ?num) if
   (< = ?num 5)
   (del_pred apptec)
```

```
 (del_pred visb_tech)
 (input_table)
 (println "If you want to change your previous input, choose the appropriate number.")
 (getline stream ?line)
 (:= ?line_no (strtonum ?line))
 (modify ?num ?line_no)
 (questions ?return ?num)
)

((review ?return c ?num) if
 (> ?num 5)
 (nl)(nl)(nl)
 (println "Here is your previous input:")
 (nl)
 (print "COLLECTIVELY APPARENTNESS  : ")
 (coll 1 ?item)
 (println ?item)
 (retract ((coll 1 ?)))
 (nl)(nl)(nl)(nl)
 (println "***********     hit RETURN to continue     **********")
 (getline stream ?)
 (more_t_five)
 (questions ?return ?num)
)


((review ?return r ?num) if
 (retract ((page ?)))
 (assert ((page 20)))
 (system cls)
 (give_exp ?return)
 (loop ?return ?num)
)

((review ?return ?a ?num) if
 (!= ?a q)
 (!= ?a d)
 (!= ?a r)
 (!= ?a c)
 (nl)(nl)(nl)(nl)
 (println "Your input is not appropriate, you must input either 'q' or 'c' or 'd' or 'r'.")
 (println "Answer the following question again !")
 (nl)
 (loop ?return ?num)
)
)


; delete the entry (apptec y) or (apptec n) in the dynamic database if it exist.
```

```
(assert
  ((del_pred ?pred) if
   (?pred ?)
   (retract ((?pred ?)))
   (cut)
  )

  ((del_pred ?)
  )
)


; (modify  ...) modifies the selected entry in dynamic db. First remove the
; entry, then pop up the appropriate question for the user.

(assert
  ((modify ?num 1) if
   (remove 1 ?num tech)
   (pre_tech ?num)
  )

  ((modify ?num 2) if
   (remove 1 ?num awe)
   (pre_awe ?num)
  )

  ((modify ?num 3) if
   (remove 1 ?num hist)
   (pre_hist ?num)
  )

  ((modify ?num 4) if
   (remove 1 ?num huma)
   (pre_huma ?num)
  )

  ((modify ?num 5) if
   (remove 1 ?num natu)
   (pre_natu ?num)
  )

  ((modify ?num 6) if
   (remove 1 ?num shel)
   (pre_shel ?num)
  )

  ((modify ?num 7) if
   (remove 1 ?num pros)
   (pre_pros ?num)
```

```
  )

  ((modify ?num 8) if
   (remove 1 ?num refl)
   (pre_refl ?num)
  )

  ((modify ?num 9) if
   (remove 1 ?num visb)
   (pre_visb ?num)
  )

  ((modify ?num 10) if
   (retract ((coll 1 ?)))
   (larger ?num)
  )

  ((modify ?num ?line) if
   (or (< ?line 1) (> ?line 10))
   (print "Your choise is not right, pick a number between 1 and 10. Try again!")
   (getline stream ?newline)
   (:= ?line_no (strtonum ?newline))
   (modify ?num ?line_no)
  )
)


(assert
  ((questions ?return ?num) if
   (nl)(nl)(nl)(nl)
   (println "Do you want to CHANGE an entry of your input (c), or")
   (println "Do you want to VIEW the changes you just made (v), or")
   (println "Do you want to RUN the program (r), or")
   (println "Do you want to QUIT (q).")
   (getline stream ?ans)
   (change_loop ?return ?ans ?num)
  )
)


(assert
  ((change_loop ?return c ?num) if
   (nl)(nl)
   (review ?return c ?num)
  )

  ((change_loop ?return v ?num) if
   (<= ?num 5)
   (input_table)
```

```
  (println "Do you want to CHANGE your input (c) or RUN the program (r) or QUIT (q) ?")
  (getline stream ?ans)
  (change_loop ?return ?ans ?num)
 )

 ((change_loop ?return v ?num) if
  (> ?num 5)
  (nl)(nl)(nl)
  (println "Your last input is:")
  (nl)
  (print "COLLECTIVELY APPARENTNESS   :  ")
  (coll 1 ?item)
  (println ?item)
  (nl)(nl)
  (println "Do you want to CHANGE your input (c) or RUN the program (r) or QUIT (q) ?")
  (getline stream ?ans)
  (change_loop ?return ?ans ?num)
 )

 ((change_loop ? r ?) if
  (not ((retract ((vis_highest ? ?)))(fail)))
  (not ((retract ((value ?)))(fail)))
  (retract ((page ?)))
  (go_exp)
 )

 ((change_loop ? q ?) if
  (quit)
 )

 ((change_loop ?return ?ans ?num) if
  (!= ?ans q)
  (!= ?ans r)
  (!= ?ans v)
  (!= ?ans c)
  (println "Your input is not right, type in only 'c' or 'v' or 'r' or 'q'.  Try again!")
  (getline stream ?newans)
  (change_loop ?return ?newans ?num)
 )
)



(assert
 ((input_table) if
  (system cls)
  (println "Here is your last input:")
  (nl)(nl)
  (print "  LANDSCAPE CHARACTERISTICS    :")
```

```
   (no_alter ?num)
   (show_obj 1 ?num)
   (print "                              ")
   (show_bar 1 ?num)
   (print "1. TECHNOLOGICAL             ")
   (show_item tech 1 ?num)
   (print "2. AWE INSPIRING             ")
   (show_item awe 1 ?num)
   (print "3. HISTORICAL                ")
   (show_item hist 1 ?num)
   (print "4. HUMAN SCALE               ")
   (show_item huma 1 ?num)
   (print "5. NATURAL MATERIALS         ")
   (show_item natu 1 ?num)
   (print "6. POTENTIAL FOR SHELTER      ")
   (show_item shel 1 ?num)
   (print "7. OPPORTUNIT FOR PROSPECT    ")
   (show_item pros 1 ?num)
   (print "8. PLACE FOR REFLECTION       ")
   (show_item refl 1 ?num)
   (print "9. APPARENTNESS              ")
   (show_item visb 1 ?num)
   (print "10. COLLECTIVE APPARENTNESS   ")
   (print "   ")
   (coll 1 ?item)
   (println ?item)
   (nl)(nl)
  )
)



(assert
  ((show_obj ? 0) if (nl)
  )

  ((show_obj ?up ?down) if
   (print "OBJECT")
   (print ?up)
   (print "   ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_obj ?new_up ?new_down)
  )
)
```

```
(assert
  ((show_bar ? 0) if (nl)
  )

  ((show_bar ?up ?down) if
   (print "-------  ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_bar ?new_up ?new_down)
  )
)



(assert
  ((show_item ? ? 0) if (nl)
  )

  ((show_item ?pred ?up ?down) if
   (?pred ?up ?item)
   (print "  ")
   (print ?item)
   (print "      ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_item ?pred ?new_up ?new_down)
  )
)



(assert
  ((further ?count) if
   (not ((retract ((vis_highest ? ?)))(fail)))
   (not ((retract ((value ?)))(fail)))
   (remove 1 ?count name)
   (remove 1 ?count desc)
   (remove 1 ?count indi)
   (remove 1 ?count tech)
   (remove 1 ?count awe)
   (remove 1 ?count hist)
   (remove 1 ?count huma)
   (remove 1 ?count natu)
   (remove 1 ?count shel)
   (remove 1 ?count pros)
   (remove 1 ?count refl)
   (remove 1 ?count visb)
```

```
  )

  ((further ?count) if
   (> ?count 5)
  )
)

(assert
  ((remove ? 0 ?))

  ((remove ?up ?down ?pred) if
   (retract ((?pred ?up ?)))
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (remove ?new_up ?new_down ?pred)
  )
)



(assert
  ((give_exp ((man.(?num.nil)).?rest)) if
   (= = ?num 8)
   (println "To have a valid conclusion, your collective apparent value should be no less")
   (println "than the highest apparent value of the alterations.")
   (nl)
   (no_alter ?no_alter)
   (show 1 ?no_alter)
   (nl)
   (print "The collective apparentness:   ")
   (coll 1 ?coll)
   (println ?coll)
   (nl)
  )

  ((give_exp ((man.(?num.nil)).?rest)) if
   (> = ?num 1)
   (< ?num 8)
   (= = (?head.?tail) ?rest)
   (deal_head ?head)
   (deal_tail ?tail)
  )
)



(assert
  ((show ? 0))
```

```
  ((show ?up ?down) if
   (name ?up ?name)
   (print "alteration name  : ")
   (print ?name)
   (visb ?up ?visb)
   (print "              apparentness :")
   (println ?visb)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show ?new_up ?new_down)
  )
)



(assert
  ((deal_head (alter.(?num.nil))) if
   (nl)
   (expla man b_c_common ?expla)
   (print_exp ?expla)
   (print ?num " man-made ")
   (alter_num ?num ?back)
   (expla man ?back ?expla1)
   (print_exp ?expla1)
   (page ?page)
   (page_hold ?page)
   (nl)
  )
)


; if the alteration number is < = 5, return b, otherwise return c for further
; matching of the explanation in static db.

(assert
  ((alter_num ?num b) if
   (< = ?num 5)
   (cut)
  )

  ((alter_num ?num c) if
   (> ?num 5)
  )
)


; take out the current head of the decision list. Matching the predicate
; of the current head with its corresponding explanation in the static db
; continueously until the decision list is run out.
```

```
(assert
  ((deal_tail nil))

  ((deal_tail ((?first.(?second.nil)).?rest)) if
   (expla ?first ?second ?expla)
   (print_exp ?expla)
   (page ?page)          ; These two sentences count the following bland
   (page_hold ?page)     ; line.
   (nl)
   (deal_tail ?rest)
  )
)


; print a paragraph

(assert
  ((print_exp nil))

  ((print_exp (?head.?tail)) if
   (page_ctrl ?head)
   (nl)
   (print_exp ?tail)
  )
)



; ***** Control page, every 20 lines of explanation + 2 lines of prompt
; ***** information is a page


(assert
  ((page_ctrl ?line) if
   (page ?page)
   (page_hold ?page)
   (put_exp ?line)
  )
)

; ***** Check the page control sign to decide to move to next page or not *****


(assert
  ((page_hold 0) if
   (retract ((page ?)))
   (assert ((page 20)))
   (nl)
   (println "     **********     please hit RETURN to continue     **********")
```

```
  (getline stream ?anykey)
  (system cls)
  (cut)
 )

 ((page_hold ?page) if
  (> ?page 0)
  (retract ((page ?)))
  (:= ?new_page (- ?page 1))
  (assert ((page ?new_page)))
 )
)



(assert
 ((put_exp nil))

 ((put_exp (?head.?tail)) if
  (print ?head)
  (print " ")
  (put_exp ?tail)
 )
)




; ***** START OF THE TRAVERSING OF THE TREE *****

; ***** NODE 2 *****

(assert
 ((travel 0 ?frame ?return) if
  (append ((man 1)) ?frame ?new_frame)
  (append ?new_frame ((resolution 1)) ?return)
  (cut)
 )

 ; ***** NODE 4 *****

 ((travel ?item ?frame ?return) if
  (> ?item 5)
  (coll 1 ?feed_back)
  (t_collect ?feed_back ?frame ?return)
  (cut)
 )
```

```
  ; *****  NODE 3  *****

  ((travel ?item ?frame ?return) if
   (> ?item 0)
   (partial tech ?feed_back)
   (t_tech ?feed_back ?frame ?return)
   (cut)
  )
)


; *****  NODE 5  *****

(assert
  ((t_tech y ?frame ?return) if
   (append ?frame ((tech y)) ?new_frame)
   (tech ?num y)
   (no_alter ?no_alter)
   (assert ((value ?num)))
   (assert ((vis_highest 1 dummy))) ; be here only for deletion later on
   (tech_high ?num ?no_alter)
   (value ?row)
   (visb ?row ?feed_back)
   (t_appa ?feed_back ?new_frame ?return)
   (cut)
  )


  ; *****  NODE 6  *****

  ((t_tech n ?frame ?return) if
   (assert ((value dummy))) ; be here only for deletion later on, no real meaning
   (append ?frame ((tech n)) ?new_frame)
   (no_alter ?no_alter)
   (all huma ?no_alter ?feed_back)
   (t_human ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 7  *****

(assert
  ((t_appa ?level ?frame ?return) if
   (>= ?level d)
   (append ?frame ((visb_tech a)) ?new_frame)
(print "new_frame: ")(println ?new_frame)
   (value ?num)
```

```
(print "value_num :  ")(println ?num)
  (awe ?num ?feed_back)
  (t_awe ?feed_back ?new_frame ?return)
  (cut)
 )


 ; *****  NODE 8  *****

 ((t_appa ?level ?frame ?return) if
  (< = ?level c)
  (append ?frame ((visb_tech b)) ?new_frame)
  (no_alter ?alter)
  (highest ?alter ?highest)
  (vis_tech ?highest ?feed_back)
  (t_apptec ?feed_back ?new_frame ?return)
 )
)


; *****  NODE 14  *****
; All the nodes following the node 14 are the same as those following node 13.
; So, only the nodes following the node 14 are present.


(assert
 ((t_human y ?frame ?return) if
  (append ?frame ((huma y)) ?new_frame)
  (find_high hist ?feed_back)
  (t_hist ?feed_back ?new_frame ?return)
  (cut)
 )


 ; *****  NODE 9  *****

 ((t_human n ?frame ?return) if
  (append ?frame ((huma n)) ?new_frame)
  (find_high natu ?feed_back)
  (t_natu ?feed_back ?new_frame ?return)
  (cut)
 )
)


; *****  NODE 10  *****

(assert
 ((t_awe y ?frame ?return) if
```

```
  (append ?frame ((awe y)) ?new_frame)
  (append ((man 2)) ?new_frame ?middle)
  (append ?middle ((resolution 2)) ?return)
  (cut)
 )


 ; *****  NODE 11  *****

 ((t_awe n ?frame ?return) if
  (append ?frame ((awe n)) ?new_frame)
  (find_high hist ?feed_back)
  (t_hist_4 ?feed_back ?new_frame ?return)
  (cut)
 )
)


; *****  NODE 12  *****

(assert
 ((t_apptec y ?frame ?return) if
  (append ?frame ((apptec y)) ?new_frame)
  (find_high hist ?feed_back)
  (t_hist_3 ?feed_back ?new_frame ?return)
  (cut)
 )

 ; *****  NODE 13  *****

 ((t_apptec n ?frame ?return) if
  (append ?frame ((apptec n)) ?new_frame)
  (find_high hist ?feed_back)
  (t_hist ?feed_back ?new_frame ?return)
  (cut)
 )
)



; *****  NODE 21  *****

(assert
 ((t_hist y ?frame ?return) if
  (append ?frame ((hist y)) ?new_frame)
  (vis_highest ?num ?highest)
  (natu ?num ?feed_back)
  (t_natu_1 ?feed_back ?new_frame ?return)
  (cut)
```

```
  )

  ; *****  NODE 22  *****

  ((t_hist n ?frame ?return) if
   (append ?frame ((hist n)) ?new_frame)
   (vis_highest ?num ?highest)
   (natu ?num ?feed_back)
   (t_natu_2 ?feed_back ?new_frame ?return)
   (cut)
  )
)

; *****  NODE 15  *****

(assert
  ((t_natu y ?frame ?return) if
   (append ?frame ((natu y)) ?new_frame)
   (vis_highest ?num ?highest)
   (shel ?num ?feed_back)
   (t_shel ?feed_back ?new_frame ?return)
   (cut)
  )

  ; *****  NODE 16  *****

  ((t_natu n ?frame ?return) if
   (append ?frame ((natu n)) ?new_frame)
   (vis_highest ?num ?highest)
   (shel ?num ?feed_back)
   (t_shel_1 ?feed_back ?new_frame ?return)
   (cut)
  )
)

; *****  NODE 17  *****

(assert
  ((t_hist_4 y ?frame ?return) if
   (append ?frame ((hist y)) ?new_frame)
   (coll 1 ?coll)
   (clear_1 ?coll ?new_frame ?return)
   (cut)
  )
```

```
  ; *****  NODE 18  *****

  ((t_hist_4 n ?frame ?return) if
   (append ?frame ((hist n)) ?new_frame)
   (coll 1 ?coll)
   (clear_2 ?coll ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 19  *****

(assert
  ((t_hist_3 y ?frame ?return) if
   (append ?frame ((hist y)) ?new_frame)
   (coll 1 ?coll)
   (clear_3 ?coll ?new_frame ?return)
   (cut)
  )


  ; *****  NODE 20  *****

  ((t_hist_3 n ?frame ?return) if
   (append ?frame ((hist n)) ?new_frame)
   (coll 1 ?coll)
   (clear_4 ?coll ?new_frame ?return)
   (cut)
  )
)

; *****  NODE 30  *****

(assert
  ((t_natu_1 y ?frame ?return) if
   (append ?frame ((natu y)) ?new_frame)
   (vis_highest ?num ?highest)
   (appart3 ?highest ?new_frame ?return)
   (cut)
  )


  ; *****  NODE 31  *****

  ((t_natu_1 n ?frame ?return) if
   (append ?frame ((natu n)) ?new_frame)
   (vis_highest ?num ?highest)
   (appart4 ?highest ?new_frame ?return)
```

```
  (cut)
 )
)

; ***** NODE 32 *****

(assert
 ((t_natu_2 y ?frame ?return) if
  (append ?frame ((natu y)) ?new_frame)
  (vis_highest ?num ?highest)
  (appart5 ?highest ?new_frame ?return)
  (cut)
 )

 ; ***** NODE 33 *****

 ((t_natu_2 n ?frame ?return) if
  (append ?frame ((natu n)) ?new_frame)
  (vis_highest ?num ?highest)
  (appart6 ?highest ?new_frame ?return)
  (cut)
 )
)

; ***** NODE 25 *****

(assert
 ((t_shel y ?frame ?return) if
  (append ?frame ((shel y)) ?new_frame)
  (vis_highest ?num ?highest)
  (pros ?num ?feed_back)
  (t_pros_1 ?feed_back ?new_frame ?return)
  (cut)
 )

 ; ***** NODE 26 *****

 ((t_shel n ?frame ?return) if
  (append ?frame ((shel n)) ?new_frame)
  (vis_highest ?num ?highest)
  (hist ?num ?feed_back)
  (t_hist_1 ?feed_back ?new_frame ?return)
  (cut)
 )
)
```

```
; *****  NODE 27  *****

(assert
  ((t_shel_1 y ?frame ?return) if
   (append ?frame ((shel y)) ?new_frame)
   (vis_highest ?num ?highest)
   (pros ?num ?feed_back)
   (t_pros_2 ?feed_back ?new_frame ?return)
   (cut)
  )

  ; *****  NODE 28  *****

  ((t_shel_1 n ?frame ?return) if
   (append ?frame ((shel n)) ?new_frame)
   (vis_highest ?num ?highest)
   (pros ?num ?feed_back)
   (t_pros_3 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 81  *****

(assert
  ((t_pros_1 y ?frame ?return) if
   (append ?frame ((pros y)) ?new_frame)
   (append ((man 2)) ?new_frame ?middle)
   (append ?middle ((resolution 2)) ?return)
   (cut)
  )


  ; *****  NODE 82  *****

  ((t_pros_1 n ?frame ?return) if
   (append ?frame ((pros n)) ?new_frame)
   (vis_highest ?num ?highest)
   (refl ?num ?feed_back)
   (t_refl_1 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 83  *****
```

```
(assert
  ((t_hist_1 y ?frame ?return) if
   (append ?frame ((hist y)) ?new_frame)
   (append ((man 2)) ?new_frame ?middle)
   (append ?middle ((resolution 2)) ?return)
   (cut)
  )


  ; *****  NODE 84  *****

  ((t_hist_1 n ?frame ?return) if
   (append ?frame ((hist n)) ?new_frame)
   (vis_highest ?num ?highest)
   (refl ?num ?feed_back)
   (t_refl_2 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 85  *****

(assert
  ((t_pros_2 y ?frame ?return) if
   (append ?frame ((pros y)) ?new_frame)
   (append ((man 2)) ?new_frame ?middle)
   (append ?middle ((resolution 2)) ?return)
   (cut)
  )


  ; *****  NODE 86  *****

  ((t_pros_2 n ?frame ?return) if
   (append ?frame ((pros n)) ?new_frame)
   (vis_highest ?num ?highest)
   (refl ?num ?feed_back)
   (t_refl_3 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 87  *****

(assert
  ((t_pros_3 y ?frame ?return) if
   (append ?frame ((pros y)) ?new_frame)
```

```
   (append ((man 3)) ?new_frame ?middle)
   (append ?middle ((resolution 3)) ?return)
   (cut)
  )

  ; *****  NODE 88  *****

  ((t_pros_3 n ?frame ?return) if
   (append ?frame ((pros n)) ?new_frame)
   (vis_highest ?num ?highest)
   (refl ?num ?feed_back)
   (t_refl_4 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 249  *****

(assert
  ((t_refl_1 y ?frame ?return) if
   (append ?frame ((refl y)) ?new_frame)
   (append ((man 2)) ?new_frame ?middle)
   (append ?middle ((resolution 2)) ?return)
   (cut)
  )


  ; *****  NODE 250  *****

  ((t_refl_1 n ?frame ?return) if
   (append ?frame ((refl n)) ?new_frame)
   (vis_highest ?num ?highest)
   (hist ?num ?feed_back)
   (t_hist_2 ?feed_back ?new_frame ?return)
   (cut)
  )
)


; *****  NODE 49  *****

(assert
  ((appart3 a ?frame ?return) if
   (coll 1 ?collect)
   (c_a_app ?collect ?frame ?return)
   (cut)
  )
```

```
  ; *****  NODE 50  *****

  ((appart3 b ?frame ?return) if
   (coll 1 ?collect)
   (c_b_app ?collect ?frame ?return)
   (cut)
  )


  ; *****  NODE 51  *****

  ((appart3 c ?frame ?return) if
   (coll 1 ?collect)
   (c_c_app ?collect ?frame ?return)
   (cut)
  )


  ; *****  NODE 51.a  *****

  ((appart3 d ?frame ?return) if
   (coll 1 ?collect)
   (c_c_app ?collect ?frame ?return)
   (cut)
  )


  ; *****  NODE 52  *****

  ((appart3 e ?frame ?return) if
   (coll 1 ?collect)
   (c_d_app ?collect ?frame ?return)
   (cut)
  )
)


; *****  NODE 53  *****

(assert
  ((appart4 a ?frame ?return) if
   (coll 1 ?collect)
   (c_a_app ?collect ?frame ?return)
   (cut)
  )


  ; *****  NODE 54  *****
```

```
  ((appart4 b ?frame ?return) if
   (coll 1 ?collect)
   (c_b_app ?collect ?frame ?return)
   (cut)
  )

  ; *****  NODE 55  *****

  ((appart4 c ?frame ?return) if
   (coll 1 ?collect)
   (c_e_app ?collect ?frame ?return)
   (cut)
  )

  ; *****  NODE 55.a  *****

  ((appart4 d ?frame ?return) if
   (coll 1 ?collect)
   (c_e_app ?collect ?frame ?return)
   (cut)
  )

  ; *****  NODE 56  *****

  ((appart4 e ?frame ?return) if
   (coll 1 ?collect)
   (c_f_app ?collect ?frame ?return)
   (cut)
  )
)


; *****  NODE 57  *****

(assert
  ((appart5 a ?frame ?return) if
   (coll 1 ?collect)
   (c_a_app ?collect ?frame ?return)
   (cut)
  )

  ; *****  NODE 58  *****

  ((appart5 b ?frame ?return) if
```

```
 (coll 1 ?collect)
 (c_b_app ?collect ?frame ?return)
 (cut)
)

; ***** NODE 59 *****

((appart5 c ?frame ?return) if
 (coll 1 ?collect)
 (c_g_app ?collect ?frame ?return)
 (cut)
)

; ***** NODE 59.a *****

((appart5 d ?frame ?return) if
 (coll 1 ?collect)
 (c_g_app ?collect ?frame ?return)
 (cut)
)

; ***** NODE 60 *****

((appart5 e ?frame ?return) if
 (coll 1 ?collect)
 (c_h_app ?collect ?frame ?return)
 (cut)
)
)

; ***** NODE 61 *****

(assert
 ((appart6 a ?frame ?return) if
  (coll 1 ?collect)
  (c_a_app ?collect ?frame ?return)
  (cut)
 )

 ; ***** NODE 62 *****

 ((appart6 b ?frame ?return) if
  (coll 1 ?collect)
  (c_i_app ?collect ?frame ?return)
```

```
  (cut)
 )


 ; *****  NODE 63  *****

 ((appart6 c ?frame ?return) if
  (coll 1 ?collect)
  (c_j_app ?collect ?frame ?return)
  (cut)
 )


 ; *****  NODE 63.a  *****

 ((appart6 d ?frame ?return) if
  (coll 1 ?collect)
  (c_j_app ?collect ?frame ?return)
  (cut)
 )


 ; *****  NODE 64  *****

 ((appart6 e ?frame ?return) if
  (coll 1 ?collect)
  (c_k_app ?collect ?frame ?return)
  (cut)
 )
)



; *****  NODE 257  *****

(assert
 ((t_hist_2 y ?frame ?return) if
  (append ?frame ((hist y)) ?new_frame)
  (append ((man 2)) ?new_frame ?middle)
  (append ?middle ((resolution 2)) ?return)
  (cut)
 )


 ; *****  NODE 258  *****

 ((t_hist_2 n ?frame ?return) if
  (append ?frame ((hist n)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
```

```
   (append ?middle ((resolution 3)) ?return)
   (cut)
  )
)


; *****  NODE 251  *****

(assert
  ((t_refl_2 y ?frame ?return) if
   (append ?frame ((refl y)) ?new_frame)
   (append ((man 3)) ?new_frame ?middle)
   (append ?middle ((resolution 3)) ?return)
   (cut)
  )


  ; *****  NODE 252  *****

  ((t_refl_2 n ?frame ?return) if
   (append ?frame ((refl n)) ?new_frame)
   (append ((man 1)) ?new_frame ?middle)
   (append ?middle ((resolution 1)) ?return)
   (cut)
  )
)


; *****  NODE 253  *****

(assert
  ((t_refl_3 y ?frame ?return) if
   (append ?frame ((refl y)) ?new_frame)
   (append ((man 2)) ?new_frame ?middle)
   (append ?middle ((resolution 2)) ?return)
   (cut)
  )


  ; *****  NODE 254  *****

  ((t_refl_3 n ?frame ?return) if
   (append ?frame ((refl n)) ?new_frame)
   (append ((man 3)) ?new_frame ?middle)
   (append ?middle ((resolution 3)) ?return)
   (cut)
  )
)
```

```
; ***** NODE 255 *****

(assert
 ((t_refl_4 y ?frame ?return) if
  (append ?frame ((refl y)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )


 ; ***** NODE 256 *****

 ((t_refl_4 n ?frame ?return) if
  (append ?frame ((refl n)) ?new_frame)
  (append ((man 1)) ?new_frame ?middle)
  (append ?middle ((resolution 1)) ?return)
  (cut)
 )
)


; ***** CLEARS after node 17, 18, 19, 20 *****

  ; ***** NODE 38.a *****

(assert
 ((clear_1 a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

  ; ***** NODE 38.b *****

 ((clear_1 b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

  ; ***** NODE 38.c *****

 ((clear_1 c ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )
```

```
  ; *****  NODE 39  *****

 ((clear_1 d ?frame ?return) if
  (append ?frame ((visb_coll r6_a_1)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

  ; *****  NODE 40  *****

 ((clear_1 e ?frame ?return) if
  (append ?frame ((visb_coll r6_a_2)) ?new_frame)
  (append ((man 7)) ?new_frame ?middle)
  (append ?middle ((resolution 7)) ?return)
  (cut)
 )
)

  ; *****  NODE 41.a  *****

(assert
 ((clear_2 a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

  ; *****  NODE 41.b  *****

 ((clear_2 b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

  ; *****  NODE 41.c  *****

 ((clear_2 c ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

  ; *****  NODE 42.a  *****

 ((clear_2 d ?frame ?return) if
  (append ?frame ((visb_coll r6_b)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
```

```
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

  ; *****  NODE 42.b  *****

 ((clear_2 e ?frame ?return) if
  (append ?frame ((visb_coll r6_b)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )
)

  ; *****  NODE 43.a  *****

(assert
 ((clear_3 a ?frame ?return) if
  (append ?frame ((visb_coll r6_g_1)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

  ; *****  NODE 43.b  *****

 ((clear_3 b ?frame ?return) if
  (append ?frame ((visb_coll r6_g_1)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

  ; *****  NODE 44  *****

 ((clear_3 c ?frame ?return) if
  (append ?frame ((visb_coll r6_g_2)) ?new_frame)
  (append ((man 7)) ?new_frame ?middle)
  (append ?middle ((resolution 7)) ?return)
  (cut)
 )

  ; *****  NODE 45.a  *****

 ((clear_3 d ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )
```

```
  ; *****  NODE 45.b  *****

 ((clear_3 e ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )
)

  ; *****  NODE 46  *****

(assert
 ((clear_4 a ?frame ?return) if
  (append ?frame ((visb_coll r6_h_1)) ?new_frame)
  (append ((man 7)) ?new_frame ?middle)
  (append ?middle ((resolution 7)) ?return)
  (cut)
 )

  ; *****  NODE 47.a  *****

 ((clear_4 b ?frame ?return) if
  (append ?frame ((visb_coll r6_h_2)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

  ; *****  NODE 47.b  *****

 ((clear_4 c ?frame ?return) if
  (append ?frame ((visb_coll r6_h_2)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

  ; *****  NODE 48.a  *****

 ((clear_4 d ?frame ?return) if
  (append ?frame ((visb_coll r6_h_3)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

  ; *****  NODE 48.b  *****

 ((clear_4 e ?frame ?return) if
```

```
  (append ?frame ((visb_coll r6_h_3)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )
)


; ***** all C_A_APPs *****

 ; ***** NODE 89 *****

(assert
 ((c_a_app a ?frame ?return) if
  (append ?frame ((visb_coll r6_c_1)) ?new_frame)
  (append ((man 1)) ?new_frame ?middle)
  (append ?middle ((resolution 1)) ?return)
  (cut)
 )

 ; ***** NODE 90 *****

 ((c_a_app b ?frame ?return) if
  (append ?frame ((visb_coll r6_c_2)) ?new_frame)
  (append ((man 7)) ?new_frame ?middle)
  (append ?middle ((resolution 7)) ?return)
  (cut)
 )

 ; ***** NODE 91 *****

 ((c_a_app c ?frame ?return) if
  (append ?frame ((visb_coll r6_c_3)) ?new_frame)
  (append ((man 4)) ?new_frame ?middle)
  (append ?middle ((resolution 4)) ?return)
  (cut)
 )

 ; ***** NODE 92 *****

 ((c_a_app d ?frame ?return) if
  (append ?frame ((visb_coll r6_c_4)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

 ; ***** NODE 93 *****
```

```
  ((c_a_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_c_5)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )
)


; ***** ALL  C_B_APP's *****

  ; *****   NODE 94   *****

(assert
  ((c_b_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****   NODE 95   *****

  ((c_b_app b ?frame ?return) if
   (append ?frame ((visb_coll r6_d_1)) ?new_frame)
   (append ((man 6)) ?new_frame ?middle)
   (append ?middle ((resolution 6)) ?return)
   (cut)
  )

  ; *****   NODE 96   *****

  ((c_b_app c ?frame ?return) if
   (append ?frame ((visb_coll r6_d_2)) ?new_frame)
   (append ((man 4)) ?new_frame ?middle)
   (append ?middle ((resolution 4)) ?return)
   (cut)
  )

  ; *****   NODE 97   *****

  ((c_b_app d ?frame ?return) if
   (append ?frame ((visb_coll r6_d_3)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )

  ; *****   NODE 98   *****
```

```
 ((c_b_app e ?frame ?return) if
  (append ?frame ((visb_coll r6_d_4)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )
)


; ***** ALL  C_C_APP's *****

 ; *****  NODE 99  *****

(assert
 ((c_c_app a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 100  *****

 ((c_c_app b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 101  *****

 ((c_c_app c ?frame ?return) if
  (append ?frame ((visb_coll r6_e_1)) ?new_frame)
  (append ((man 2)) ?new_frame ?middle)
  (append ?middle ((resolution 2)) ?return)
  (cut)
 )

 ; *****  NODE 102  *****

 ((c_c_app d ?frame ?return) if
  (append ?frame ((visb_coll r6_e_2)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

 ; *****  NODE 103  *****

 ((c_c_app e ?frame ?return) if
```

```
  (append ?frame ((visb_coll r6_e_3)) ?new_frame)
  (append ((man 4)) ?new_frame ?middle)
  (append ?middle ((resolution 4)) ?return)
  (cut)
 )
)


; ***** ALL  C_D_APP's *****

  ; *****  NODE 104  *****

(assert
 ((c_d_app a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 105  *****

 ((c_d_app b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 106  *****

 ((c_d_app c ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 107  *****

 ((c_d_app d ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 108  *****

 ((c_d_app e ?frame ?return) if
  (append ?frame ((visb_coll r6_f_1)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
```

```
  (cut)
 )
)


; ***** ALL C_E_APP's *****

 ; *****  NODE 119  *****

(assert
 ((c_e_app a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 120  *****

 ((c_e_app b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 121  *****

 ((c_e_app c ?frame ?return) if
  (append ?frame ((visb_coll r6_e_1)) ?new_frame)
  (append ((man 2)) ?new_frame ?middle)
  (append ?middle ((resolution 2)) ?return)
  (cut)
 )

 ; *****  NODE 122  *****

 ((c_e_app d ?frame ?return) if
  (append ?frame ((visb_coll r6_e_2)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

 ; *****  NODE 123  *****

 ((c_e_app e ?frame ?return) if
  (append ?frame ((visb_coll r6_e_3)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
```

```
  )
)


; ***** ALL  C_F_APP's *****

  ; *****  NODE 124  *****

(assert
  ((c_f_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 125  *****

  ((c_f_app b ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 126  *****

  ((c_f_app c ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )


  ; *****  NODE 127  *****

  ((c_f_app d ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )


  ; *****  NODE 128  *****

  ((c_f_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_f_1)) ?new_frame)
   (append ((man 9)) ?new_frame ?middle)
   (append ?middle ((resolution 9)) ?return)
   (cut)
  )
```

```
)

; ***** ALL C_G_APP's *****

 ; *****  NODE 139  *****

(assert
 ((c_g_app a ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 140  *****

 ((c_g_app b ?frame ?return) if
  (append ((man 8)) ?frame ?middle)
  (append ?middle ((resolution 8)) ?return)
  (cut)
 )

 ; *****  NODE 141  *****

 ((c_g_app c ?frame ?return) if
  (append ?frame ((visb_coll r6_e_1)) ?new_frame)
  (append ((man 3)) ?new_frame ?middle)
  (append ?middle ((resolution 3)) ?return)
  (cut)
 )

 ; *****  NODE 142  *****

 ((c_g_app d ?frame ?return) if
  (append ?frame ((visb_coll r6_e_2)) ?new_frame)
  (append ((man98)) ?new_frame ?middle)
  (append ?middle ((resolution 9)) ?return)
  (cut)
 )

 ; *****  NODE 143  *****

 ((c_g_app e ?frame ?return) if
  (append ?frame ((visb_coll r6_e_3)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )
)
```

```
; ***** ALL C_H_APP's *****

  ; *****  NODE 144  *****

(assert
  ((c_h_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 145  *****

  ((c_h_app b ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 146  *****

  ((c_h_app c ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 147  *****

  ((c_h_app d ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 148  *****

  ((c_h_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_f_1)) ?new_frame)
   (append ((man 4)) ?new_frame ?middle)
   (append ?middle ((resolution 4)) ?return)
   (cut)
  )
)


; ***** ALL C_I_APP's *****
```

```
  ; *****  NODE 154  *****

 (assert
  ((c_i_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 155  *****

  ((c_i_app b ?frame ?return) if
   (append ?frame ((visb_coll r6_d_1)) ?new_frame)
   (append ((man 1)) ?new_frame ?middle)
   (append ?middle ((resolution 1)) ?return)
   (cut)
  )

  ; *****  NODE 156  *****

  ((c_i_app c ?frame ?return) if
   (append ?frame ((visb_coll r6_d_2)) ?new_frame)
   (append ((man 4)) ?new_frame ?middle)
   (append ?middle ((resolution 4)) ?return)
   (cut)
  )

  ; *****  NODE 157  *****

  ((c_i_app d ?frame ?return) if
   (append ?frame ((visb_coll r6_d_3)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )

  ; *****  NODE 158  *****

  ((c_i_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_d_4)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )
 )


; ***** ALL  C_J_APP's *****
```

```
  ; *****  NODE 159  *****

 (assert
  ((c_j_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 160  *****

  ((c_j_app b ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 161  *****

  ((c_j_app c ?frame ?return) if
   (append ?frame ((visb_coll r6_e_1)) ?new_frame)
   (append ((man 4)) ?new_frame ?middle)
   (append ?middle ((resolution 4)) ?return)
   (cut)
  )

  ; *****  NODE 162  *****

  ((c_j_app d ?frame ?return) if
   (append ?frame ((visb_coll r6_e_2)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )

  ; *****  NODE 163  *****

  ((c_j_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_e_3)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )
 )



; ***** ALL  C_K_APP's *****
```

```
  ; *****  NODE 164  *****

 (assert
  ((c_j_app a ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 165  *****

  ((c_j_app b ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 166  *****

  ((c_j_app c ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 167  *****

  ((c_j_app d ?frame ?return) if
   (append ((man 8)) ?frame ?middle)
   (append ?middle ((resolution 8)) ?return)
   (cut)
  )

  ; *****  NODE 168  *****

  ((c_j_app e ?frame ?return) if
   (append ?frame ((visb_coll r6_f_1)) ?new_frame)
   (append ((man 5)) ?new_frame ?middle)
   (append ?middle ((resolution 5)) ?return)
   (cut)
  )
 )


; *****  ALL T_COLLECTs after NODE 4  *****

  ; *****  NODE 259  *****
```

```
(assert
 ((t_collect a ?frame ?return) if
  (append ?frame ((coll a)) ?new_frame)
  (append ((man 1)) ?new_frame ?middle)
  (append ?middle ((resolution 1)) ?return)
  (cut)
 )

 ; *****  NODE 260  *****

 ((t_collect b ?frame ?return) if
  (append ?frame ((coll b)) ?new_frame)
  (append ((man 7)) ?new_frame ?middle)
  (append ?middle ((resolution 7)) ?return)
  (cut)
 )

 ; *****  NODE 261  *****

 ((t_collect c ?frame ?return) if
  (append ?frame ((coll c)) ?new_frame)
  (append ((man 4)) ?new_frame ?middle)
  (append ?middle ((resolution 4)) ?return)
  (cut)
 )

 ; *****  NODE 262  *****

 ((t_collect d ?frame ?return) if
  (append ?frame ((coll d)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )

 ; *****  NODE 263  *****

 ((t_collect e ?frame ?return) if
  (append ?frame ((coll e)) ?new_frame)
  (append ((man 5)) ?new_frame ?middle)
  (append ?middle ((resolution 5)) ?return)
  (cut)
 )
)


; *****  ALL MISLANEIES PREDICATES  *****
```

```
(assert
  ((append nil ?z ?z))
  ((append (?x.?y) ?z (?x.?w)) if
   (append ?y ?z ?w)
  )
)


(assert
  ((partial ?pred y) if
   (?pred ? y)
   (cut)
  )

  ((partial ?pred n))
)



(assert
  ((all ?pred 0 y) if
   (cut)
  )

  ((all ?pred ?total ?feed_back) if
   (?pred ?total y)
   (:= ?down_total (- ?total 1))
   (all ?pred ?down_total ?feed_back)
   (cut)
  )

  ((all ?pred ? n)
  )
)


(assert
  ((find_high ?pred ?feed_back) if
   (no_alter ?total)
   (highest ?total ?highest)
(print "highest:   ")(println ?highest)
   (visb ?num ?highest)
(print "num:   ")(println ?num)
   (?pred ?num ?feed_back)
   (assert ((vis_highest ?num ?highest)))
  )
)
```

```
(assert
  ((highest ?total ?highest) if
   (get_set ?total ?set)
(print "nset:   ")(println ?set)
   (qsort ?set ?new_set)
(print "qsort_nwe_set:   ")(println ?new_set)
   (reverse ?new_set ?rev_set)
   (= = (?highest.?tail) ?rev_set)
  )
)


(assert
  ((get_set 0 nil))

  ((get_set ?total (?visb.?add_set)) if
   (visb ?total ?visb)
   (:= ?down_total (- ?total 1))
   (get_set ?down_total ?add_set)
  )
)


(assert
  ((reverse (?c.?l1) ?l2) if
   (reverse ?l1 ?out)
   (append ?out (?c) ?l2)
  )

  ((reverse nil nil))
)


(assert
  ((tech_high ? 0))

  ((tech_high ?start ?count) if
   (tech ?count ?tech)
   (compare ?start ?count ?tech)
   (:= ?new_count (- ?count 1))
   (value ?high_count)
   (tech_high ?high_count ?new_count)
  )
)


(assert
  ((compare ? ? n))
```

```
  ((compare ?first ?second y) if
   (visb ?first ?visb_1)
   (visb ?second ?visb_2)
   (> = ?visb_1 ?visb_2)
  )

  ((compare ?first ?second y) if
   (visb ?first ?visb_1)
   (visb ?second ?visb_2)
   (< ?visb_1 ?visb_2)
   (retract ((value ?value)))
   (assert ((value ?second)))
  )
)


(assert
  ((qsort nil nil))

  ((qsort (?x.?y) ?ans) if
   (partition ?x ?y ?less ?more)
   (qsort ?less ?less1)
   (qsort ?more ?more1)
   (append ?less1 (?x.?more1) ?ans)
  )
)


(assert
  ((partition ?x nil nil nil))

  ((partition ?x (?y.?z) (?y.?less) ?more) if
   (< ?y ?x)
   (partition ?x ?z ?less ?more)
  )

  ((partition ?x (?y.?z) ?less (?y.?more)) if
   (> = ?y ?x)
   (partition ?x ?z ?less ?more)
  )
)



(assert
  ((vis_tech ?highest y) if
   (visb ?num ?highest)
   (tech ?num y)
   (cut)
```

```
 )

 ((vis_tech ? n)
 )
)


; ***** VARIOUS FACTS *****

(assert

 ((expla resolution 1 ((FINAL CONCLUSION:) (No effect on visual quality))))
 ((expla resolution 2 ((FINAL CONCLUSION:) (High visual quality))))
 ((expla resolution 3 ((FINAL CONCLUSION:) (Morderately high visual quality))))
 ((expla resolution 4 ((FINAL CONCLUSION:) (Morderately low visual quality))))
 ((expla resolution 5 ((FINAL CONCLUSION:) (Low visual quality))))
 ((expla resolution 6 ((FINAL CONCLUSION:) (SlightlY positive effect on visual quality))))
 ((expla resolution 7 ((FINAL CONCLUSION:) (Slightly negative effect on visual ality))))
 ((expla resolution 8 ((FINAL CONCLUSION:) (Not consistent, re-evaluate individual and
collective apparentness rating))))
 ((expla resolution 9 ((FINAL CONCLUSION:) (Moderate visual quality))))

 ((expla man a ((Visually evident man-made alterations of the landscape or structures in the)
  (natural landscape tend to have a negative influence on visual quality of this)
  (type of the landscape unless the alteration or structure is awe "inspiring,")
  (related to historic use of the landscape or are pleasant places for "prople.")
  (The extent of the influence on visual quality is dependent upon how many)
  (man-made features are "present." This landscape was found to contain no)
  (visually evident man-made alterations or "structures." Therefore, man-made)
  (features has no effect on visual "quality.")))
 )

 ((expla man b_c_common ((Visually evident man-made alterations of the landscape or
structures in the)
  (natural landscape tend to have a negative influence on visual quality of)
  (this type of the landscape unless the alteration or structure is awe)
  ("inspiring," related to historic use of the landscape or are pleasant places)
  (for "prople." The extent of the influence on visual quality is dependent upon)
  (how many man-made features are "present." This landscpae was found to contain)))
 )

 ((expla man b ((alterations or "structures." The extent of negative or positive)
  (influence on visual quality of the landscape depends on the individual and)
  (collective characteristics of the man-made alterations or "structures.")))
 )

 ((expla man c ((alterations or "structures." The collective effect on visual quality of)
  (this number of alterations will out weigh the effect of any individual)
```

```
 ("alteration.")))
)

((expla tech y ((Industrial, quasi-industrial or technological alterations and/or structures in)
 (the natural landscape tend to have a negative influence on the visual quality)
 (of the landscpae unless they are a source of visual interest due to their size)
 ("(awe inspiring)" or historic "significance." Industrial, quasi-industrial or)
 (technological alterations and/or structures tend to have a negative influence)
 (on visual quality because they seem out of place in the natural landscape and)
 (contrast with the natural character of the "landscape." This landscape was)
 (found contain industrial, quasi-industrial or technological alterations)
 (and/or "structures.")))
)

((expla tech n ((Industrial, quasi-industrial or technological alterations and/or structures in)
 (the natural landscape tend to have a negative influence on the visual quality)
 (of the landscpae unless they are a source of visual interest due to their size)
 ("(awe inspiring)" or historic "significance." Industrial, quasi-industrial or)
 (technological alterations and/or structures tend to have a negative influence)
 (on visual quality because they seem out of place in the natural landscape and)
 (contrast with the natural character of the "landscape." This landscape was not)
 (found contain industrial, quasi-industrial or technological alterations)
 (and/or "structures.")))
)

((expla visb_tech a ((Industrial, quasi-industrial or technological alterations and/or structures
in)
 (the landscape must be visually apparent in order to inspire awe in "people.")
 (One or more of the industrial, quasi-industrial or technological alterations)
 (and/or structures in this landscape were judged to be visually apparent)
 ("(rated as modified or dominant for apparentness)." Therefore, this landscape)
 (has the potential to be awe "inspiring.")))
)

((expla visb_tech b ((Industrial, quasi-industrial or technological alterations and/or structures
in)
 (the landscape must be visually apparent in order to inspire awe in "people.")
 (None of the industrial, quasi-industrial or technological alterations)
 (and/or structures in this landscape were judged to be visually apparent)
 ("(rated as modified or dominant for apparentness)." Therefore, this landscape)
 (does not have the potential to be awe "inspiring.")))
)

((expla awe y ((Man-made alterations or built structures that inspires that awe in people are)
 (those that are immense in size such as large pit mines or those that require)
 (major engineering feats such as large "dams." These types of alterations are)
 (fascinating to people and increase the visual quality of the "landscape." One)
 (or more of the man-made alterations or built structures in this landscape were)
 (judged to be awe "inspiring.")))
```

```
 )

 ((expla awe n ((Man-made alterations or built structures that inspires that awe in people are)
  (those that are immense in size such as large pit mines or those that require)
  (major engineering feats such as large "dams." These types of alterations are)
  (fascinating to people and increase the visual quality of the "landscape.")
  (None of the man-made alterations or built structures in this landscape were)
  (judged to be awe "inspiring.")))
 )

 ((expla hist y ((Alterations or built structures that "'appear'" to be old or historic enhance)
  (the visual quality of the "landscape." The symbolize past cultures and times)
  (which tend to fascinate people, thus contributing to people's appreciation of)
  (the "landscape." If the landscape contains more than one man-made alteration)
  (or built structure then the alteration or structure which is most visually)
  (apparent dominates and is responsible for the effect of man-made alterations)
  (or built structures on the visual quality of the "landscape." The most visually)
  (apparent man-made alteration or built structure in this landscape appear to be)
  ("historic." Therefore, this alteration or structure has positive influence on)
  (visual "quality.")))
 )

 ((expla hist n ((Alterations or built structures that "'appear'" to be old or historic enhance)
  (the visual quality of the "landscape." The symbolize past cultures and times)
  (which tend to fascinate people, thus contributing to people's appreciation of)
  (the "landscape." If the landscape contains more than one man-made alteration)
  (or built structure then the alteration or structure which is most visually)
  (apparent dominates and is responsible for the effect of man-made alterations)
  (or built structures on the visual quality of the "landscape." The most visually)
  (apparent man-made alteration or built structure in this landscape does not)
  (appear to be "historic." Therefore, this alteration or structure has positive)
  (influence on visual "quality.")))
 )

 ((expla  apptec y ((The most visually apparent man-made "alteration(s)" or "structure(s)" in
the)
  (landscape will have the greatest effect on the visual quality of the "landscape.")
  (The most visually apparent "alteration(s)" or "structure(s)" in the landscape)
  (are related to industrial, quasi-industrial or technological uses of the)
  ("landscape.")))
 )

 ((expla  apptec n ((The most visually apparent man-made "alteration(s)" or "structure(s)" in
the)
  (landscape will have the greatest effect on the visual quality of the "landscape.")
  (The most visually apparent "alteration(s)" or "structure(s)" in the landscape)
  (are not related to industrial, quasi-industrial or technological uses of the)
  ("landscape.")))
 )
```

```
((expla coll a ((The extent to which man-made structures and alterations collectively have a)
 (negative impact on the visual quality of the natural landscape depends on how)
 ("'visually apparent'" they are in the landscape or the extent to which the)
 (alterations and/or structures visibly alter the natural characterof the)
 ("landscape." Note, this determination is based on how an average lay person)
 (would perceive the "'natural character'" of the landscape "(i.e." a)
 (professional forester might be able to perceive a variety of non-natural)
 (changes in the landscape such as prescribed burns and stand thinning which)
 (would not be evident of a lay "person." The man-made alterations and/or)
 (structures are barely visible and the natural visual character of the)
 (landscape has not been "altered.")))
)

((expla coll b ((The extent to which man-made structures and alterations collectively have a)
 (negative impact on the visual quality of the natural landscape depends on how)
 ("'visually apparent'" they are in the landscape or the extent to which the)
 (alterations and/or structures visibly alter the natural characterof the)
 ("landscape." Note, this determination is based on how an average lay person)
 (would perceive the "'natural character'" of the landscape "(i.e." a)
 (forester might be able to perceive a variety of non-natural changes in the)
 (landscape such as prescribed burns and stand thinning which would not be)
 (evident of a lay "person." The man-made alterations and/or structures are)
 (visible, but the natural character of the landscape is essentially "intact.")))
)

((expla coll c ((The extent to which man-made structures and alterations collectively have a)
 (negative impact on the visual quality of the natural landscape depends on how)
 ("'visually apparent'" they are in the landscape or the extent to which the)
 (alterations and/or structures visibly alter the natural characterof the)
 ("landscape." Note, this determination is based on how an average lay person)
 (would perceive the "'natural character'" of the landscape "(i.e." a)
 (forester might be able to perceive a variety of non-natural changes in the)
 (landscape such as prescribed burns and stand thinning which would not be)
 (evident of a lay "person." The man-made alterations and/or structures are)
 (clearly visible, but they are still subordinate to the natural character of the)
 ("landscape." The natural character has been altered, but the natural character)
 (still "predominates." The landscape has a moderately low visual quality due to)
 (man-made alterations and "structures.")))
)

((expla coll d ((The extent to which man-made structures and alterations collectively have a)
 (negative impact on the visual quality of the natural landscape depends on how)
 ("'visually apparent'" they are in the landscape or the extent to which the)
 (alterations and/or structures visibly alter the natural characterof the)
 ("landscape." Note, this determination is based on how an average lay person)
 (would perceive the "'natural character'" of the landscape "(i.e." a)
 (forester might be able to perceive a variety of non-natural changes in the)
 (landscape such as prescribed burns and stand thinning which would not be)
 (evident of a lay "person." The man-made alterations and/or structures are)
```

```
 (obvious and the natural character of the landscape no longer "predominates,")
 (however, some of the natural character remains and is visually "evident."))))
)

((expla coll e ((The extent to which man-made structures and alterations collectively have a)
 (negative impact on the visual quality of the natural landscape depends on how)
 ("'visually apparent'" they are in the landscape or the extent to which the)
 (alterations and/or structures visibly alter the natural characterof the)
 ("landscape." Note, this determination is based on how an average lay person)
 (would perceive the "'natural character'" of the landscape "(i.e." a)
 (forester might be able to perceive a variety of non-natural changes in the)
 (landscape such as prescribed burns and stand thinning which would not be)
 (evident of a lay "person." The man-made alterations and/or structures are)
 (visually dominant and little or none of the natural character "remains."))))
)

((expla huma y (("'Human scale'" alterations or structures are those that are small enough
that)
 (people feel comfortable when placed in close proximity "(next to, within or on)")
 (to the alteration or "structure." Human scale alterations or structures alter)
 (or change the natural landscape less, and thus, have a less negative influence)
 (on the visual "quality." In some situations, human scale alterations or)
 (structures may even have a positive influence on visual "quality." The man-made)
 ("alteration(s)" or built "structure(s)" in this landscape are human "scale.")
 (This has a positive influence on visual "quality."))))
)

((expla huma n (("'Human scale'" alterations or structures are those that are small enough
that)
 (people feel comfortable when placed in close proximity "(next to, within or on)")
 (to the alteration or "structure." Human scale alterations or structures alter)
 (or change the natural landscape less, and thus, have a less negative influence)
 (on the visual "quality." In some situations, human scale alterations or)
 (structures may even have a positive influence on visual "quality." The man-made)
 ("alteration(s)" or built "structure(s)" in this landscape are not human "scale.")
 (This has no positive influence on visual "quality."))))
)

((expla natu y ((Structures constructed of natural materials appear as if they belong or fit)
 (into the natural "environment." Thus, they appear to alter the natural)
 (landscape a less negative influence on visual "quality." In addition, these)
 (materials also often appear as if it were hand crafted rather than mass)
 ("produced." If the landscape contains more than one man-made alteration or)
 (built structure then the alteration or structure which is most visually)
 (apparent dominates and is responsible for the effect of man-made alterations)
 (or built structures on the visual quality of the "landscape." The most visually)
 (apparent man-made alteration or built structure in this landscape appears to)
 (be constructed of natural "materials." Therefore, this alteration or structure)
 (has a positive influence on visual "quality."))))
```

```
)

((expla natu n ((Structures constructed of natural materials appear as if they belong or fit)
 (into the natural "environment." Thus, they appear to alter the natural)
 (landscape a less negative influence on visual "quality." In addition, these)
 (materials also often appear as if it were hand crafted rather than mass)
 ("produced." If the landscape contains more than one man-made alteration or)
 (built structure then the alteration or structure which is most visually)
 (apparent dominates and is responsible for the effect of man-made alterations)
 (or built structures on the visual quality of the "landscape." The most visually)
 (apparent man-made alteration or built structure in this landscape does not)
 (appear to be constructed of natural "materials.")))
)

((expla shel y ((Landscape which offer the "'potential for shelter'" have high visual "quality,")
 (particularly if combined with "prospect." They "'appear'" to be places that one)
 (could enter and spend the night or seek protection from inclement "weather,")
 (regardless of weather the opportunity to seek shelter really "exists." If the )
 (landscape contains more one man-made alteration or built structure then the)
 (alteration or structure which is most visually apparent dominates and is)
 (responsible for the effect of man-made alterations or built structures on the)
 (visual quality of the "landscape." The most visually apparent man-made)
 (alteration or built structure in this landscape appears to offer the)
 ("'potential for shelter.'" Therefore, this alteration or structure has a)
 (positive influence on visual "quality.")))
)

((expla shel n ((Landscape which offer the "'potential for shelter'" have high visual "quality,")
 (particularly if combined with "prospect." They "'appear'" to be places that one)
 (could enter and spend the night or seek protection from inclement "weather,")
 (regardless of weather the opportunity to seek shelter really "exists." If the )
 (landscape contains more one man-made alteration or built structure then the)
 (alteration or structure which is most visually apparent dominates and is)
 (responsible for the effect of man-made alterations or built structures on the)
 (visual quality of the "landscape." The most visually apparent man-made)
 (alteration or built structure in this landscape does not appear to offer the)
 ("'potential for shelter.'")))
)

((expla pros y ((Landscapes which offer the "'opportunity for prospect.'" have high visual)
 (quality, particularly if they also offer the potential for "shelter.")
 (Alterations or structures appear to improve the opportunity for prospect or the)
 (opportunity to survey the surrounding landscape "visually." The structures or)
 (alterations appear to offer superior vantage points if one were to enter or)
 (climb on top of the structure or alteration, regardless of weather the)
 (opportunity to do so really "exists." If the landscape contains more than one)
 (man-made alteration or built structure then the alteration or sstructure which)
 (is most visually apparent dominates and is responsible for the effect of)
 (man-made alterations or built structures on the visual quality of the)
```

```
 ("landscape." The most visually apparent man-made alteration or built structure)
 (or built structure in this landscape appears to offer the "'opportunity for")
 ("prospect.'" Therefore, this alteration or structure has a positive influence)
 (on visual "quality.")))
)

((expla pros n ((Landscapes which offer the "'opportunity for prospect.'" have high visual)
 (quality, particularly if they also offer the potential for "shelter.")
 (Alterations or structures appear to improve the opportunity for prospect or the)
 (opportunity to survey the surrounding landscape "visually." The structures or)
 (alterations appear to offer superior vantage points if one were to enter or)
 (climb on top of the structure or alteration, regardless of weather the)
 (opportunity to do so really "exists." If the landscape contains more than one)
 (man-made alteration or built structure then the alteration or sstructure which)
 (is most visually apparent dominates and is responsible for the effect of)
 (man-made alterations or built structures on the visual quality of the)
 ("landscape." The most visually apparent man-made alteration or built structure)
 (or built structure in this landscape does not appear to offer the)
 ("'opportunity for prospect.'")))
)

((expla refl y ((The potential for relaxation and contemplation in a landscape has a positive)
 (influence on the visual quality of the "landscape." These are alterations or)
 (structures in the landscape which appear to be quiet and/or serene "places,")
 (regardless of whether they actually "are." If the landscape contains more than)
 (one man-made alteration or built structure then the alteration or structure)
 (which is most visually apparent dominates and is responsible for the effect of)
 (man-made alterations or built structures on the visual quality of the)
 ("landscape." The most visually apparent man-made alteration or built structure)
 (in this landscape appears to have "'potential for relaxation and")
 ("contemplation.'" Therefore, this alteration or structure has a positive has a)
 (positive influence on visual "quality.")))
)

((expla refl n ((The potential for relaxation and contemplation in a landscape has a positive)
 (influence on the visual quality of the "landscape." These are alterations or)
 (structures in the landscape which appear to be quiet and/or serene "places,")
 (regardless of whether they actually "are." If the landscape contains more than)
 (one man-made alteration or built structure then the alteration or structure)
 (which is most visually apparent dominates and is responsible for the effect of)
 (man-made alterations or built structures on the visual quality of the)
 ("landscape." The most visually apparent man-made alteration or built structure)
 (in this landscape does not appear to have "'potential for relaxation and")
 ("contemplation.'")))
)

((expla visb_coll r6_a_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
 (structure influences visual quality depends on how visually dominate it is)
```

```
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are)" visually evident "(reted as 'modified or dominant')." The collective)
  (apparentness rating of all alterations or structures is "'modified'.")))
 )

 ((expla visb_coll r6_a_2 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are)" visually evident "(reted as 'modified or dominant')." The collective)
  (apparentness rating of all alterations or structures is "'dominant'.")))
 )

 ((expla visb_coll r6_b ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are)" visually evident "(reted as 'modified or dominant')."))
 )

 ((expla visb_coll r6_c_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'barely visible'." The collective apparentness rating of all)
  (alterations or structures is "'slightly visible'.")))
 )

 ((expla visb_coll r6_c_2 ((The extent to which the "'most visually apparent'" man-made
```

```
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible'." The collective apparentness rating of all)
   (alterations or structures is "'noticeable'.")))
  )

  ((expla visb_coll r6_c_3 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible'." The collective apparentness rating of all)
   (alterations or structures is "'clearly visible'.")))
  )

  ((expla visb_coll r6_c_4 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible'." The collective apparentness rating of all)
   (alterations or structures is "'modified'.")))
  )

  ((expla visb_coll r6_c_5 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible'." The collective apparentness rating of all)
   (alterations or structures is "'dominant'.")))
```

```
 )

 ((expla visb_coll r6_d_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'noticeable'." The collective apparentness rating of all)
  (alterations or structures is "'noticeable'.")))
 )

 ((expla visb_coll r6_d_2 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'noticeable'." The collective apparentness rating of all)
  (alterations or structures is "'clearly visible'.")))
 )

 ((expla visb_coll r6_d_3 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'noticeable'." The collective apparentness rating of all)
  (alterations or structures is "'modified'.")))
 )

 ((expla visb_coll r6_d_4 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
```

```
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'noticeable'." The collective apparentness rating of all)
   (alterations or structures is "'dominant'.")))
  )

  ((expla visb_coll r6_e_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) visually evident (rated as 'clearly visible' or 'modified')." The)
   (collective apparentness rating of all alterations or structures is)
   ("'clearly visible'.")))
  )

  ((expla visb_coll r6_e_2 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) visually evident (rated as 'clearly visible' or 'modified')." The)
   (collective apparentness rating of all alterations or structures is "'modified'.")))
  )

  ((expla visb_coll r6_e_3 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) visually evident (rated as 'clearly visible' or 'modified')." The)
   (collective apparentness rating of all alterations or structures is "'dominant'.")))
  )

  ((expla visb_coll r6_f_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
```

```
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) visually evident (rated as 'clearly visible' or 'modified')." The)
   (collective apparentness rating of all alterations or structures is "'dominant'."))))
  )

  ((expla visb_coll r6_g_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
   (apparentness rating of all alterations or structures is 'slightly visible' or)
   ("'noticeable'.")))
  )

  ((expla visb_coll r6_g_2 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
   (apparentness rating of all alterations or structures is "'clearly visible'."))))
  )

  ((expla visb_coll r6_g_3 ((The extent to which the "'most visually apparent'" man-made
alteration or)
   (structure influences visual quality depends on how visually dominate it is)
   (relative to the other man-made alterations or structures which are present in)
   (the "landscape." The greater the difference between the apparentness rating)
   (of the "'most visually apparent'" alteration or structure and the collective)
   (apparentness rating of all alterations or structures the greater the influence)
   (of the "'most visually apparent'" alteration or structure on visual "quality.")
   (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
   ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
   (apparentness rating of all alterations or structures is "'modified'" or)
   ("'dominant'.")))
  )
```

```
 ((expla visb_coll r6_h_1 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
  (apparentness rating of all alterations or structures is "'slightly visible'.")))
 )

 ((expla visb_coll r6_h_2 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
  (apparentness rating of all alterations or structures is "'noticeable'" or)
  ("'clearly visible'.")))
 )

 ((expla visb_coll r6_h_3 ((The extent to which the "'most visually apparent'" man-made
alteration or)
  (structure influences visual quality depends on how visually dominate it is)
  (relative to the other man-made alterations or structures which are present in)
  (the "landscape." The greater the difference between the apparentness rating)
  (of the "'most visually apparent'" alteration or structure and the collective)
  (apparentness rating of all alterations or structures the greater the influence)
  (of the "'most visually apparent'" alteration or structure on visual "quality.")
  (The "'most visually apparent'" man-made "alteration(s)" or "structure(s)")
  ("is(are) 'barely visible', 'noticeable'" or "'clearly visible'." The collective)
  (apparentness rating of all alterations or structures is "'modified'" or)
  ("'dominant'.")))
 )
)
```

APPENDIX B

CODE OF NATURAL FEATURE SUBSYSTEM

```
(assert
  ((do_nat) if
   (nat)
   (go_nat)
  )
)


(assert
  ((nat) if
   (system cls)
   (println "***********************************************")
   (println "**                                           **")
   (println "**      Visual Quality Component I: Landscape content     **")
   (println "**              Part B -- NATURAL Feature                 **")
   (println "**                                           **")
   (println "***********************************************")
   (nl)(nl)
   (println "To evaluate natural features, you need to answer the following questions.")
   (nl)(nl)
   (open stream "TXB1:" read)
   (println "Q1: Are there any visually significant natural features in the landscape?")
   (nl)(nl)
   (println "            PLEASE ANSWER Y/N")
   (getline stream ?alteration)
   (continue1 ?alteration)
   (nl)(nl)
   (system cls)
   (println "Q2: Please indicate how many visually striking features are visbile")
   (nl)(nl)
   (println "        PLEASE INPUT THE NUMBER OF STRIKING NATURAL FEATURES")
   (getline stream ?no_of_alter)
   (:= ?no_alter (strtonum ?no_of_alter))
   (assert ((count ?no_alter)))
   (nl)(nl)
   (whether1 ?no_alter)
  )
)


(assert
  ((continue1 ?c) if
   (or (= = ?c n) (= = ?c N))
   (cut)
   (system cls)
   (expla nat n ?expla)
   (assert ((page 20)))
   (print_exp ?expla) ; This ?expla should include CONCLUSION 1 *********
   (retract ((page ?)))
```

```
   (nl)(nl)
   (println "You now can go to the SPATIAL ORGANIZATION part by hitting RETURN key.")
   (getline stream ?a)
   (again do_nat ?a)
  )

  ((continue1 ?c) if
   (or (= = ?c y) (= = ?c Y))
   (cut)
  )

  ((continue1 ?) if
   (println "Your input is not right, try again!")
   (getline stream ?b)
   (continue1 ?b)
  )
)



(assert
  ((again ? q) if
   (quit)
  )

  ((again ?pred s) if
   (close stream)
   (?pred)
  )

  ((again ?pred ?a) if
   (!= ?a q)
   (!= ?a s)
   (println "Do you want to QUIT or START again, input (q/s).")
   (getline stream ?b)
   (again ?pred ?b)
  )
)


(assert
  ((whether1 ?count) if
   (> ?count 3)
   (println "The features are not visually striking if more than three are identified.")
   (println "Please reconsider your answer, input a new number.")
   (getline stream ?no_alter)
   (:= ?count1 (strtonum ?no_alter))
   (retract ((count ?)))
   (assert ((count ?count1)))
```

```
   (whether1 ?count1)
  )

  ((whether1 ?count) if
   (< = ?count 0)
   (println "Not consistent. You should input an integer that is greater than zero !")
   (nl)
   (println "Do you want to QUIT or START again?  Input (q/s)")
   (getline stream ?again)
   (do_over nat count ?again)
  )

  ((whether1 ?count) if
   (> ?count 0)
   (< = ?count 3)
   (nat_name ?count)
   (nat_desc ?count)
   (nat_reac ?count)
   (nat_visb ?count)
   (nat_intr ?count)
  )
)


(assert
  ((do_over ? ? q) if
   (quit)
  )

  ((do_over ?pred1 ?pred2 s) if
   (retract ((?pred2 ?)))
   (close stream)
   (?pred1)
  )

  ((do_over ?pred1 ?pred2 ?a) if
   (! = ?a q)
   (! = ?a s)
   (nl)(nl)(nl)(nl)
   (println "Your input is not appropriate, input (q/s) only.")
   (getline stream ?b)
   (do_over ?pred1 ?pred2 ?b)
  )
)



(assert
  ((show_common ?num) if
```

```
    (println "You have identified the following visually striking natural")
    (println "features being present in the landscape being assessed.")
    (nl)
    (show_names 1 ?num)
   )
 )


 (assert
   ((show_names ? 0) if
    (nl)
   )

   ((show_names ?up ?down) if
    (print "          ")
    (print ?up)
    (print ". ")
    (name1 ?up ?name)
    (println ?name)
    (:= ?new_up (+ ?up 1))
    (:= ?new_down (- ?down 1))
    (show_names ?new_up ?new_down)
   )
)


(assert
  ((nat_name ?end_cond) if
   (system cls)
   (println "Q3: Please list and name EACH visually striking natural feature.")
   (nl)(nl)
   (name1_loop 1 ?end_cond)
  )
)




(assert
  ((nat_desc ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q4: Please list and briefly describe the visual characteristics of each")
   (println "visually striking features. The description should include the size,")
   (println "color, texture, visibility and position in the landscape as well as")
   (println "any other important aspects of the feature's appearance.")
   (nl)(nl)
   (desc1_loop 1 ?end_cond)
  )
```

```
)


(assert
  ((nat_reac ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q5: Would you react differently to this landscape if the following feature")
   (println "did not exist in the landscape.")
   (nl)
   (reac_loop 1 ?end_cond)
  )
)


(assert
  ((nat_visb ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q6: How visually apparent is the following visually striking feature?")
   (nl)
   (visb1_loop 1 ?end_cond)
  )
)


(assert
  ((nat_intr ?end_cond) if
   (system cls)
   (show_common ?end_cond)
   (println "Q7: How interesting or fascinating is the following visually striking features?")
   (nl)
   (intr_loop 1 ?end_cond)
  )
)


(assert
  ((name1_loop ? 0))
  ((name1_loop ?up ?down) if
   (print "Please input the name for the feature NO. ")
   (print ?up)
   (println ".")
   (getline stream ?name)
   (nl)(nl)(nl)(nl)
   (assert ((name1 ?up ?name)))
```

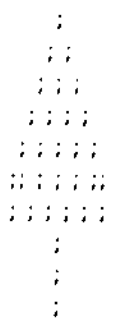
```
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (name1_loop ?new_up ?new_down)
  )
 )


 (assert
  ((desc1_loop ? 0))
  ((desc1_loop ?up ?down) if
   (print "INPUT FOR FEATURE: ")
   (name1 ?up ?name)
   (print ?name)
   (println ".")
   (nl)
   (println "BE SURE TO BRACKET YOUR DESCRIPTION BY '(' & ')'.")
   (println "IF YOU DO NOT HAVE ANYTHING TO INPUT, TYPE IN () FOR AN EMPTY INPUT.")
   (println "BE SURE NOT TO USE ANY '.' SIGN INSIDE THE BRACKET.")
   (read ?desc)
   (nl)(nl)(nl)(nl)
   (assert ((desc1 ?up ?desc)))
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (desc1_loop ?new_up ?new_down)
  )
 )



 (assert
  ((reac_loop ? 0))

  ((reac_loop ?up ?down) if
   (print "PLEASE ANSWER y/n FOR THE FEATURE ")
   (name1 ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?react)
   (nl)(nl)(nl)
   (y_or_n_3 ?up ?down ?react)
  )
 )



 (assert
  ((visb1_loop ? 0))
  ((visb1_loop ?up ?down) if
   (paragra8)
```

```
   (print "   FOR:   ")
   (name1 ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?visb)
   (y_or_n_4 ?up visb1 paragra8 ?visb)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (nl)(nl)(nl)(nl)(nl)
   (visb1_loop ?new_up ?new_down)
  )
)


(assert
  ((intr_loop ? 0))
  ((intr_loop ?up ?down) if
   (paragra9)
   (print "   FOR:   ")
   (name1 ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?intr)
   (y_or_n_4 ?up intr paragra9 ?intr)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (nl)(nl)(nl)(nl)(nl)
   (intr_loop ?new_up ?new_down)
  )
)



(assert
  ((paragra8) if
   (nl)
   (println "a. NOT IMPORTANT: The feature is easily visible. It is difficult for the")
   (println "   viewer to spot even when pointed.")
   (nl)
   (println "b. VISIBLE: The feature is visible, but does not standout. The viewer can")
   (println "   easily see the feature if it is pointed, but may easily overlook it if not.")
   (nl)
   (println "c. IMPORTANT: The feature is highly visible but does not dominate the")
   (println "   landscape. The viewer readily notices the feature but it does not capture")
   (println "   his/her attention.")
   (nl)
   (println "d. DOMINANT: Due to its size or position in the landscape the feature is")
   (println "   visually dominant and the viewer's attention is pulled to the feature.")
   (nl)
```

```
   (print "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c, d'.")
  )
)



(assert
  ((paragra9) if
   (println "a. LOW INTEREST: Most prople would find this feature not to be of much interest")
   (println "   and would not interested in exploring the feature further or finding out")
   (println "   additional information about the feature.")
   (println "b. MODERATE INTEREST: Most people would find this feature somewhat
interesting")
   (println "   and would only be moderately interested in exploring or finding out moer")
   (println "   about the feature.")
   (println "c. MODERATE/HIGH INTEREST: Most people would find feature rather interesting
and")
   (println "   would be interested in exploring the feature further or finding out more")
   (println "   about the feature.")
   (println "d. HIGH INTEREST: Most people would be very fascinated by this feature and")
   (println "   would be very interested in exploring the feature further or finding out")
   (println "   more about it.")
   (nl)
   (print "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c, d'.")
  )
)



(assert
  ((y_or_n_3 ?up ?down y) if
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (reac_loop ?new_up ?new_down)
  )

  ((y_or_n_3 ?up ?down n) if
   (println "Since you would not react differently to the landscape without the feature")
   (name1 ?up ?name)
   (print ?name)
   (println ",  so the feature is in fact not visually striking.")
   (nl)
   (println "You can drop the feature " ?name " and start form the beginning (d), or")
   (println "you can reevaluate your reaction to all the features (r).")
   (getline stream ?conform)
   (count ?count)
   (conf ?count ?conform)
  )
```

```
  ((y_or_n_3 ?up ?down ?ans) if
   (!= ?ans y)
   (!= ?ans n)
   (println "Your answer is not appropriate, input only (y/n).")
   (getline stream ?b)
   (y_or_n_3 ?up ?down ?b)
  )
 )



 (assert
  ((y_or_n_4 ?up ?pred ?paragra ?range) if
   (>= ?range a)
   (<= ?range d)
   (assert ((?pred ?up ?range)))
   (cut)
  )

  ((y_or_n_4 ?up ?pred ?paragra ?) if
   (system cls)
   (println "Your input is not correct, please do it over!")
   (nl)
   (?paragra)
   (print "   FOR:   ")
   (name1 ?up ?name)
   (print ?name)
   (println ".")
   (getline stream ?repeat)
   (y_or_n_4 ?up ?pred ?repeat)
  )
 )



 (assert
  ((conf ?count d) if
   (remove 1 ?count name1)
   (remove 1 ?count desc1)
   (retract ((count ?)))
   (close stream)
   (do_nat)
  )

  ((conf ?count r) if
   (nat_reac ?count)
  )

  ((conf ?count ?ans) if
```

```
    (or (!= ?ans d) (!= ?ans r))
    (println "Your input is not right, answer only 'd/r'. Try again!")
    (getline stream ?b)
    (conf ?count ?b)
   )
 )
```



```
; *****  TREE starts here  *****

(assert
  ((go_nat) if
   (count ?no_alter)
   (travel_1 nil ?return)
   (cut)
   (system cls)
   (assert ((page 20)))
   (append ((nat ?no_alter)) ?return ?list)
   (give_exp ?list)      ; be careful here, ?return may be changed
   (nl)
   (loop review4 ?list)   ; also be careful here
```

```
  )
)


(assert
  ((loop ?pred ?return) if
   (println "Do you want to REVIEW the result (r) or CHANGE the input (c) or")
   (print "DO IT OVER (d) or QUIT (q):   ")
   (getline stream ?look)
   (?pred ?return ?look)
  )
)


(assert
  ((review4 ?return q) if
   (quit)
  )

  ((review4 ?return c) if
   (system cls)
   (println "Here is your last input:")
   (nl)(nl)
   (print "LANDSCAPE CHARACTERISTICS          |    ")
   (count ?count)
   (show_obj 1 ?count)
   (print "-------------------------          |    ")
   (show_bar 1 ?count)
   (print "1. VISUAL APPARENTNESS             |    ")
   (show_item visb1 1 ?count)
   (print "2. HOW INTERESTING or FACINATING   |    ")
   (show_item intr 1 ?count)
   (nl)(nl)
   (println "To change your input, choose an appropriate number.")
   (getline stream ?pri_num)
   (:= ?num (strtonum ?pri_num))
   (modify2 ?count ?num)
   (println "Do you want to CHANGE the input (c) or RUN the program (r) or QUIT (q)?")
   (getline stream ?ans)
   (loop1 ?return ?ans)
  )

  ((review4 ? d) if
   (retract ((page ?)))
   (retract ((line ?)))
   (count ?count)
   (retract ((count ?)))
   (remove 1 ?count name1)
```

```
    (remove 1 ?count desc1)
    (remove 1 ?count visb1)
    (remove 1 ?count intr)
    (close stream)
    (do_nat)
   )

   ((review4 ?return r) if
    (retract ((page ?)))
    (assert ((page 20)))
    (system cls)
    (give_exp ?return)
    (loop review4 ?return)
   )

   ((review4 ?return ?a) if
    (!= ?a q)
    (!= ?a c)
    (!= ?a d)
    (!= ?a r)
    (nl)(nl)(nl)(nl)
    (println "Your input is not appropriate, you must input either 'q' or 'c' or 'd' or 'r'.")
    (println "Answer the following question again !")
    (nl)
    (loop review4 ?return)
   )
)



(assert
  ((show_bar ? 0) if (nl)
  )

  ((show_bar ?up ?down) if
   (print "-------  ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_bar ?new_up ?new_down)
  )
)



(assert
  ((show_item ? ? 0) if (nl)
  )
```

```
  ((show_item ?pred ?up ?down) if
   (?pred ?up ?item)
   (print "   ")
   (print ?item)
   (print "      ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_item ?pred ?new_up ?new_down)
  )
)



(assert
  ((show_obj ? 0) if (nl)
  )

  ((show_obj ?up ?down) if
   (print "OBJECT")
   (print ?up)
   (print "   ")
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show_obj ?new_up ?new_down)
  )
)



(assert
  ((loop1 ? q) if
   (quit)
  )

  ((loop1 ? r) if
   (retract ((page ?)))
   (retract ((line ?)))
   (go_nat)
  )

  ((loop1 ?return c) if
   (review4 ?return c)
  )

  ((loop1 ?return ?a) if
   (!= ?a q)
   (!= ?a c)
   (!= ?a r)
   (nl)(nl)
```

```
    (println "Your input is not appropriate, you must input either 'q' or 'c' or 'r'.")
    (println "Answer the following question again !")
    (nl)
    (getline stream ?b)
    (loop1 ?return ?b)
   )
 )


 (assert
   ((modify2 ?count 1) if
    (remove 1 ?count visb1)
    (nat_visb ?count)
   )

   ((modify2 ?count 2) if
    (remove 1 ?count intr)
    (nat_intr ?count)
   )

   ((modify2 ?count ?num) if
    (or (< ?num 1) (> ?num 2))
    (println "Your input is wrong, input a number between 1 and 2 only, try again!")
    (getline stream ?a)
    (:= ?b (strtonum ?a))
    (modify2 ?count ?b)
   )
 )


 (assert
   ((remove ? 0 ?))

   ((remove ?up ?down ?pred) if
    (retract ((?pred ?up ?)))
    (:= ?new_up (+ ?up 1))
    (:= ?new_down (- ?down 1))
    (remove ?new_up ?new_down ?pred)
   )
 )


 (assert
  ((give_exp ((nat.(?num.nil)).?rest)) if
    (>= ?num 1)
    (<= ?num 3)
```

```
    (deal_head (nat.(?num.nil)))
    (print_des 1 ?num)
    (page ?page)
    (page_hold ?page)
    (nl)
    (deal_tail ?rest)
   )
 )


 (assert
   ((redo ? q) if
    (quit)
   )

   ((redo ?num r) if
    (> ?num 5)
    (remove 1 ?num visb)
    (retract ((coll 1 ?)))
    (pre_all_visb ?num)
    (more_t_five)
   )

   ((redo ?num r) if
    (< = ?num 5)
    (remove 1 ?num visb)
    (retract ((coll 1 ?)))
    (pre_visb ?num)
    (larger ?num)
   )

   ((redo ?num ?a) if
    (! = ?a q)
    (! = ?a r)
    (nl)(nl)(nl)(nl)
    (println "Your input is not appropriate, input 'q' for QUIT or")
    (println "input 'r' for RE-EVALUAT the apparentness.")
    (getline stream ?b)
    (redo ?num ?b)
   )
 )


 (assert
   ((show ? 0))

   ((show ?up ?down) if
    (name1 ?up ?name)
    (print "alteration name  : ")
```

```
   (print ?name)
   (visb ?up ?visb)
   (print "            apparentness :")
   (println ?visb)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (show ?new_up ?new_down)
  )
)


(assert
  ((deal_head (?head.(?num.nil))) if
   (nl)
   (expla ?head y ?expla)
   (print_num ?expla ?num)
  )
)


(assert
  ((deal_tail nil))

  ((deal_tail ((?first.(?second.nil)).?rest)) if
   (expla ?first ?second ?expla)
   (print_exp ?expla)
   (page ?page)        ; These two sentences count for the following blank
   (page_hold ?page)   ; line.
   (nl)
   (deal_tail ?rest)
  )
)



(assert
  ((print_num ?rear_part ?num) if
   (print "You identified ")
   (print ?num)
   (print " ")
   (print_exp ?rear_part)
  )
)



(assert
  ((print_exp nil))
```

```
  ((print_exp (?head.?tail)) if
   (page_ctrl ?head)
   (nl)
   (print_exp ?tail)
  )
)


(assert
  ((print_des ? 0))

  ((print_des ?up ?down) if
   (page ?page)
   (page_hold ?page)
   (nl)
   (name1 ?up ?name)
   (page ?page1)
   (page_hold ?page1)
   (println "The object" ?up " is " ?name " that has the following description:")
   (desc1 ?up ?desc)
   (print_exp ?desc)
   (:= ?new_up (+ ?up 1))
   (:= ?new_down (- ?down 1))
   (print_des ?new_up ?new_down)
  )
)



; ***** Control page, every 20 lines of explanation + 2 lines of prompt
; ***** information is a page


(assert
  ((page_ctrl ?line) if
   (page ?page)
   (page_hold ?page)
   (put_exp ?line)
  )
)

; ***** Check the page control sign to decide to move to next page or not *****


(assert
  ((page_hold 0) if
   (retract ((page ?)))
```

```
   (assert ((page 20)))
   (nl)
   (println "     **********     please hit RETURN to continue     **********")
   (getline stream ?anykey)
   (system cls)
   (cut)
  )

  ((page_hold ?page) if
   (> ?page 0)
   (retract ((page ?)))
   (:= ?new_page (- ?page 1))
   (assert ((page ?new_page)))
  )
)



(assert
  ((put_exp nil))

  ((put_exp (?head.?tail)) if
   (print ?head)
   (print " ")
   (put_exp ?tail)
  )
)



; *****  traverse of the tree of natural part starts here  *****


; *****   NODE 4   *****


(assert
  ((travel_1 ?frame ?return) if
   (get_high visb1 ?feed_back)
   (= = (?head.?tail) ?feed_back)
   (features ?head ?tail ?num)
   (co_high ?feed_back ?cohigh ?num)
   (intr ?line1 ?cohigh)
   (visb1 ?line2 ?head)
   (= = ?line1 ?line2)
   (assert ((line ?line2)))
   (visb1 ?line2 ?apparent)
   (t1_appa ?apparent ?num ?frame ?return)
  )
```

```
)

; *****  ALL NODES OF T1_APPA's  *****

; *****  NODE 6  *****


(assert
  ((t1_appa d  ?num ?frame ?return) if
   (append ?frame ((visb1 d)) ?new_frame)
   (t1_feat1  ?num ?new_frame ?return)
  )
)

; *****  NODE 7  *****


(assert
  ((t1_appa c ?num ?frame ?return) if
   (append ?frame ((visb1 c)) ?new_frame)
   (t1_feat2 ?num ?new_frame ?return)
  )
)


; *****  NODE 8  *****


(assert
  ((t1_appa b ?num ?frame ?return) if
   (append ?frame ((visb1 b)) ?new_frame)
   (t1_feat3 ?num ?new_frame ?return)
  )
)


; *****  NODE 9  *****


(assert
  ((t1_appa a ? ?frame ?return) if
   (append ?frame ((visb1 a)) ?middle)
   (append ?middle ((resolution 1)) ?return)
  )
)

; *****  ALL NODES OF T1_FEAT'S  *****

; *****  NODE 10  *****
```

```
(assert
  ((t1_feat1 1 ?frame ?return) if
   (append ?frame ((feat 1)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina1 ?intr ?new_frame ?return)
  )
)


; *****  NODE 11  *****


(assert
  ((t1_feat1 2 ?frame ?return) if
   (append ?frame ((feat 2)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina1 ?intr ?new_frame ?return)
  )
)


; *****  NODE 12  *****


(assert
  ((t1_feat1 3 ?frame ?return) if
   (append ?frame ((feat 3)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina2 ?intr ?new_frame ?return)
  )
)


; *****  NODE 13  *****


(assert
  ((t1_feat2 1 ?frame ?return) if
   (append ?frame ((feat 1)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina3 ?intr ?new_frame ?return)
  )
)
```

```
; *****  NODE 14  *****


(assert
  ((t1_feat2 2 ?frame ?return) if
   (append ?frame ((feat 2)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina4 ?intr ?new_frame ?return)
  )
)


; *****  NODE 15  *****


(assert
  ((t1_feat2 3 ?frame ?return) if
   (append ?frame ((feat 3)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina5 ?intr ?new_frame ?return)
  )
)


; *****  NODE 16  *****


(assert
  ((t1_feat3 1 ?frame ?return) if
   (append ?frame ((feat 1)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina6 ?intr ?new_frame ?return)
  )
)


; *****  NODE 17  *****


(assert
  ((t1_feat3 2 ?frame ?return) if
   (append ?frame ((feat 2)) ?new_frame)
   (line ?num)
   (intr ?num ?intr)
   (t1_fina6 ?intr ?new_frame ?return)
  )
```

```
)

; ***** NODE 18 *****

(assert
 ((t1_feat3 3 ?frame ?return) if
  (append ?frame ((feat 3)) ?new_frame)
  (line ?num)
  (intr ?num ?intr)
  (t1_fina7 ?intr ?new_frame ?return)
 )
)

; ***** ALL NODES OF T1_FINA'S *****

 ; ***** NODE 19 *****

(assert
 ((t1_fina1 d ?frame ?return) if
  (append ?frame ((intr d)) ?new_frame)
  (append ?new_frame ((resolution 10)) ?return)
 )

 ; ***** NODE 20 *****

 ((t1_fina1 c ?frame ?return) if
  (append ?frame ((intr c)) ?new_frame)
  (append ?new_frame ((resolution 10)) ?return)
 )

 ; ***** NODE 21 *****

 ((t1_fina1 b ?frame ?return) if
  (append ?frame ((intr b)) ?new_frame)
  (append ?new_frame ((resolution 10)) ?return)
 )

 ; ***** NODE 22 *****

 ((t1_fina1 a ?frame ?return) if
```

```
   (append ?frame ((intr a)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )
 )

  ; *****  NODE 27  *****

 (assert
  ((t1_fina2 d ?frame ?return) if
   (append ?frame ((intr d)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )

  ; *****  NODE 28  *****

  ((t1_fina2 c ?frame ?return) if
   (append ?frame ((intr c)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )

  ; *****  NODE 29  *****

  ((t1_fina2 b ?frame ?return) if
   (append ?frame ((intr b)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )

  ; *****  NODE 30  *****

  ((t1_fina2 a ?frame ?return) if
   (append ?frame ((intr a)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )
 )

  ; *****  NODE 31  *****

 (assert
  ((t1_fina3 d ?frame ?return) if
```

```
   (append ?frame ((intr d)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )

  ; *****  NODE 32  *****

  ((t1_fina3 c ?frame ?return) if
   (append ?frame ((intr c)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; *****  NODE 33  *****

  ((t1_fina3 b ?frame ?return) if
   (append ?frame ((intr b)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 34  *****

  ((t1_fina3 a ?frame ?return) if
   (append ?frame ((intr a)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )
)

; *****  NODE 35  *****

(assert
  ((t1_fina4 d ?frame ?return) if
   (append ?frame ((intr d)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )

  ; *****  NODE 36  *****

  ((t1_fina4 c ?frame ?return) if
   (append ?frame ((intr c)) ?new_frame)
   (append ?new_frame ((resolution 10)) ?return)
  )
```

```
  ; *****  NODE 37  *****

  ((t1_fina4 b ?frame ?return) if
   (append ?frame ((intr b)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; *****  NODE 38  *****

  ((t1_fina4 a ?frame ?return) if
   (append ?frame ((intr a)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )
)

; *****  NODE 39  *****

(assert
 ((t1_fina5 d ?frame ?return) if
  (append ?frame ((intr d)) ?new_frame)
  (append ?new_frame ((resolution 10)) ?return)
 )

 ; *****  NODE 40  *****

 ((t1_fina5 c ?frame ?return) if
  (append ?frame ((intr c)) ?new_frame)
  (append ?new_frame ((resolution 10)) ?return)
 )

 ; *****  NODE 41  *****

 ((t1_fina5 b ?frame ?return) if
  (append ?frame ((intr b)) ?new_frame)
  (append ?new_frame ((resolution 2)) ?return)
 )

 ; *****  NODE 42  *****

 ((t1_fina5 a ?frame ?return) if
```

```
    (append ?frame ((intr a)) ?new_frame)
    (append ?new_frame ((resolution 3)) ?return)
   )
  )

  ;  *****  NODE 43  *****

  (assert
   ((t1_fina6 d ?frame ?return) if
    (append ?frame ((intr d)) ?new_frame)
    (append ?new_frame ((resolution 3)) ?return)
   )

   ; *****  NODE 44  *****

   ((t1_fina6 c ?frame ?return) if
    (append ?frame ((intr c)) ?new_frame)
    (append ?new_frame ((resolution 3)) ?return)
   )

   ; *****  NODE 45  *****

   ((t1_fina6 b ?frame ?return) if
    (append ?frame ((intr b)) ?new_frame)
    (append ?new_frame ((resolution 1)) ?return)
   )

   ; *****  NODE 46  *****

   ((t1_fina6 a ?frame ?return) if
    (append ?frame ((intr a)) ?new_frame)
    (append ?new_frame ((resolution 1)) ?return)
   )
  )

  ;  *****  NODE 51  *****

  (assert
   ((t1_fina7 d ?frame ?return) if
    (append ?frame ((intr d)) ?new_frame)
```

```
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; *****  NODE 52  *****

  ((t1_fina7 c ?frame ?return) if
   (append ?frame ((intr c)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 53  *****

  ((t1_fina7 b ?frame ?return) if
   (append ?frame ((intr b)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 54  *****

  ((t1_fina7 a ?frame ?return) if
   (append ?frame ((intr a)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )
)



; *****  ALL MISCELLANEOUS PREDICATES  *****

(assert
  ((append nil ?z ?z))
  ((append (?x.?y) ?z (?x.?w)) if
   (append ?y ?z ?w)
  )
)



(assert
  ((get_high ?pred ?feed_back) if
   (count ?count)
   (get_set ?pred ?count ?set)
```

```
   (qsort ?set ?new_set)
   (reverse ?new_set ?feed_back)
  )
 )


 (assert
  ((get_set ? 0 nil))

  ((get_set ?pred ?total (?visb.?add_set)) if
   (?pred ?total ?visb)
   (:= ?down_total (- ?total 1))
   (get_set ?pred ?down_total ?add_set)
  )
 )


 (assert
  ((reverse (?c.?l1) ?l2) if
   (reverse ?l1 ?out)
   (append ?out (?c) ?l2)
  )

  ((reverse nil nil))
 )


 (assert
  ((qsort nil nil))

  ((qsort (?x.?y) ?ans) if
   (partition ?x ?y ?less ?more)
   (qsort ?less ?less1)
   (qsort ?more ?more1)
   (append ?less1 (?x.?more1) ?ans)
  )
 )


 (assert
  ((partition ?x nil nil nil))

  ((partition ?x (?y.?z) (?y.?less) ?more) if
   (< ?y ?x)
   (partition ?x ?z ?less ?more)
  )
```

```
  ((partition ?x (?y.?z) ?less (?y.?more)) if
   (> = ?y ?x)
   (partition ?x ?z ?less ?more)
  )
)


(assert
  ((features ? nil 1))

  ((features ?first (?first.?tail) ?feed_back) if
   (features ?first ?tail ?new_feed)
   (: = ?feed_back (+ ?new_feed 1))
  )

  ((features ?first (?second.?tail) ?feed_back) if
   (! = ?first ?second)
   (features ? nil ?feed_back)
  )
)


(assert
  ((co_high (?first.?) ?cohigh 1) if
   (visb1 ?num ?first)
   (intr ?num ?cohigh)
   (cut)
  )

  ((co_high (?first.?) ?cohigh 2) if
   (visb1 ?num1 ?first)
   (visb1 ?num2 ?first)
   (! = ?num1 ?num2)
   (intr ?num1 ?a)
   (intr ?num2 ?b)
   (bigger ?a ?b ?cohigh)
   (cut)
  )

  ((co_high ? ?cohigh 3) if
   (get_high intr (?cohigh.?))
  )
)


(assert
```

```
  ((bigger ?a ?b ?a) if
   (> = ?a ?b)
  )

  ((bigger ?a ?b ?b) if
   (< ?a ?b)
  )
)



; ***** VARIOUS FACTS *****

(assert
  ((expla resolution 1 ((FINAL CONCLUSION:) (No effect on visual quality))))
  ((expla resolution 2 ((FINAL CONCLUSION:) (High visual quality))))
  ((expla resolution 3 ((FINAL CONCLUSION:) (Morderately high visual quality))))
  ((expla resolution 4 ((FINAL CONCLUSION:) (Morderately low visual quality))))
  ((expla resolution 5 ((FINAL CONCLUSION:) (Low visual quality))))
  ((expla resolution 6 ((FINAL CONCLUSION:) (SlightlY positive effect on visual quality))))
  ((expla resolution 7 ((FINAL CONCLUSION:) (Slightly negative effect on visual quality))))
  ((expla resolution 8 ((FINAL CONCLUSION:) (Not consistent, re-evaluate individual and
collective apparentness rating))))
  ((expla resolution 10 ((FINAL CONCLUSION:) (Outstanding visual quality))))

  ((expla nat n ((There are no natural features present in this "landscape." Therefore, natural)
   (features have no effect on visual "quality." Visual quality will be determined)
   (by other "factors.")()()(FINAL CONCLUSION:) (No effect on visual quality)))
  )

  ((expla nat y ((visually striking natural "feature(s)." The presence of natural)
   (features has a influence on visual "quality." People are often fascinated or)
   (intrigued by visually significant natural "features." A 'visually significant')
   (natural feature is one that sets a landscape apart from form a similar)
   (landscape which does not contain the natural "feature." The presence of one or)
   (more visually significant natural features has a positive influence on the)
   (visual quality of the landscape that overrides other factors, as long as no)
   (negative man-made alterations of the landscape or built structures are)
   ("present.")))
  )

  ((expla visb1 a ((The degree of influence a natural feature has on visual quality depends on
how)
   (visible or visually apparent the feature "is." The greater the visibility or)
   (the more visually apparent, the greater the influence of the natural feature on)
   (visual "quality." If there are more than one feature with different ratings the)
   (feature or features with the highest visual apparentness rating determines the)
   (visual quality of the "landscape." This landscape was found to contain a)
```

```
  (DOMINANT natural "feature." Due to the size or position in the natural feature)
  (in the landscape the feature is visually dominant and the viewer's attention is)
  (pulled to the "feature."))))
 )

 ((expla visb1 b ((The degree of influence a natural feature has on visual quality depends on
how)
  (visible or visually apparent the feature "is." The greater the visibility or)
  (the more visually apparent, the greater the influence of the natural feature on)
  (visual "quality." If there are more than one feature with different ratings the)
  (feature or features with the highest visual apparentness rating determines the)
  (visual quality of the "landscape." This landscape was found to contain an)
  (IMPORTANT natural "feature." The natural feature is highly visible but does not)
  (dominate the "landscape." The viewer readily nmotices the feature but it does)
  (not capture the viewer's "attention."))))
 )

 ((expla visb1 c ((The degree of influence a natural feature has on visual quality depends on
how)
  (visible or visually apparent the feature "is." The greater the visibility or)
  (the more visually apparent, the greater the influence of the natural feature on)
  (visual "quality." If there are more than one feature with different ratings the)
  (feature or features with the highest visual apparentness rating determines the)
  (visual quality of the "landscape." This landscape was found to contain a)
  (VISIBLE natural "feature." The feature is visible, but does not "standout." The)
  (viewer can easily bsee the natural feature if it is pointed, but may easily)
  (overlook it if "not."))))
 )

 ((expla visb1 d ((The degree of influence a natural feature has on visual quality depends on
how)
  (visible or visually apparent the feature "is." The greater the visibility or)
  (the more visually apparent, the greater the influence of the natural feature on)
  (visual "quality." If there are more than one feature with different ratings the)
  (feature or features with the highest visual apparentness rating determines the)
  (visual quality of the "landscape." This landscape was found to contain a NOT)
  (IMPORTANT natural "feature." The feature is barely "visible." It is difficult)
  (for the viewer to spot the natural feature even when "pointed." Natural)
  (feature has no effect on the visual quality of this "landscape." Viaual quality)
  (will be determined by other "factors."))))
 )


 ((expla feat 1 ((When more then one natural feature has the same highest visual
apparentness)
  (rating then there is an additional or cumulative influence on visual "quality.")
  (If two natural features have the same highest apparentness rating the visual)
  (quality will be greater than if just one feature had the same "rating." If)
  (three natural features have the same highest apparentness rating the visual)
```

```
 (quality will be greater than if just one or two features has the same features)
 (had the same "rating." This landscape was found to have one natural feature)
 (with the highest visual apparentness "rating.")))
)

((expla feat 2 ((When more then one natural feature has the same highest visual
apparentness)
 (rating then there is an additional or cumulative influence on visual "quality.")
 (If two natural features have the same highest apparentness rating the visual)
 (quality will be greater than if just one feature had the same "rating." If)
 (three natural features have the same highest apparentness rating the visual)
 (quality will be greater than if just one or two features has the same features)
 (had the same "rating." This landscape was found to have two featural features)
 (with the highest visual apparentness "rating.")))
)

((expla feat 3 ((When more then one natural feature has the same highest visual
apparentness)
 (rating then there is an additional or cumulative influence on visual "quality.")
 (If two natural features have the same highest apparentness rating the visual)
 (quality will be greater than if just one feature had the same "rating." If)
 (three natural features have the same highest apparentness rating the visual)
 (quality will be greater than if just one or two features has the same features)
 (had the same "rating." This landscape was found to have three natural features)
 (with the highest visual apparentness "rating.")))
)

((expla intr a ((The degree of influence that natural feature which is visually "dominant,")
 (dominat, visually important or visible have on visual quality depends on how)
 (interesting and would be fascinating the feature "is." The greater the)
 (influence of the natural feature on visual "quality." This landscape was found)
 (to have natural features of HIGH "INTEREST." People would be very fascinated)
 (by this feature and would be very interested in exploring the feature further)
 (or finding out more about "it.")))
)

((expla intr b ((The degree of influence that natural feature which is visually "dominant,")
 (dominat, visually important or visible have on visual quality depends on how)
 (interesting and would be fascinating the feature "is." The greater the)
 (influence of the natural feature on visual "quality." This landscape was found)
 (to have natural features of MODERATELY/HIGH "INTEREST." People would find this)
 (feature rather interesting and would be interested in exploring the feature)
 (further or finding out more about the "feature.")))
)

((expla intr c ((The degree of influence that natural feature which is visually "dominant,")
 (dominat, visually important or visible have on visual quality depends on how)
 (interesting and would be fascinating the feature "is." The greater the)
 (influence of the natural feature on visual "quality." This landscape was found)
```

```
 (to have natural features of MODERATE "INTEREST." People would find this feature)
 (somewhat interesting and would be moderately interested in exploring or finding)
 (out more about the "feature.")))
 )

((expla intr d ((The degree of influence that natural feature which is visually "dominant,")
 (dominat, visually important or visible have on visual quality depends on how)
 (interesting and would be fascinating the feature "is." The greater the)
 (influence of the natural feature on visual "quality." This landscape was found)
 (to have natural features of LOW "INTEREST." People would find this feature not)
 (to be of much interest and would be interested in exploring the feature further)
 (or finding out additional information about the "feature.")))
 )
)
```

APPENDIX C

CODE OF SPATIAL ORGANIZATION SUBSYSTEM

```
(assert
  ((do_spa) if
   (spa)
   (go_spa)
  )
)


(assert
  ((spa) if
   (system cls)
   (println "*************************************************")
   (println "**                                             **")
   (println "**      Visual Quality Component II: Landscape Content **")
   (println "**              -- SPATIAL ORGANIZATION         **")
   (println "**                                             **")
   (println "*************************************************")
   (nl)(nl)

   (println "To evaluate the spatial organization, you need to answer the following")
   (println "questions.")
   (nl)(nl)
   (println "******    hit the RETURN key to continue    ******")
   (open stream "TXB1:" read)
   (getline stream ?any)
   (obsc_spa)
   (acce_spa)
   (sopen_spa)
   (edge_spa)
   (scale_spa)
   (safe_spa)
   (pect_spa)
   (solit_spa)
  )
)



(assert
  ((obsc_spa) if
   (system cls)
   (println "1. Please indicate which of the following most closely describes the extent to")
   (println "   which the landscape is visible?")
   (paragra3)
   (getline stream ?obsc)
   (check_ans c paragra3 ?obsc)
   (assert ((obsc ?obsc)))
  )
)
```

```
(assert
  ((acce_spa) if
   (system cls)
   (println "2. Please indicate which of the following most closely describes the apparent")
   (println "   ease with which one could walk through the landscape?")
   (paragra4)
   (getline stream ?acce)
   (check_ans d paragra4 ?acce)
   (obsc ?obsc)
   (examine ?obsc ?acce)
  )
)


(assert
  ((sopen_spa) if
   (system cls)
   (println "3. Please indicate which of the following most closely describes the extent to")
   (println "   which open space exists that is enclosed by landscape features such as")
   (println "   vegetation or topography?")
   (paragra5)
   (getline stream ?sopen)
   (check_ans e paragra5 ?sopen)
   (acce ?acce)
   (go_or_not ?acce ?sopen)
  )
)



(assert
  ((edge_spa) if
   (system cls)
   (paragra6)
   (getline stream ?edge)
   (fuzzy ?edge)
  )
)



(assert
  ((scale_spa) if
   (system cls)
   (println "5. Do places exist in the landscape that are human scale?")
   (nl)(nl)
   (println "   PLEASE ANSWER only 'y/n'.")
   (getline stream ?scale)
```

```
   (y_or_n_5 ?scale ?feed_back)
   (assert ((scale ?feed_back)))
  )
 )


 (assert
  ((safe_spa) if
   (system cls)
   (println "6. Does the space appear to provide opportunities for refuge, or safe and")
   (println "   comfortable places for prople to locate themselves in the landscape?")
   (nl)(nl)
   (println "PLEASE ANSWER only 'y/n'.")
   (getline stream ?safe)
   (y_or_n_5 ?safe ?feed_back)
   (assert ((safe ?feed_back)))
  )
 )


 (assert
  ((pect_spa) if
   (system cls)
   (println "7. Does the space appear to provide opportunities for prospect or a vantage")
   (println "   point or points from which most of the surrounding landscape can be seen?")
   (nl)(nl)
   (println "PLEASE ANSWER only 'y/n'.")
   (getline stream ?pect)
   (y_or_n_5 ?pect ?feed_back)
   (assert ((pect ?feed_back)))
  )
 )


 (assert
  ((solit_spa) if
   (system cls)
   (println "8. Does the space appear to be good places for solitude and reflection?")
   (nl)(nl)
   (println "   PLEASE ANSWER only 'y/n'.")
   (getline stream ?solit)
   (y_or_n_5 ?solit ?feed_back)
   (assert ((solit ?feed_back)))
  )
 )
```

```
(assert
 ((examine a a) if
  (retract ((obsc ?)))
  (close stream)
  (quit)          ; In fact, it should go to next part, i.e. VISUAL COMPOSITION
 )

 ((examine a ?acce) if
  (!= ?acce a)
  (cut)
  (nl)(nl)(nl)(nl)
  (retract ((obsc ?)))
  (println "If the input for question 1 is 'a', then the input for question 2")
  (println "should also be 'a'.")
  (println "If the input for question 2 is not 'a', then the input for question 1")
  (println "should also not be 'a'.")
  (nl)(nl)
  (println "Here is your inputs:")
  (println "QUESTION 1:  a")
  (print "QUESTION 2: ")
  (println ?acce)
  (nl)(nl)
  (println "Since there is no conformation of the inputs, you have to reconsider the")
  (println "response for question 1 and 2.")
  (nl)
  (println "          **********    HIT RETURN KEY TO GO ON    **********")
  (getline stream ?any)
  (obsc_spa)
  (acce_spa)
 )

 ((examine b a) if
  (cut)
  (nl)(nl)(nl)(nl)
  (retract ((obsc ?)))
  (println "If the input for question 1 is 'b', then the input for question 2")
  (println "should not be 'a'.")
  (println "If the input for question 2 is 'a', then the input for question 1")
  (println "should not be 'b'.")
  (nl)(nl)
  (println "Here is your inputs:")
  (println "QUESTION 1:  b")
  (println "QUESTION 2:  a")
  (nl)(nl)
  (println "Since there is no conformation of the inputs, you have to reconsider the")
  (println "response for question 1 and 2.")
  (nl)
```

```
   (println "          **********    HIT RETURN KEY TO GO ON    **********")
   (getline stream ?any)
   (obsc_spa)
   (acce_spa)
  )

  ((examine c ?acce) if
   (or (= = ?acce a) (= = ?acce b))
   (cut)
   (retract ((obsc ?)))
   (nl)(nl)(nl)(nl)
   (println "If the input for question 1 is 'c', then the input for question 2")
   (println "should be neither 'a' nor 'b'.")
   (nl)(nl)
   (println "Here is your inputs:")
   (println "QUESTION 1:  c")
   (print "QUESTION 2:  ")
   (println ?acce)
   (nl)(nl)
   (println "Since there is no conformation of the inputs, you have to reconsider the")
   (println "response for question 1 and 2.")
   (nl)
   (println "          **********    HIT RETURN KEY TO GO ON    **********")
   (getline stream ?any)
   (obsc_spa)
   (acce_spa)
  )

  ((examine ?obsc ?acce) if
   (assert ((acce ?acce)))
  )
)



(assert
  ((go_or_not ?acce a) if
   (or (= = ?acce c) (= = ?acce d))
   (cut)
   (nl)(nl)(nl)(nl)
   (println "If the input for question 3 is 'a', then the input for question 2")
   (println "should neither be 'c' nor 'd'.")
   (nl)(nl)
   (println "Here is your inputs:")
   (print "QUESTION 2: ")
   (println ?acce)
   (print "QUESTION 3: a")
   (nl)(nl)
   (println "Since there is no conformation of the inputs, you have to reconsider the")
```

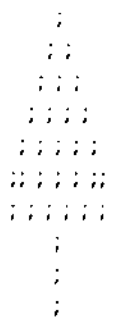
```
   (println "response to question 2 and 3.")
   (nl)
   (println "          **********    HIT RETURN KEY TO GO ON    **********")
   (getline stream ?any)
   (retract ((acce ?)))
   (acce_spa)
   (sopen_spa)
  )

  ((go_or_not ? ?sopen) if
   (assert ((sopen ?sopen)))
  )
)



(assert
  ((check_ans ?end ?para ?ans) if
   (> = ?ans a)
   (< = ?ans ?end)
   (cut)
  )

  ((check_ans ?end ?para ?ans) if
   (or (< ?ans a) (> ?ans ?end))
   (nl)(nl)(nl)
   (println "Your input is inappropriate, do it over")
   (nl)
   (?para)
   (getline stream ?a)
   (check_ans ?end ?para ?a)
  )
)


(assert
  ((fuzzy ?edge) if
   (or (= = ?edge y) (= = ?edge n))
   (assert ((edge ?edge)))
   (cut)
  )

  ((fuzzy c) if
   (system cls)
   (paragra7)
   (getline stream ?sure)
   (sure ?sure)
   (cut)
  )
```

```
  ((fuzzy ?edge) if
   (println "Your input is not right, choose from only 'a' or 'b' or 'c'.  Try again!")
   (getline stream ?a)
   (fuzzy ?a)
  )
)


(assert
  ((sure ?sure) if
   (or (= = ?sure a) (= = ?sure b))
   (assert ((edge y)))
   (cut)
  )

  ((sure ?sure) if
   (or (= = ?sure c) (= = ?sure d))
   (assert ((edge n)))
   (cut)
  )

  ((sure ?sure) if
   (println "Your input is not right, choose from only 'a' or 'b' or 'c' or 'd'. Try again!")
   (getline stream ?a)
   (sure ?a)
  )
)


(assert
  ((y_or_n_5 ?ans ?feed_back) if
   (or (= = ?ans y) (= = ?ans n))
   (= = ?feed_back ?ans)
   (cut)
  )

  ((y_or_n_5 ?ans ?feed_back) if
   (println "Your input is inappropriate, only answer 'y/n', try again!")
   (getline stream ?b)
   (nl)
   (y_or_n_5 ?b ?feed_back)
  )
)
```

```
(assert
  ((paragra3) if
    (nl)
    (println "a. VISIBLE: Little or none of the landscpape is obscured. One can easily")
    (println "   observe most of the landscape.")
    (nl)
    (println "b. PARTIALLY OBSCURED: The landscape is partially obsecured by trees and|or")
    (println "   topography. A portion of the landscape is hidden by intervening obstacles.")
    (nl)
    (println "c. COMPETELY OBSCURED: The view of the landscape is entirely or almost
entirely")
    (println "   obscured by trees and|or topography. Only the immediate foreground is")
    (println "   visible. The rest is hidden by intervening obstacles.")
    (nl)(nl)
    (println "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c'.")
  )
)



(assert
  ((paragra4) if
    (nl)
    (println "a. COMPLETE ACCESS: One could walk relatively unimpeded ove the entire")
    (println "   landscape.")
    (nl)
    (println "b. ACCESS THROUGH AND AROUND: While one or more obstacles exist, it appears
as")
    (println "   if one or more routes are still available to easily walk through the")
    (println "   landscape.")
    (nl)
    (println "c. ACCESS SOMEWHAT IMPEDED: While there appears to be one or more routes
through")
    (println "   the landscape, one would have to step and|or climb over ground level")
    (println "   obstacles such as fallen logs and vegetation.")
    (nl)
    (println "d. ACCESS IMPEDED: There does not appear to be any routes for proceeding
further")
    (println "   into the landscape or potential routes appear to be blocked by a sizeabel")
    (println "   obstacles such as a solid mass of vegetation or stream.")
    (nl)(nl)
    (println "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c, d'.")
  )
)



(assert
  ((paragra5) if
```

```
   (nl)
   (println "a. COMPLETELY OR MOSTLY OPEN: Open space which has few or no vegetative
or")
   (println "   landform features surrounding the space. The space is open on all sides.")
   (nl)
   (println "b. NO OPEN SPACES: The landscape is being viewed from within a stand of")
   (println "   vegetation and appears to be continuous, lacking openings in the vegetation.")
   (nl)
   (println "c. PARTIALLY ENCLOSED: Space that is contained by vegetative or topographic")
   (println "   features on one or two sides, while remaining open on the other sides; or")
   (println "   space that is more or less surrounded by broken or widely spaced vegetative")
   (println "   or topographic features.")
   (nl)
   (println "d. MOSTLY ENCLOSED: Space that is contained by vegetative or topographic")
   (println "   features on all but one side; or space that is more or less surrounded by")
   (println "   closely spaced vegetative or topographic features, but is broken in a few")
   (println "   places.")
   (nl)
   (println "e. ENCLOSED: Space that is contained on all sides or surrounded by vegetative")
   (println "   or topographic features.")
   (nl)
   (println "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c, d, e'.")
  )
)



(assert
  ((paragra6) if
   (nl)
   (println "4. Are the edges of the open space well defined in most palces?")
   (nl)
   (println "CHOOSE ONE CHARACTER FROM  ('y' (yes), 'n' (no) and 'u' (unsure)).")
  )
)


(assert
  ((paragra7) if
   (nl)
   (println "4.a. If you are walking around in this landscape, could you easily tell when")
   (println "     you entered into and when you left the open space?")
   (nl)
   (println "a. YES.")
   (println "b. IN MOST PLACES.")
   (println "c. IN A FEW PLACES.")
   (println "d. NO.")
   (nl)
   (println "PLEASE CHOOSE ONE CHARACTER FROM 'a, b, c, d'.")
```

```
  )
 )
```



```
; *****  TREE starts here  *****

(assert
  ((go_spa) if
   (travel_2 nil ?return)
   (cut)
(println ?return)(nl)
   (system cls)
   (assert ((page 20)))
   (append ((spa 1)) ?return ?list) ; (con 1) added here is only for processing
                    ; of (give_exp). Itself has no real meaning.
   (give_exp ?list)      ; be careful here, ?return may be changed
   (nl)
   (loop review3 ?list)   ; also be careful here
  )
)



(assert
  ((loop ?pred ?return) if
   (println "Do you want to REVIEW the result (r) or CHANGE the input (c) or")
   (print "DO IT OVER (d) or QUIT (q) :   ")
   (getline stream ?look)
   (?pred ?return ?look)
  )
)


(assert
  ((review3 ?return q) if
```

```
 (quit)
)

((review3 ?return c) if
 (system cls)
 (println "Your last input is: ")
 (nl)(nl)
 (println "LANDSCAPE CHARACTERISTICS      |    INPUT VALUE")
 (println "------------------------      |    -----------")
 (print "1. EXTENT OF VISIBILITY         |      ")
 (obsc ?obsc)
 (println ?obsc)
 (print "2. EASE OF ACCESSIBILITY        |      ")
 (acce ?acce)
 (println ?acce)
 (print "3. EXTENT OF ENCLOSION          |      ")
 (sopen ?sopen)
 (println ?sopen)
 (print "4. EDGE OF OPEN SPACE           |      ")
 (edge ?edge)
 (println ?edge)
 (print "5. HUMAN SCALE                  |      ")
 (scale ?scale)
 (println ?scale)
 (print "6. OPPORTUNITIES FOR REFUGE     |      ")
 (safe ?safe)
 (println ?safe)
 (print "7. OPPORTUNITIES FOR PROSPECT   |      ")
 (pect ?pect)
 (println ?pect)
 (print "8. PLACE FOR SOLITUDE           |      ")
 (solit ?solit)
 (println ?solit)
 (nl)(nl)
 (println "To change an entry, type in the appropriate number.")
 (getline stream ?pri_num)
 (:= ?num (strtonum ?pri_num))
 (modify1 ?num)
 (nl)(nl)(nl)(nl)
 (println "Do you want to CHANGE the input (c) or RUN the program (r) or QUIT (q)?")
 (getline stream ?a)
 (loop2 ?a)
)


((review3 ? d) if
 (retract ((page ?)))
 (retract ((obsc ?)))
 (retract ((acce ?)))
```

```
   (retract ((sopen ?)))
   (retract ((edge ?)))
   (retract ((scale ?)))
   (retract ((safe ?)))
   (retract ((pect ?)))
   (retract ((solit ?)))
   (close stream)
   (do_spa)
  )

  ((review3 ?return r) if
   (retract ((page ?)))
   (assert ((page 20)))
   (system cls)
   (give_exp ?return)
   (loop review3 ?return)
  )

  ((review3 ?return ?a) if
   (!= ?a q)
   (!= ?a c)
   (!= ?a d)
   (!= ?a r)
   (nl)(nl)(nl)(nl)
   (println "Your input is not right, you must input either 'q' or 'c' or 'd' or 'r'.")
   (println "Answer the following question again !")
   (nl)
   (loop review3 ?return)
  )
)



(assert
 ((modify1 1) if
  (retract ((obsc ?)))
  (obsc_spa)
 )

 ((modify1 2) if
  (retract ((acce ?)))
  (acce_spa)
 )

 ((modify1 3) if
  (retract ((sopen ?)))
  (sopen_spa)
 )
```

```
  ((modify1 4) if
   (retract ((edge ?)))
   (edge_spa)
  )

  ((modify1 5) if
   (retract ((scale ?)))
   (scale_spa)
  )

  ((modify1 6) if
   (retract ((safe ?)))
   (safe_spa)
  )

  ((modify1 7) if
   (retract ((pect ?)))
   (pect_spa)
  )

  ((modify1 8) if
   (retract ((solit ?)))
   (solit_spa)
  )

  ((modify1 ?a) if
   (or (< ?a 1) (> ?a 8))
   (println "Your choise is not right, pick up a number only between 1 and 8. Try again!")
   (getline stream ?b)
   (:= ?c (strtonum ?b))
   (modify1 ?c)
  )
)



(assert
  ((loop2 q) if
   (quit)
  )

  ((loop2 r) if
   (retract ((page ?)))
   (go_spa)
  )

  ((loop2 c) if
   (review3 ? c)
  )
```

```
  ((loop2 ?a) if
   (!= ?a q)
   (!= ?a r)
   (!= ?a c)
   (println "Your input is not right, choose from only 'r' or 'c' or 'q', try again!")
   (getline stream ?b)
   (loop2 ?b)
  )
)


(assert
  ((give_exp ((man.(?num.nil)).?rest)) if
   (= = ?num 8)
   (println "To have a valid conclusion, your collective apparent value should be no less")
   (println "than the highest apparent value of the alterations.")
   (nl)
   (no_alter ?no_alter)
   (show 1 ?no_alter)
   (nl)
   (print "The collective apparentness:  ")
   (coll 1 ?coll)
   (println ?coll)
   (nl)
   (println "Do you want to QUIT or RE-EVALUATE the apparentness, input (q/r)")
   (getline stream ?choose)
   (redo ?no_alter ?choose)
   (go_exp)
  )

  ((give_exp ((man.(?num.nil)).?rest)) if
   (> = ?num 1)
   (< ?num 9)
   (= = (?head.?tail) ?rest)
   (deal_head ?head)
   (deal_tail ?tail)
  )

  ((give_exp ((spa.(?.nil)).?rest)) if
   (deal_tail ?rest)
  )

)


(assert
  ((redo ? q) if
   (quit)
```

```
  )

  ((redo ?num r) if
   (> ?num 5)
   (remove 1 ?num visb)
   (retract ((coll 1 ?)))
   (pre_all_visb ?num)
   (more_t_five)
  )

  ((redo ?num r) if
   (< = ?num 5)
   (remove 1 ?num visb)
   (retract ((coll 1 ?)))
   (pre_visb ?num)
   (larger ?num)
  )

  ((redo ?num ?a) if
   (! = ?a q)
   (! = ?a r)
   (nl)(nl)(nl)(nl)
   (println "Your input is not appropriate, input 'q' for QUIT or")
   (println "input 'r' for RE-EVALUAT the apparentness.")
   (getline stream ?b)
   (redo ?num ?b)
  )
)


(assert
  ((show ? 0))

  ((show ?up ?down) if
   (name ?up ?name)
   (print "alteration name  : ")
   (print ?name)
   (visb ?up ?visb)
   (print "              apparentness :")
   (println ?visb)
   (: = ?new_up (+ ?up 1))
   (: = ?new_down (- ?down 1))
   (show ?new_up ?new_down)
  )
)



(assert
```

```
  ((deal_head (?head.(?num.nil))) if
   (nl)
   (expla ?head ? ?expla)
   (print_num ?expla ?num)
  )
)


(assert
  ((deal_tail nil))

  ((deal_tail ((?first.(?second.nil)).?rest)) if
   (expla ?first ?second ?expla)
   (print_exp ?expla)
   (page ?page)          ; These two sentences count for the following blank
   (page_hold ?page)     ; line.
   (nl)
   (deal_tail ?rest)
  )
)



(assert
  ((print_num ?rear_part ?num) if
   (print "You identified ")
   (print ?num)
   (print " ")
   (print_exp ?rear_part)
  )
)



(assert
  ((print_exp nil))

  ((print_exp (?head.?tail)) if
   (page_ctrl ?head)
   (nl)
   (print_exp ?tail)
  )
)


; ***** Control page, every 20 lines of explanation + 2 lines of prompt
; ***** information is a page
```

```
(assert
 ((page_ctrl ?line) if
  (page ?page)
  (page_hold ?page)
  (put_exp ?line)
 )
)

; ***** Check the page control sign to decide to move to next page or not *****


(assert
 ((page_hold 0) if
  (retract ((page ?)))
  (assert ((page 20)))
  (nl)
  (println "     **********     please hit RETURN to continue     **********")
  (getline stream ?anykey)
  (system cls)
  (cut)
 )

 ((page_hold ?page) if
  (> ?page 0)
  (retract ((page ?)))
  (:= ?new_page (- ?page 1))
  (assert ((page ?new_page)))
 )
)



(assert
 ((put_exp nil))

 ((put_exp (?head.?tail)) if
  (print ?head)
  (print " ")
  (put_exp ?tail)
 )
)



; *****  traverse of the tree of natural part starts here  *****


; *****  NODE 1  *****
```

```
(assert
  ((travel_2 ?frame ?return) if
   (obsc ?obsc)
   (t2_obsc ?obsc ?frame ?return)
  )
)

; *****  ALL NODES OF T2_OBSC's  *****

; *****  NODE 2 is actually processed when data were input.  *****
; *****  So, there is no need to do anything here.          *****

; *****  NODE 3  *****


(assert
  ((t2_obsc b ?frame ?return) if
   (append ?frame ((obsc b)) ?new_frame)
   (acce ?acce)
   (t2_acce1 ?acce ?new_frame ?return)
  )
)


; *****  NODE 4  *****


(assert
  ((t2_obsc c ?frame ?return) if
   (append ?frame ((obsc c)) ?new_frame)
   (acce ?acce)
   (t2_acce2 ?acce ?new_frame ?return)
  )
)



; *****  ALL NODES OF T2_ACCE'S  *****

; *****  NODE 5, 6, 7, 8, 9 are no needed to be here  *****

; *****  NODE 10  *****

(assert
  ((t2_acce1 b ?frame ?return) if
   (append ?frame ((resolution 2)) ?return)
  )

  ; *****  NODE 11  *****
```

```
  ((t2_acce1 c ?frame ?return) if
   (append ?frame ((acce c)) ?new_frame)
   (sopen ?sopen)
   (t2_open1 ?sopen ?new_frame ?return)
  )

  ; *****  NODE 12  *****

  ((t2_acce1 d ?frame ?return) if
   (append ?frame ((acce d)) ?new_frame)
   (sopen ?sopen)
   (t2_open2 ?sopen ?new_frame ?return)
  )
)


; *****  NODE 13, 14  need not to be here  *****
; *****  NODE 15  *****

(assert
  ((t2_acce2 C ?frame ?return) if
   (append ?frame ((acce c)) ?new_frame)
   (sopen ?sopen)
   (t2_open3 ?sopen ?new_frame ?return)
  )

  ; *****  NODE 16  *****

  ((t2_acce2 d ?frame ?return) if
   (append ?frame ((acce d)) ?new_frame)
   (sopen ?sopen)
   (t2_open4 ?sopen ?new_frame ?return)
  )
)


; *****  ALL NODES OF T2_OPEN'S  *****
; *****  NODE 17 need not to be here  *****
; *****  NODE 18, 20  *****
; *****  NODE 19 is left unmarked because of neglection  *****


(assert
  ((t2_open1 ?sopen ?frame ?return) if
   (or (= = ?sopen b) (= = ?sopen c))
   (append ?frame ((sopen ?sopen)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )
```

```
  ; *****  NODE 21  *****

  ((t2_open1 ?sopen ?frame ?return) if
   (or (= = ?sopen d) (= = ?sopen e))
   (append ?frame ((sopen ?sopen)) ?new_frame)
   (edge ?edge)
   (t2_edge1 ?edge ?new_frame ?return)
  )
)


; *****  NODE 22 need not to be here  *****
; *****  NODE 23  *****
; *****  Here I put in another case, i.e. partial enclosed. Dr. Miller
; *****  forgot to consider this case.

(assert
  ((t2_open2 b ?frame ?return) if
   (append ?frame ((sopen b)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )

  ; *****  NODE 23'  *****

  ((t2_open2 c ?frame ?return) if
   (append ?frame ((sopen c)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 24  *****

  ((t2_open2 ?sopen ?frame ?return) if
   (or (= = ?sopen d) (= = ?sopen e))
   (append ?frame ((sopen ?sopen)) ?new_frame)
   (edge ?edge)
   (t2_edge2 ?edge ?new_frame ?return)
  )
)


; *****  NODE 25 need not to be here  *****
; *****  NODE 26  *****
; *****  Here I put in another case, i.e. partial enclosed. Dr. Miller
; *****  forgot to consider this case.

(assert
  ((t2_open3 b ?frame ?return) if
   (append ?frame ((sopen b)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
```

```
  )

  ; *****  NODE 26'  *****

  ((t2_open3 c ?frame ?return) if
   (append ?frame ((sopen c)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 27  *****

  ((t2_open3 ?sopen ?frame ?return) if
   (or (= = ?sopen d) (= = ?sopen e))
   (append ?frame ((sopen ?sopen)) ?new_frame)
   (edge ?edge)
(println "It goes to EDGE3")
   (t2_edge3 ?edge ?new_frame ?return)
  )
)


; *****  NODE 28 need not to be here  *****
; *****  NODE 29  *****
; *****  Here I put in another case, i.e. partial enclosed. Dr. Miller
; *****  forgot to consider this case.

(assert
  ((t2_open4 b ?frame ?return) if
   (append ?frame ((sopen b)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )

  ; *****  NODE 29'  *****

  ((t2_open4 c ?frame ?return) if
   (append ?frame ((sopen c)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 30  *****

  ((t2_open4 ?sopen ?frame ?return) if
   (or (= = ?sopen d) (= = ?sopen e))
   (append ?frame ((sopen ?sopen)) ?new_frame)
   (edge ?edge)
(println "It goes to EDGE4")
   (t2_edge4 ?edge ?new_frame ?return)
  )
)
```

```
; *****  ALL NODES OF T2_EDGE'S  *****
; *****  NODE 31, 43 and 44 have been processed when data were input.  *****
; *****  So, there is no need to consider about them here.  *****
; *****  NODE 32  *****


(assert
  ((t2_edge1 y ?frame ?return) if
   (append ?frame ((edge y)) ?new_frame)
   (safe ?safe)
   (t2_safe1 ?safe ?new_frame ?return)
  )

  ; *****  NODE 33  *****

  ((t2_edge1 n ?frame ?return) if
   (append ?frame ((edge n)) ?new_frame)
   (safe ?safe)
   (t2_safe2 ?safe ?new_frame ?return)
  )
)



; *****  NODE 34, 49 and 50 have been processed when data were input.  *****
; *****  So, there is no need to consider about them here.  *****
; *****  NODE 35  *****


(assert
  ((t2_edge2 y ?frame ?return) if
   (append ?frame ((edge y)) ?new_frame)
   (safe ?safe)
   (t2_safe3 ?safe ?new_frame ?return)
  )

  ; *****  NODE 36  *****

  ((t2_edge2 n ?frame ?return) if
   (append ?frame ((edge n)) ?new_frame)
   (safe ?safe)
   (t2_safe4 ?safe ?new_frame ?return)
  )
)



; *****  NODE 37, 55 and 56 have been processed when data were input.  *****
```

```
; *****  So, there is no need to consider about them here.  *****
; *****  NODE 38  *****


(assert
 ((t2_edge3 y ?frame ?return) if
  (append ?frame ((edge y)) ?new_frame)
  (safe ?safe)
  (t2_safe5 ?safe ?new_frame ?return)
 )

 ; *****  NODE 39  *****

 ((t2_edge3 n ?frame ?return) if
  (append ?frame ((edge n)) ?new_frame)
  (safe ?safe)
  (t2_safe6 ?safe ?new_frame ?return)
 )
)



; *****  NODE 40, 61 and 62 have been processed when data were input.  *****
; *****  So, there is no need to consider about them here.  *****
; *****  NODE 41  *****


(assert
 ((t2_edge4 y ?frame ?return) if
  (append ?frame ((edge y)) ?new_frame)
  (safe ?safe)
  (t2_safe7 ?safe ?new_frame ?return)
 )

 ; *****  NODE 42  *****

 ((t2_edge4 n ?frame ?return) if
  (append ?frame ((edge n)) ?new_frame)
  (safe ?safe)
  (t2_safe8 ?safe ?new_frame ?return)
 )
)


; *****  ALL NODES OF T2_SAFE'S  *****
; *****  NODE 45  *****


(assert
```

```
  ((t2_safe1 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect1 ?pect ?new_frame ?return)
  )

  ; *****  NODE 46  *****

  ((t2_safe1 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect2 ?pect ?new_frame ?return)
  )
)



; *****  NODE 47  *****


(assert
  ((t2_safe2 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect3 ?pect ?new_frame ?return)
  )

  ; *****  NODE 48  *****

  ((t2_safe2 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect4 ?pect ?new_frame ?return)
  )
)



; *****  NODE 51  *****


(assert
  ((t2_safe3 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect5 ?pect ?new_frame ?return)
  )

  ; *****  NODE 52  *****
```

```
  ((t2_safe3 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect6 ?pect ?new_frame ?return)
  )
)


; *****  NODE 53  *****


(assert
  ((t2_safe4 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect7 ?pect ?new_frame ?return)
  )

  ; *****  NODE 54  *****

  ((t2_safe4 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect8 ?pect ?new_frame ?return)
  )
)


; *****  NODE 57  *****


(assert
  ((t2_safe5 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect9 ?pect ?new_frame ?return)
  )

  ; *****  NODE 58  *****

  ((t2_safe5 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect10 ?pect ?new_frame ?return)
  )
)
```

```
; *****  NODE 59  *****

(assert
  ((t2_safe6 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect11 ?pect ?new_frame ?return)
  )

  ; *****  NODE 60  *****

  ((t2_safe6 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect12 ?pect ?new_frame ?return)
  )
)



; *****  NODE 63  *****

(assert
  ((t2_safe7 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
   (pect ?pect)
   (t2_pect13 ?pect ?new_frame ?return)
  )

  ; *****  NODE 64  *****

  ((t2_safe7 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect14 ?pect ?new_frame ?return)
  )
)



; *****  NODE 65  *****

(assert
  ((t2_safe8 y ?frame ?return) if
   (append ?frame ((safe y)) ?new_frame)
```

```
   (pect ?pect)
   (t2_pect15 ?pect ?new_frame ?return)
  )

  ; *****  NODE 66  *****

  ((t2_safe8 n ?frame ?return) if
   (append ?frame ((safe n)) ?new_frame)
   (pect ?pect)
   (t2_pect16 ?pect ?new_frame ?return)
  )
)


; *****  ALL NODES OF T2_PECT'S  *****
; *****  NODE 67  *****


(assert
  ((t2_pect1 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; *****  NODE 68  *****

  ((t2_pect1 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (solit ?solit)
   (t2_solit1 ?solit ?new_frame ?return)
  )
)


; *****  NODE 69  *****


(assert
  ((t2_pect2 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 70  *****

  ((t2_pect2 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale1 ?scale ?new_frame ?return)
```

```
  )
 )


 ; *****  NODE 71  *****


 (assert
  ((t2_pect3 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 72  *****

  ((t2_pect3 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (solit ?solit)
   (t2_solit2 ?solit ?new_frame ?return)
  )
 )


 ; *****  NODE 73  *****


 (assert
  ((t2_pect4 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 74  *****

  ((t2_pect4 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
 )


 ; *****  NODE 75  *****


 (assert
  ((t2_pect5 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
```

```
  )

  ; ***** NODE 76 *****

  ((t2_pect5 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale1 ?scale ?new_frame ?return)
  )
)


; ***** NODE 77 *****


(assert
  ((t2_pect6 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; ***** NODE 78 *****

  ((t2_pect6 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)


; ***** NODE 79 *****


(assert
  ((t2_pect7 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; ***** NODE 80 *****

  ((t2_pect7 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)
```

```
; ***** NODE 81 *****

(assert
 ((t2_pect8 y ?frame ?return) if
  (append ?frame ((pect y)) ?new_frame)
  (append ?new_frame ((resolution 6)) ?return)
 )

 ; ***** NODE 82 *****

 ((t2_pect8 n ?frame ?return) if
  (append ?frame ((pect n)) ?new_frame)
  (scale ?scale)
  (t2_scale2 ?scale ?new_frame ?return)
 )
)

; ***** NODE 83 *****

(assert
 ((t2_pect9 y ?frame ?return) if
  (append ?frame ((pect y)) ?new_frame)
  (append ?new_frame ((resolution 3)) ?return)
 )

 ; ***** NODE 84 *****

 ((t2_pect9 n ?frame ?return) if
  (append ?frame ((pect n)) ?new_frame)
  (scale ?scale)
  (t2_scale1 ?scale ?new_frame ?return)
 )
)

; ***** NODE 85 *****

(assert
 ((t2_pect10 y ?frame ?return) if
  (append ?frame ((pect y)) ?new_frame)
  (append ?new_frame ((resolution 3)) ?return)
 )

 ; ***** NODE 86 *****
```

```
  ((t2_pect10 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)

; *****  NODE 87  *****

(assert
  ((t2_pect11 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 88  *****

  ((t2_pect11 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)

; *****  NODE 89  *****

(assert
  ((t2_pect12 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 90  *****

  ((t2_pect12 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)

; *****  NODE 91  *****
```

```
(assert
  ((t2_pect13 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 92  *****

  ((t2_pect13 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)


; *****  NODE 93  *****


(assert
  ((t2_pect14 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 94  *****

  ((t2_pect14 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)


; *****  NODE 95  *****


(assert
  ((t2_pect15 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 96  *****

  ((t2_pect15 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
```

```
   (t2_scale2 ?scale ?new_frame ?return)
  )
 )


 ; *****  NODE 97  *****


(assert
  ((t2_pect16 y ?frame ?return) if
   (append ?frame ((pect y)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 98  *****

  ((t2_pect16 n ?frame ?return) if
   (append ?frame ((pect n)) ?new_frame)
   (scale ?scale)
   (t2_scale2 ?scale ?new_frame ?return)
  )
)



; *****  NODE 99  *****

(assert
  ((t2_solit1 y ?frame ?return) if
   (append ?frame ((solit y)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; *****  NODE 100  *****

  ((t2_solit1 n ?frame ?return) if
   (append ?frame ((solit n)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )
)


; *****  NODE 101  *****

(assert
  ((t2_scale1 y ?frame ?return) if
   (append ?frame ((scale y)) ?new_frame)
   (solit ?solit)
   (t2_solit3 ?solit ?new_frame ?return)
```

```
  )

  ; *****  NODE 102  *****

  ((t2_scale1 n ?frame ?return) if
   (append ?frame ((scale n)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )
)


; *****  NODE 103  *****

(assert
  ((t2_solit2 y ?frame ?return) if
   (append ?frame ((solit y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 104  *****

  ((t2_solit2 n ?frame ?return) if
   (append ?frame ((solit n)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )
)


; *****  NODE 105  *****

(assert
  ((t2_scale2 y ?frame ?return) if
   (append ?frame ((scale y)) ?new_frame)
   (solit ?solit)
   (t2_solit4 ?solit ?new_frame ?return)
  )

  ; *****  NODE 106  *****

  ((t2_scale2 n ?frame ?return) if
   (append ?frame ((scale n)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )
)


; *****  NODE 131  *****

(assert
```

```
  ((t2_solit3 y ?frame ?return) if
   (append ?frame ((solit y)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )

  ; *****  NODE 132  *****

  ((t2_solit3 n ?frame ?return) if
   (append ?frame ((solit n)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )
)


; *****  NODE 133  *****

(assert
  ((t2_solit4 y ?frame ?return) if
   (append ?frame ((solit y)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; *****  NODE 134  *****

  ((t2_solit4 n ?frame ?return) if
   (append ?frame ((solit n)) ?new_frame)
   (append ?new_frame ((resolution 1)) ?return)
  )
)


; *****  Since all the other nodes have the same pattern as (node 101, 102,
; *****  131 and 132) and (node 105, 106, 133 and 134), therefore, they are
; *****  not needed any more here.



; *****  ALL MISCELLANEOUS PREDICATES  *****


(assert
  ((append nil ?z ?z))
  ((append (?x.?y) ?z (?x.?w)) if
   (append ?y ?z ?w)
  )
)
```

```
; ***** VARIOUS FACTS *****

(assert

 ((expla resolution 1 ((FINAL CONCLUSION:) (No effect on visual quality))))
 ((expla resolution 2 ((FINAL CONCLUSION:) (High visual quality))))
 ((expla resolution 3 ((FINAL CONCLUSION:) (Morderately high visual quality))))
 ((expla resolution 4 ((FINAL CONCLUSION:) (Morderately low visual quality))))
 ((expla resolution 5 ((FINAL CONCLUSION:) (Low visual quality))))
 ((expla resolution 6 ((FINAL CONCLUSION:) (SlightlY positive effect on visual quality))))
 ((expla resolution 7 ((FINAL CONCLUSION:) (Slightly negative effect on visual quality))))
 ((expla resolution 8 ((FINAL CONCLUSION:) (Not consistent, re-evaluate individual and
collective apparentness rating))))
 ((expla resolution 9 ((FINAL CONCLUSION:) (Outstanding visual quality))))

 ((expla obsc a ((Visiblity is one factor important to the sense of mystery in a "landscape.")
  (Mystery has positive effect on visual "quality." Mystery is THE PROMISE OF)
  (ADDITIONAL INFORMATION IF ONE COULD PROCEED FURTHER INTO THE "LANDSCAPE."
This)
  (landscape was judged to be "visible." Little or none of the landscape is)
  ("obscured." One can easily observe most of the "landscape." Since the entire)
  (landscape is "visible," there is little potential for additional information)
  (and mystery is "low." "Therefore," mystery does not enhance the visual quality)
  (of this "landscape.")))
 )

 ((expla obsc b ((Visiblity is one factor important to the sense of mystery in a "landscape.")
  (Mystery has positive effect on visual "quality." Mystery is THE PROMISE OF)
  (ADDITIONAL INFORMATION IF ONE COULD PROCEED FURTHER INTO THE "LANDSCAPE."
This)
  (landscape was judged to be partial "obscured." The landscape is partially)
  (obscured by trees and|or "topography." A portion of the landscape is hidden by)
  (intervening "obstacles." The secse of mystery would be high since part of the)
  (landscape is "obscured," but glimpses of the landscape beyond are still)
  ("available." Therefore mystery does not enhance the visual quality of this)
  ("landscape.")))
 )

 ((expla obsc c ((Visiblity is one factor important to the sense of mystery in a "landscape.")
  (Mystery has positive effect on visual "quality." Mystery is THE PROMISE OF)
  (ADDITIONAL INFORMATION IF ONE COULD PROCEED FURTHER INTO THE "LANDSCAPE."
This)
  (landscape was judged to be completely "obscured." The view of the landscape is)
  (entirely or almost entirely obscured by trees and|or "topography." Only the)
  (immediate foreground is "visible." The rest is hidden by intervining)
  ("obstacles." If the landscape is obscured there is little hint of potential)
  (information that the landscape may contain and mystery would also be "low.")
  ("Therefore," mystery does enhance the visual quality of this "landscape.")))
 )
```

```
((expla acce a ((Potential accessibility is one factor important to the sense of mystery in a)
 ("lancscape." Mystery is the promise of additional information if one could)
 (proceed further into the "landscape." Mystery is an inportant aspect of the )
 (visual quality of the "landscape." This landscape was judged to offer complete)
 ("access:" One could walk relatively unimpeded over the entire "landscape.")
 (Access over the entire landscape means there is little opportunity for the)
 (landscape to offer additional information that is not already "apparent.")
 ("Thus," mystery is low and there is no affect on visual "quality."))))
)

 ((expla acce b ((Potential accessibility is one factor important to the sense of mystery in a)
 ("lancscape." Mystery is the promise of additional information if one could)
 (proceed further into the "landscape." Mystery is an inportant aspect of the)
 (visual quality of the "landscape." This landscape was judged to offer access)
 (though and around obltacles in the "landscape." While one or more obstacles)
 ("exist," it appears as if one or mor eroutes are available to easily walk)
 (through the "landscape." Easily passable routes through the landscape and)
 (around obstructions increases the potential to acquire additional)
 ("information," and "thus," mystery is also be "high. Therefore," the element)
 (of mystery cuases this landscape to have high visual "quality.")))
)

((expla acce c ((Potential accessibility is one factor important to the sense of mystery in a)
 ("lancscape." Mystery is the promise of additional information if one could)
 (proceed further into the "landscape." Mystery is an inportant aspect of the)
 (visual quality of the "landscape." This landscape was judged to offer access)
 (somewhat impeded through the "landscape." While there appeared to be one or)
 (more routes through the "landscape," one would have to step and|or climb over)
 (ground level obstacles such as fallen logs and "vegetation." If there appears)
 (to be passable routes through and around the "landscape," but considerable)
 (effort would be required in negotiating obstacles then the potential for)
 (obtaining additional information would be "reduced," thus reducing the sense)
 (of "mystery. Therefore," mystery does not enhance the visual quality of this)
 ("landscape.")))
)

((expla acce d ((Potential accessibility is one factor important to the sense of mystery in a)
 ("lancscape." Mystery is the promise of additional information if one could)
 (proceed further into the "landscape." Mystery is an inportant aspect of the)
 (visual quality of the "landscape." This landscape was judged to offer impeded)
 ("access." There does not appear to be any routes for proceeding further into)
 (this landscape or potential routes appear to be blocked by a sizable obstacles)
 (such as a solid mass of vegetation or "stream." Since access into the)
 (landscape appears not to exist or be blocked the potential for obtaining)
 (additional information is low thus the sense of mystery would also be "low.")))
)

((expla sopen a ((Open spaces are places in the landscape which are not filled with trees or)
 (topographic "features." The presence of enclosed and well defined open spaces)
```

(in the landscape has a positive influence on visual "quality," particularly if)
(the spaces are human scale "(pleasant places for people to be)" and or offer)
(opportunities for prospect and "refuge." People feel comfortable in these)
(types of "landscapes." This landscape was judged to be completely or mostly)
("open:" The landscape is opne with few or no vegetative or land form features")
(surrounding and creating a "space. Therefore," the lack of spatial enclosure)
(would have a negative influence on visual "quality.")))
)

((expla sopen b ((Open spaces are places in the landscape which are not filled with trees or)
(topographic "features." The presence of enclosed and well defined open spaces)
(in the landscape has a positive influence on visual "quality," particularly if)
(the spaces are human scale "(pleasant places for people to be)" and or offer)
(opportunities for prospect and "refuge." People feel comfortable in these)
(types of "landscapes." This landscape was judged to have no open "spaces:" The)
(landscape is being viewed from within a stand of vetgetation and appears to be)
(continuous lacking openings in the "vegetation. Therefore," the lack of)
(spatial enclosure would have a negative influence on visual "quality.")))
)

((expla sopen c ((Open spaces are places in the landscape which are not filled with trees or)
(topographic "features." The presence of enclosed and well defined open spaces)
(in the landscape has a positive influence on visual "quality," particularly if)
(the spaces are human scale "(pleasant places for people to be)" and or offer)
(opportunities for prospect and "refuge." People feel comfortable in these)
(types of "landscapes." This landscape was judged to be partially "enclosed:")
(The landscape contains space that is contained by vegetative or topographic)
(features on one or two "sides," while remaining open on the other "sides," or)
(space that is more less surrounded by broken or widely spaced vegetative or)
(topographic "features. Therefore," spatial enclosure would have a slight)
(positive influence on visual "quality.")))
)

((expla sopen d ((Open spaces are places in the landscape which are not filled with trees or)
(topographic "features." The presence of enclosed and well defined open spaces)
(in the landscape has a positive influence on visual "quality," particularly if)
(the spaces are human scale "(pleasant places for people to be)" and or offer)
(opportunities for prospect and "refuge." People feel comfortable in these)
(types of "landscapes." This landscape was judged to be mostly "closed:" The)
(landscape contains space that is contained by vegetative or topographic)
(features on all but one "side," or space that is more less surrounded by)
(closely spaced vegetative or topographic "features," but is broken in a few)
("places. Therefore," spatial enclosure would have a positive influence on)
(visual "quality.")))
)

((expla sopen e ((Open spaces are places in the landscape which are not filled with trees or)
(topographic "features." The presence of enclosed and well defined open spaces)

```
 (in the landscape has a positive influence on visual "quality," particularly if)
 (the spaces are human scale "(pleasant places for people to be)" and or offer)
 (opportunities for prospect and "refuge." People feel comfortable in these)
 (types of "landscapes." This landscape was judged to "be:" This landscape)
 (contains space that is contained on all sides or surrounded by vegetative or)
 (topographic "features. Therefore," spatial enclosure would have a positive)
 (influence on visual "quality.")))
)

((expla edge y ((Spaces with well defined edges are more imageable or "memorable."
Imageability)
 (is an important factor in visual "quality." This landscape was judged to have)
 (spaces with well defined "edges. Therefore," there is a positive influence on)
 (visual "quality.")))
)

((expla edge n ((Since it was difficult to tell when one was entering or leaving a space then
it)
 (has poorly defined "edges. Therefore," there is no positive influence on visual)
 ("quality.")))
)

((expla scale y ((HUMAN SCALE spaces are those that are small enough that people feel)
 (comfortable or secure when located in the "space. Thus," they have the)
 (potential to increase visual "quality," particularly if they also offer)
 (prospect or "refuge." This space was judged to be hUMAN "SCALE," and thus)
 (there is a positive influence on visual "quality.")))
)

((expla scale n ((Need to be done)))
)

((expla safe y ((Landscapes which offer the POTENTIAL for reguge APPEAR to have places
that)
 (one could safely locate "(hide or defend one's self)." This has a positive)
 (influence on visual "quality. Note," that this is regardless of whether the)
 (need or opportunity to seek refuge really "exists." For example a rocky)
 (hillside may appear to be a place where one would likely find refuge in a)
 (small "cave," thus increasing visual "quality." In actuality it may not)
 (provide refuge because no cave "exists." This landscape was judged to have)
 (spaces which offer POTENTIAL for "refuge, and thus," there is a positive)
 (influence on visual "quality.")))
)

((expla safe n ((Need to be done)))
)

((expla safe n ((If a space offer the potential for "refuge," people feel safe to hide or defind)
 (themselves and therefore the visual quality is "improved. However," this)
```

```
   (landscape does not offer the potential for "refuge," it can not improve the)
   (visual "quality.")))
  )

  ((expla pect y (("Landscape," which appear to offer opportunities for "prospect," appear to
have)
   (opportunities to survey the surrounding landscape "visually." The spaces in the)
   (landscape appear to be a place where one could position "one's" self in the)
   (landscape so that one would be able to see any approaching "danger," regardless)
   (of whether the opportunity position "one's" self or potential danger really)
   ("exist." Opportunities was judged to offer opportunities for "prospect," and)
   ("therefors," there is a positive influence on visual "quality.")))
  )

  ((expla pect n ((Need to be done)))
  )

  ((expla pect n ((when a place fail to offer opportunities for prospect or opportunities to)
   (survey the surrounding landscape "visually," the visual quality of the)
   (landscape can not be "improved.")))
  )

  ((expla solit y ((Spaces in the landscape which appear to facilitate or enhance the potential
for)
   (relaxation and contemplation are visually more "appealing," regardless of)
   (whether the opportunity for relaxation and contempaltion actually "exists.")
   (Spaces which appear to be a quiet and serene place facilitate or enhance the)
   (potential for relaxation and "contemplation. For example," a shaded space)
   (adjacent to a stream may appear to be a tranquil and peaceful "place," and)
   ("thus," the visual quality of the landscape would be "enhanced.")))
  )

  ((expla solit n ((Need to be done)))
  )
)
```

APPENDIX D

CODE OF VISUAL COMPOSITION SUBSYSTEM

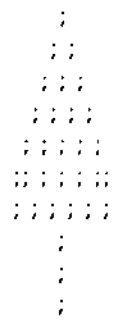
```
(assert
  ((do_vis) if
   (vis)
   (go_vis)
  )
)


(assert
  ((vis) if
   (system cls)
   (println "*****************************************************")
   (println "*                                                   *")
   (println "*          Visual Quality Component III:            *")
   (println "*              VISUAL  COMPOSITION                  *")
   (println "*                                                   *")
   (println "*****************************************************")
   (nl)(nl)
   (println "To evaluate the visual composition, you need to answer the following")
   (println "questions.")
   (nl)(nl)
   (println "******    hit the RETURN key to continue    ******")
   (open stream "TXB1:" read)
   (getline stream ?any)
   (npat_vis)
   (vcom_vis)
   (vivi_vis)
   (svpa_vis)
  )
)


(assert
  ((npat_vis) if
   (paragra23)
   (getline stream ?npat)
   (check_pat ?npat)
  )
)

(assert
  ((vcom_vis) if
   (system cls)
   (paragra20)
   (getline stream ?vcom)
   (check_ans e paragra20 ?vcom)
   (assert ((vcom ?vcom)))
  )
)
```

```
(assert
  ((vivi_vis) if
   (system cls)
   (paragra21)
   (getline stream ?vivi)
   (check_ans e paragra21 ?vivi)
   (assert ((vivi ?vivi)))
  )
)



(assert
  ((svpa_vis) if
   (system cls)
   (paragra22)
   (getline stream ?svpa)
   (check_ans e paragra22 ?svpa)
   (assert ((svpa ?svpa)))
  )
)



(assert
  ((check_ans ?end ?para ?ans) if
   (> = ?ans a)
   (< = ?ans ?end)
   (cut)
  )

  ((check_ans ?end ?para ?ans) if
   (or (< ?ans a) (> ?ans ?end))
   (nl)(nl)(nl)
   (println "Your input is inappropriate, do it over")
   (nl)
   (?para)
   (getline stream ?a)
   (check_ans ?end ?para ?a)
  )
)



(assert
  ((check_pat y) if
   (assert ((npat y)))
```

```
   (paragra24)
   (read ?description)
   (assert ((description ?description)))
  )

  ((check_pat n) if
   (assert ((npat n)))
  )

  ((check_pat ?ans) if
   (or (!= ?ans y) (!= ?ans n))
   (nl)(nl)
   (println "Your answer is not right, choose only 'y/n'. Try again!")
   (getline stream ?b)
   (check_pat ?b)
  )
)



(assert
  ((paragra20) if
   (system cls)
   (println "Q2. Visual complexity is the quantity or number of different visual elements")
   (println "    in the landscape, or the degree of variation in the lines, forms, colors")
   (println "    and textures which comprise the visual landscape. It is a measure of how")
   (println "    intricate or rich the landscape is visually.")
   (nl)(nl)
   (println "a. HIGH COMPLEXITY.")
   (nl)
   (println "b. MODERATELY/HIGH COMPLEXITY.")
   (nl)
   (println "c. MODERATE COMPLEXITY.")
   (nl)
   (println "d. MODERATELY/LOW COMPLEXITY.")
   (nl)
   (println "e. LOW COMPLEXITY.")
   (nl)(nl)
   (println "Please choose one character from 'A, B, C, D, E' to identify the appropriate")
   (println "level of visual complexity for the landscape being assessed.")
  )
)



(assert
  ((paragra21) if
   (system cls)
   (println "Q3. Vividness is the extent to which individual landscape features and/or")
```

```
   (println "    patterns within the landscape are visually distinct and readily visible.")
   (nl)(nl)
   (println "a. HIGH VIVIDNESS.")
   (nl)
   (println "b. MODERATELY/HIGH VIVIDNESS.")
   (nl)
   (println "c. MODERATE VIVIDNESS.")
   (nl)
   (println "d. MODERATELY/LOW VIVIDNESS.")
   (nl)
   (println "e. LOW VIVIDNESS.")
   (nl)(nl)
   (println "Please choose one character from 'A, B, C, D, E' to identify the appropriate")
   (println "level of vividness for the landscape being assessed.")
  )
)



(assert
  ((paragra22) if
   (system cls)
   (println "Q4. 'Striking' visual patterns are that immediately catch the eye and the")
   (println "    attention of the viewer.")
   (nl)(nl)
   (println "a. NOT STRIKING: Visual patterns are not striking or eye catching. The")
   (println "    viewer's attention would not be immediately attracted to the pattern.")
   (nl)
   (println "b. MODERATELY STRIKING: The visual pattern would be apparent, but not
dominate")
   (println "    the visual composition. The visual pattern would attract the viewer's")
   (println "    attention, but the composition would be complex enough that the viewer's")
   (println "    attention would not be dominated by the visual pattern.")
   (nl)
   (println "c. STRIKING: The viusal pattern would dominate the visual composition. The")
   (println "    pattern would immediately catch and hold the viewer's attention.")
   (nl)(nl)
   (println "Please choose one character from 'A, B, C' to identify the appropriate")
   (println "level of striking visual pattern for the landscape being assessed.")
  )
)



(assert
  ((paragra23) if
   (system cls)
   (println "Q1. A visual pattern is created when visual elements (lines, forms,")
   (println "    colors and textures) are repeated in the landscape in a structured")
```

```
   (println "   or regular manner. These visual elements result from the occurrence and/or")
   (println "   repetition of common landscape features, such as tree trunks and geologic")
   (println "   formations. The landscape features should not be 'significant natural")
   (println "   features' either collectively or individually.")
   (nl)
   (println "Please indicate whether visual patterns are present in the landscape")
   (println "being assessed?        Answer   'y/n'   only.")
  )
)


(assert
  ((paragra24) if
   (system cls)
   (println "Q1.1. Please beiefly describe the visual pattern present in the landscape in")
   (println "      terms of its visual characteristics. The description should include the")
   (println "      type of the landscape features (e.g. tree trunks, soil colors and")
   (println "      geologic formation) which contribute to the pattern and description of")
   (println "      the visual elements (lines, forms, colors and textures) which make up")
   (println "      the pattern, as well as any other important aspect of the pattern's")
   (println "      appearance.")
   (nl)(nl)
   (println "BE SURE TO BRACKET YOUR DESCRIPTION BY '(' AND ')'.")
   (println "IF YOU DOT HAVE ANYTHING TO INPUT, TYPE IN () FOR A EMPTY
DESCRIPTION.")
   (println "BE SURE NOT TO USE ANY '.' SIGN INSIDE THE BRACKET.")
  )
)
```



```
; *****  TREE starts here  *****

(assert
  ((go_vis) if
   (travel_4 nil ?return)
   (cut)
   (system cls)
   (assert ((page 20)))
   (append ((vis 1)) ?return ?list) ; (vis 1) added here is only for processing
                    ; of (give_exp). Itself has no real meaning.
   (give_exp ?list)      ; be careful here, ?return may be changed
   (nl)
   (loop review4 ?list)   ; also be careful here
  )
)



(assert
  ((loop ?pred ?return) if
   (println "Do you want to REVIEW the result (r) or CHANGE the input (c) or")
   (print "DO IT OVER (d) or QUIT (q) :   ")
   (getline stream ?look)
   (?pred ?return ?look)
  )
)


(assert
  ((review4 ?return q) if
   (quit)
  )

  ((review4 ?return c) if
   (system cls)
```

```
 (println "Your last input is: ")
 (nl)(nl)
 (println "LANDSCAPE CHARACTERISTICS      |    INPUT VALUE")
 (println "-------------------------      |    -----------")
 (print "1. VISUAL PATTERN               |          ")
 (npat ?npat)
 (println ?npat)
 (print "2. EXTENT OF VISUAL COMPLEXITY  |          ")
 (vcom ?vcom)
 (println ?vcom)
 (print "3. EXTENT OF VIVIDNESS          |          ")
 (vivi ?vivi)
 (println ?vivi)
 (print "4. STRIKING VISUAL PATTERN      |          ")
 (svpa ?svpa)
 (println ?svpa)
 (nl)(nl)
 (println "To change an entry, type in the appropriate number.")
 (getline stream ?pri_num)
 (:= ?num (strtonum ?pri_num))
 (modify4 ?num)
 (nl)(nl)(nl)(nl)
 (println "Do you want to CHANGE the input (c) or RUN the program (r) or QUIT (q)?")
 (getline stream ?a)
 (loop2 go_vis review4 ?a)
)


((review4 ? d) if
 (retract ((page ?)))
 (retract ((npat ?)))
 (retract ((vcom ?)))
 (retract ((vivi ?)))
 (retract ((svpa ?)))
 (retract ((description ?)))
 (close stream)
 (do_vis)
)

((review4 ?return r) if
 (retract ((page ?)))
 (assert ((page 20)))
 (system cls)
 (give_exp ?return)
 (loop review4 ?return)
)

((review4 ?return ?a) if
 (!= ?a q)
```

```
    (!= ?a c)
    (!= ?a d)
    (!= ?a r)
    (nl)(nl)(nl)(nl)
    (println "Your input is not right, you must input either 'q' or 'c' or 'd' or 'r'.")
    (println "Answer the following question again !")
    (nl)
    (loop review4 ?return)
   )
 )


 (assert
   ((modify4 1) if
    (retract ((npat ?)))
    (npat_vis)
   )

   ((modify4 2) if
    (retract ((vcom ?)))
    (vcom_vis)
   )

   ((modify4 3) if
    (retract ((vivi ?)))
    (vivi_vis)
   )

   ((modify4 4) if
    (retract ((svpa ?)))
    (svpa_vis)
   )

   ((modify4 ?a) if
    (or (< ?a 1) (> ?a 4))
    (println "Your choise is not right, pick up a number only between 1 and 4. Try again!")
    (getline stream ?b)
    (:= ?c (strtonum ?b))
    (modify4 ?c)
   )
 )


 (assert
   ((loop2 ? ? q) if
    (quit)
   )
```

```
  ((loop2 ?go ? r) if
   (retract ((page ?)))
   (?go)
  )

  ((loop2 ? ?rev c) if
   (?rev ? c)
  )

  ((loop2 ?go ?rev ?a) if
   (! = ?a q)
   (! = ?a r)
   (! = ?a c)
   (println "Your input is not right, choose from only 'r' or 'c' or 'q', try again!")
   (getline stream ?b)
   (loop2 ?go ?rev ?b)
  )
)



(assert
  ((give_exp ((man.(?num.nil)).?rest)) if
   (= = ?num 8)
   (println "To have a valid conclusion, your collective apparent value should be no less")
   (println "than the highest apparent value of the alterations.")
   (nl)
   (no_alter ?no_alter)
   (show 1 ?no_alter)
   (nl)
   (print "The collective apparentness:   ")
   (coll 1 ?coll)
   (println ?coll)
   (nl)
   (println "Do you want to QUIT or RE-EVALUATE the apparentness, input (q/r)")
   (getline stream ?choose)
   (redo ?no_alter ?choose)
   (go_exp)
  )

  ((give_exp ((man.(?num.nil)).?rest)) if
   (> = ?num 1)
   (< ?num 9)
   (= = (?head.?tail) ?rest)
   (deal_head ?head)
   (deal_tail ?tail)
  )

  ((give_exp ((spa.(?.nil)).?rest)) if
```

```
    (deal_tail ?rest)
   )

   ((give_exp ((vis.(?.nil)).?rest)) if
    (deal_tail ?rest)
   )
 )



 (assert
   ((deal_tail nil))

   ((deal_tail ((?first.(?second.nil)).?rest)) if
    (expla ?first ?second ?expla)
    (print_exp ?expla)
    (page ?page)        ; These two sentences count for the following blank
    (page_hold ?page)   ; line.
    (nl)
    (deal_tail ?rest)
   )
 )



 (assert
   ((print_num ?rear_part ?num) if
    (print "You identified ")
    (print ?num)
    (print " ")
    (print_exp ?rear_part)
   )
 )



 (assert
   ((print_exp nil))

   ((print_exp (?head.?tail)) if
    (page_ctrl ?head)
    (nl)
    (print_exp ?tail)
   )
 )


 ; ***** Control page, every 20 lines of explanation + 2 lines of prompt
 ; ***** information is a page
```

```
(assert
  ((page_ctrl ?line) if
   (page ?page)
   (page_hold ?page)
   (put_exp ?line)
  )
)

; ***** Check the page control sign to decide to move to next page or not *****


(assert
  ((page_hold 0) if
   (retract ((page ?)))
   (assert ((page 20)))
   (nl)
   (println "     **********     please hit RETURN to continue     **********")
   (getline stream ?anykey)
   (system cls)
   (cut)
  )

  ((page_hold ?page) if
   (> ?page 0)
   (retract ((page ?)))
   (:= ?new_page (- ?page 1))
   (assert ((page ?new_page)))
  )
)



(assert
  ((put_exp nil))

  ((put_exp (?head.?tail)) if
   (print ?head)
   (print " ")
   (put_exp ?tail)
  )
)



; *****  traverse of the tree of visual composition starts here  *****


; *****  NODE 1  *****
```

```
(assert
  ((travel_4 ?frame ?return) if
   (npat ?npat)
   (t4_npat ?npat ?frame ?return)
  )
)


; ***** NODE 2 *****

(assert
  ((t4_npat y ?frame ?return) if
   (append ?frame ((npat y)) ?new_frame)
   (svpa ?svpa)
   (t4_svpa ?svpa ?new_frame ?return)
  )

  ; ***** NODE 3 *****

  ((t4_npat n ?frame ?return) if
   (append ?frame ((npat n)) ?new_frame)
   (vcom ?vcom)
   (t4_vcom ?vcom ?new_frame ?return)
  )
)


; ***** NODE 4 *****

(assert
  ((t4_svpa a ?frame ?return) if
   (append ?frame ((svpa a)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )

  ; ***** NODE 5 *****

  ((t4_svpa b ?frame ?return) if
   (append ?frame ((svpa b)) ?new_frame)
   (append ?new_frame ((resolution 6)) ?return)
  )

  ; ***** NODE 6 *****

  ((t4_svpa c ?frame ?return) if
   (append ?frame ((svpa c)) ?new_frame)
```

```
  (vcom ?vcom)
  (t4_vcom ?vcom ?new_frame ?return)
 )
)

; *****  ALL NODES OF 12, 13, 14, 15, 16 and their following
; *****  nodes from NODE 17 to 41 follow the NODE 3

; *****  NODE 7  *****

(assert
 ((t4_vcom a ?frame ?return) if
  (append ?frame ((vcom a)) ?new_frame)
  (vivi ?vivi)
  (t4_vivi1 ?vivi ?new_frame ?return)
 )


 ; *****  NODE 8  *****

 ((t4_vcom b ?frame ?return) if
  (append ?frame ((vcom b)) ?new_frame)
  (vivi ?vivi)
  (t4_vivi2 ?vivi ?new_frame ?return)
 )


 ; *****  NODE 9  *****

 ((t4_vcom c ?frame ?return) if
  (append ?frame ((vcom c)) ?new_frame)
  (vivi ?vivi)
  (t4_vivi3 ?vivi ?new_frame ?return)
 )


 ; *****  NODE 10  *****

 ((t4_vcom d ?frame ?return) if
  (append ?frame ((vcom d)) ?new_frame)
  (vivi ?vivi)
  (t4_vivi4 ?vivi ?new_frame ?return)
 )


 ; *****  NODE 11  *****

 ((t4_vcom e ?frame ?return) if
  (append ?frame ((vcom e)) ?new_frame)
```

```
   (vivi ?vivi)
   (t4_vivi5 ?vivi ?new_frame ?return)
  )
)


; *****  ALL NODE's OF VIVI1  *****

(assert
  ((t4_vivi1 a ?frame ?return) if
   (append ?frame ((vivi a)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )


  ((t4_vivi1 b ?frame ?return) if
   (append ?frame ((vivi b)) ?new_frame)
   (append ?new_frame ((resolution 4)) ?return)
  )


  ((t4_vivi1 c ?frame ?return) if
   (append ?frame ((vivi c)) ?new_frame)
   (append ?new_frame ((resolution 4)) ?return)
  )


  ((t4_vivi1 d ?frame ?return) if
   (append ?frame ((vivi d)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi1 e ?frame ?return) if
   (append ?frame ((vivi e)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )
)


; *****  ALL NODE's OF VIVI2  *****

(assert
  ((t4_vivi2 a ?frame ?return) if
   (append ?frame ((vivi a)) ?new_frame)
   (append ?new_frame ((resolution 2)) ?return)
  )
```

```
  ((t4_vivi2 b ?frame ?return) if
   (append ?frame ((vivi b)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )


  ((t4_vivi2 c ?frame ?return) if
   (append ?frame ((vivi c)) ?new_frame)
   (append ?new_frame ((resolution 3)) ?return)
  )


  ((t4_vivi2 d ?frame ?return) if
   (append ?frame ((vivi d)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )


  ((t4_vivi2 e ?frame ?return) if
   (append ?frame ((vivi e)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )
)



; *****   ALL NODE's OF VIVI3   *****

(assert
  ((t4_vivi3 a ?frame ?return) if
   (append ?frame ((vivi a)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )


  ((t4_vivi3 b ?frame ?return) if
   (append ?frame ((vivi b)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )


  ((t4_vivi3 c ?frame ?return) if
   (append ?frame ((vivi c)) ?new_frame)
   (append ?new_frame ((resolution 9)) ?return)
  )


  ((t4_vivi3 d ?frame ?return) if
```

```
   (append ?frame ((vivi d)) ?new_frame)
   (append ?new_frame ((resolution 4)) ?return)
  )


  ((t4_vivi3 e ?frame ?return) if
   (append ?frame ((vivi e)) ?new_frame)
   (append ?new_frame ((resolution 4)) ?return)
  )
)


; *****  ALL NODE's OF VIVI4  *****

(assert
  ((t4_vivi4 a ?frame ?return) if
   (append ?frame ((vivi a)) ?new_frame)
   (append ?new_frame ((resolution 4)) ?return)
  )


  ((t4_vivi4 b ?frame ?return) if
   (append ?frame ((vivi b)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi4 c ?frame ?return) if
   (append ?frame ((vivi c)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi4 d ?frame ?return) if
   (append ?frame ((vivi d)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi4 e ?frame ?return) if
   (append ?frame ((vivi e)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )
)


; *****  ALL NODE's OF VIVI5  *****
```

```
(assert
  ((t4_vivi5 a ?frame ?return) if
   (append ?frame ((vivi a)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi5 b ?frame ?return) if
   (append ?frame ((vivi b)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi5 c ?frame ?return) if
   (append ?frame ((vivi c)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi5 d ?frame ?return) if
   (append ?frame ((vivi d)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )


  ((t4_vivi5 e ?frame ?return) if
   (append ?frame ((vivi e)) ?new_frame)
   (append ?new_frame ((resolution 5)) ?return)
  )
)




; *****  ALL MISCELLANEOUS PREDICATES  *****


(assert
  ((append nil ?z ?z))
  ((append (?x.?y) ?z (?x.?w)) if
   (append ?y ?z ?w)
  )
)



; *****  VARIOUS FACTS  *****
```

```
(assert

  ((expla resolution 1 ((FINAL CONCLUSION:) (No effect on visual quality))))
  ((expla resolution 2 ((FINAL CONCLUSION:) (High visual quality))))
  ((expla resolution 3 ((FINAL CONCLUSION:) (Morderately high visual quality))))
  ((expla resolution 4 ((FINAL CONCLUSION:) (Morderately low visual quality))))
  ((expla resolution 5 ((FINAL CONCLUSION:) (Low visual quality))))
  ((expla resolution 6 ((FINAL CONCLUSION:) (SlightlY positive effect on visual quality))))
  ((expla resolution 7 ((FINAL CONCLUSION:) (Slightly negative effect on visual quality))))
  ((expla resolution 8 ((FINAL CONCLUSION:) (Not consistent, re-evaluate individual and
collective apparentness rating))))
  ((expla resolution 9 ((FINAL CONCLUSION:) (Moderate visual quality))))

  ((expla  npat y ((Natural occurring visual patterns increase the visual quality of the)
   ("landscape," particularly if they are visually striking or eye "catching." A)
   (visual pattern is created when visual elements "(lines, forms, colors and")
   ("textures)" are repeated in the landscape in a structured or regular "manner.")
   (These visual elements results from the occurrence and/or repetion of common)
   (landscape "features," such as tree trunks and geologic "formations.")))
  )

  ((expla npat n ((To be done))))

  ((expla svpa a ((The extent to which a landscape contains striking visual pattern has a
direct)
   (influence on the visual quality of the "landscape." The pattern observed in)
   (this landscape was found to be visually striking and immediately catches and)
   (holds the attention of the "viewer." The visual pattern dominates the visual)
   ("composition." the pattern causes the landscape to have high visual "quality.")))
  )

  ((expla svpa b ((The extent to which a landscape contains striking visual pattern has a
direct)
   (influence on the visual quality of the "landscape." The pattern observed in)
   (this landscape was found to be moderately visually "striking." It is "apparent,")
   (but does not dominate the visual "composition." The visual pattern attracts)
   (the viewer's attention , but the composition is complex enough that viewer's)
   (attention is not dominated by the visual "pattern." The visual pattern has a)
   (slight positive influence on visual "quality.")))
  )

  ((expla svpa c ((The extent to which a landscape contains striking visual pattern has a
direct)
   (influence on the visual quality of the "landscape." The pattern observed in)
   (this landscape was found not to be striking or eye "catching." The viewer's)
   (attention would not be attracted to the "pattern." And it has no effect on)
   (visual "quality.")))
  )
```

```
 ((expla vcom a ((Visual complexity is directly related to visual quality in natural
"landscapes.")
  (The more visually complex the landscape the greater the visual "quality." This)
  (landscape was judged as having a high degree of visual "complexity." There is)
  (large number of different visual elements in the landscape or high degree of)
  (variation in the lines, forms, colors and textures which is very intricate or)
  (visual "richness.")))
 )

 ((expla vcom b ((Visual complexity is directly related to visual quality in natural
"landscapes.")
  (The more visually complex the landscape the greater the visual "quality." This)
  (landscape was judged as having moderately high visual "complexity." There is a)
  (moderately large number of different visual elements in the landscape or)
  (moderately high degree of variation in the lines, forms, colors and textures)
  (which comprise the visual "landscape." The landscape has a moderate degree of)
  (intricateness or visual "richness.")))
 )

 ((expla vcom c ((Visual complexity is directly related to visual quality in natural
"landscapes.")
  (The more visually complex the landscape the greater the visual "quality." This)
  (landscape was judged as having moderate visual "complexity." There is a)
  (moderate number of different visual elements in the landscape, or a moderate)
  (degree of variation in the lines, forms, colors and textures which comprise the)
  (visual "landscape." The landscape has a moderate degree of intricateness or)
  (visual "richness.")))
 )

 ((expla vcom d ((Visual complexity is directly related to visual quality in natural
"landscapes.")
  (The more visually complex the landscape the greater the visual "quality." This)
  (landscape was judged as having moderately low visual "complexity." There is a)
  (moderately low number of defferent visual elements in the landscape, or)
  (moderately low degree of variation in the lines, forms, colors and textures)
  (which comprise the visual landscape "landscape." The landscape has a moderately)
  (low degree of intricateness or visual "richness.")))
 )

 ((expla vcom e ((Visual complexity is directly related to visual quality in natural
"landscapes.")
  (The more visually complex the landscape the greater the visual "quality." This)
  (landscape was judged as having low visual "complexity." There is a low number)
  (of different visual elements in the landscape or low degree of variation in the)
  (lines, forms, colors and textures which comprise the visual "landscape." The)
  (landscape has a low degree of intricateness or visual "richness.")))
 )
```

```
 ((expla vivi a ((Vividness is directly related to visual quality of the natural "landscape."
The)
  (more vivid a landscape the greater the visual "quality." This landscape was)
  (judged to be highly "vivid." The individual landscape features and/or patterns)
  (within the landscape are very visually distinct and readily "visible." This)
  (landscape is very distinctive and "memorabel.")))
 )

 ((expla vivi b ((Vividness is directly related to visual quality of the natural "landscape."
The)
  (more vivid a landscape the greater the visual "quality." This landscape was)
  (judged to have moderately high "vividness." The individual landscape features)
  (and/or patterns within the landscape have a moderately high degree of visual)
  (distinctiveness and "visiblity." This landscape is moderately high in terms of)
  (distinctiveness and "memorability.")))
 )

 ((expla vivi c ((Vividness is directly related to visual quality of the natural "landscape."
The)
  (more vivid a landscape the greater the visual "quality." This landscape was)
  (judged to have moderate "vividness." The individual landscape features and/or)
  (patterns within the landscape have a moderately low degree of visual)
  (distinctiveness and "visibility." This landscape is moderately low in terms of)
  (distinctiveness and "memorability.")))
 )

 ((expla vivi d ((Vividness is directly related to visual quality of the natural "landscape."
The)
  (more vivid a landscape the greater the visual "quality." This landscape was)
  (judged to have moderately low "vividness." The individual landscape features)
  (and/or patterns within the landscape have a moderately low degree of visual)
  (distinctiveness and "visibility." This landscape is moderately low in terms of)
  (distinctiveness and "memorability.")))
 )

 ((expla vivi e ((Vividness is directly related to visual quality of the natural "landscape."
The)
  (more vivid a landscape the greater the visual "quality." This landscape was)
  (judged to have low "vividness." The individual landscape features and/or)
  (patterns within the landscape have a low degree of visual distinctiveness and)
  ("visibility." This landscape is low in terms of distinctiveness and)
  ("memorability.")))
 )
)
```

June 2,1992

Visual Quality Component I-A: Man-made Features Explanatory Rules

Rule 1 -- Visually evident man-made alterations of the landscape or structures in the natural landscape tend to have a negative influence on visual quality of this type of landscape unless the alteration or structure is awe inspiring, related to historic use of the landscape or are pleasant places for people. The extent of the influence on visual quality is dependent upon how many man-made features are present.

a. This landscape was found to contain no visually evident man-made alterations or structures. Therefore, man-made features has no effect on visual quality.

b. This landscape was found to contain (fill in the number) man-made alterations or structures. The extent of negative or positive influence on the visual quality of the landscape depends on the individual and collective characteristics of the man-made alterations or structures.

c. This landscape was found to contain (fill in the appropriate number 6 or more) man-made alterations or structures. The collective effect on visual quality of this number of alterations will out weigh the effect of any individual alteration.

Rule 2 -- Industrial, quasi-industrial or technological alterations and/or structures in the natural landscape tend to have a negative influence on the visual quality of the landscape unless they are a source of visual interest due to their size (awe inspiring) or historic significance. Industrial, quasi-industrial or technological alterations and /or structures tend to have a negative influence on visual quality because they seem out of place in the natural landscape and contrast with the natural character of the landscape.

a. This landscape was found to contain industrial, quasi-industrial or technological alterations and/or structures.

b. This landscape was not found to contain industrial, quasi-industrial or technological alterations and/or structures.

Rule 3 -- Industrial, quasi-industrial or technological alterations and/or structures in the landscape must be visually apparent in order to inspire awe in people.

a. One or more of the industrial, quasi-industrial or technological alterations and/or structures in this landscape were judged to be visually apparent (rated as modified or dominant for apparentness). Therefore, this landscape has the potential to be awe inspiring.

b. None of the industrial, quasi-industrial or technological alterations and/or structures in this landscape were judged to be visually apparent (rated as modified or dominant for apparentness). Therefore, this landscape does not have the potential to be awe inspiring.

Rule 4 -- Man-made alterations or built structures that inspire awe in people are those that are immense in size such as large pit mines or those that required major engineering feats such as large dams. These types of alterations are fascinating to people and increase the visual quality of the landscape.

a. One or more of the man-made alterations or built structures in this landscape were judged to be awe inspiring.

b. None of the man-made alterations or built structures in this landscape were judged to be awe inspiring.

Rule 5 – Alterations or built structures that "appear" to be old or historic enhance the visual quality of the landscape. They symbolize past cultures and times which tend to fascinate people, thus contributing to people's appreciation of the landscape. If the landscape contains more than one man-made alteration or built structure then the alteration or structure which is most visually apparent dominates and is responsible for the effect of man-made alterations or built structures on the visual quality of the landscape.

a. The most visually apparent man-made alteration or built structure in this landscape appears to be historic. Therefore, this alteration or structure has a positive influence on visual quality.

b. The most visually apparent man-made alteration or built structure in this landscape does not appear to be historic.

Rule 6 -- The extent to which the "most visually apparent" man-made alteration or structure influences visual quality depends on how visually dominate it is relative to the other man-made alterations or structures which are present in the landscape. The greater the difference between the apparentness rating of the "most visually apparent" alteration or structure and the collective apparentness rating of all alterations or structures the greater the influence of the "most visually apparent" alteration or structure on visual quality.

a. The "most visually apparent" man-made alteration(s) or structure(s) is (are) visually evident (rated modified or dominant).

(1) The collective apparentness rating of all alterations or structures is "modified."

(2) The collective apparentness rating of all alterations or structures is "dominant."

b. Collectively the man-made alteration(s) and/or structure(s) contained in this landscape are visually evident (rated as modified or dominant).

c. The "most visually apparent" man-made alteration(s) or structure(s) is (are) "barely visible."

(1) The collective apparentness rating of all alterations or structures is "slightly visible."

(2) The collective apparentness rating of all alterations or structures is "noticeable."

(3) The collective apparentness rating of all alterations or structures is "clearly visible."

(4) The collective apparentness rating of all alterations or structures is "modified."

(5) The collective apparentness rating of all alterations or structures is "dominant."

d. The "most visually apparent" man-made alteration(s) or structure(s) is "noticeable."

(1) The collective apparentness rating of all alterations or structures is "noticeable."

(2) The collective apparentness rating of all alterations or structures is "clearly visible."

(3) The collective apparentness rating of all alterations or structures is "modified."

(4) The collective apparentness rating of all alterations or structures is "dominant."

e. The "most visually apparent" man-made alteration(s) or structure(s) is visually evident (rated as "clearly visible" or "modified").

(1) The collective apparentness rating of all alterations or structures is "clearly visible."

(2) The collective apparentness rating of all alterations or structures is "modified."

(3) The collective apparentness rating of all alterations or structures is "dominant."

f. The "most visually apparent" man-made alteration(s) or structure(s) is visually evident (rated as "clearly visible" or "modified").

(1) The collective apparentness rating of all alterations or structures is "dominant."

g. The "most visually apparent" man-made alteration(s) or structure(s) is (are) "barely visible," "noticeable" or "clearly visible."

(1) The collective apparentness rating of all alterations or structures is "slightly visible" or "noticeable."

(2) The collective apparentness rating of all alterations or structures is "clearly visible."

(3) The collective apparentness rating of all alterations or structures is "modified" or "dominant."

h. The "most visually apparent" man-made alteration(s) or structure(s) is (are) "barely visible," "noticeable" or "clearly visible."

(1) The collective apparentness rating of all alterations or structures is "slightly visible."

(2) The collective apparentness rating of all alterations or structures is "noticeable" or "clearly visible."

(3) The collective apparentness rating of all alterations or structures is "modified" or "dominant."

Rule 7 -- The most visually apparent man-made alteration(s) or structure(s) in the will have the greatest effect on the visual quality of the landscape.

a. The most visually apparent alteration(s) or structure(s) in the landscape are related to industrial, quasi-industrial or technological uses of the landscape.

b. The most visually apparent alteration(s) or structure(s) in the landscape are not related to industrial, quasi-industrial or technological uses of the landscape.

Rule 8 -- The extent to which man-made structures and alterations collectively have a negative impact on the visual quality of the natural landscape depends on how "visually apparent" they are in the landscape or the extent to which the alterations and or structures visibly alter the natural character of the landscape. Note, this is determination is based on how an average lay person would perceive the "natural character" of the landscape (i.e. a professional forester might be able to perceive a variety of non-natural changes in the landscape such as prescribed burns and stand thinning which would not be evident of a lay person).

a. The man-made alterations and/or structures are barely visible and the natural visual character of the landscape has not been altered.

b. The man-made alterations and/or structures are visible, but the natural character of the landscape is essentially intact.

c. The man-made alterations and/or structures are clearly visible, but they are still subordinate to the natural character of the landscape. The natural character has been altered, but the natural character still the predominates. The landscape has a moderately low visual quality due to man-made alterations and structures.

d. The man-made alterations and/or structures are obvious and the natural character of the landscape no longer predominates, however some of the natural character remains and is visually evident.

e The man-made alterations and/or structures are visually dominant and little or none of the natural character remains.

Rule 9 - "Human scale" alterations or structures are those that are small enough that people feel comfortable when placed in close proximity (next to, within or on) to the alteration or structure. Human scale alterations or structures alter or change the natural landscape less, and thus, have a less negative influence on the visual quality. In some situations human scale alterations or structures may even have a positive influence on visual quality.

a. The man-made alteration(s) or built structure(s) in this landscape are human scale. This has a positive influence on visual quality.

b. The man-made alteration(s) or built structure(s) in this landscape are not human scale. This has a positive influence on visual quality.

Rule 10 - Structures constructed of natural materials appear as if they belong or fit into the natural environment. Thus, they appear to alter the natural landscape less and have a less negative influence on visual quality. In addition, these materials also often appear as if it were hand crafted rather than mass produced. The perception of hand craftsmanship has a positive influence on visual quality. If the landscape contains more than one man-made alteration or built structure then the alteration or structure which is most visually apparent dominates and is responsible for the effect of man-made alterations or built structures on the visual quality of the landscape.

a. The most visually apparent man-made alteration or built structure in this landscape appears to be constructed of natural materials. Therefore, this alteration or structure has a positive influence on visual quality.

b. The most visually apparent man-made alteration or built structure in this landscape does not appear to be constructed of natural materials.

Rule 11 - Landscapes which offer the "potential for shelter" have high visual quality, particularly if combined with prospect. They "appear" to be places that one could enter and spend the night or seek protection from inclement weather, regardless of whether the opportunity to seek shelter really exists. If the landscape contains more than one man-made alteration or built structure then the alteration or structure which is most visually apparent dominates and is responsible for the effect of man-made alterations or built structures on the visual quality of the landscape.

a. The most visually apparent man-made alteration or built structure in this landscape appears to offer the "potential for shelter." Therefore, this alteration or structure has a positive influence on visual quality.

b. The most visually apparent man-made alteration or built structure in this landscape does not appear to offer the potential for shelter.

Rule 12 - Landscapes which offer the "opportunity for prospect" have high visual quality, particularly if they also offer the potential for shelter. Alterations or structures appear to improve the opportunity for prospect or the opportunity to survey the surrounding landscape visually. The structures or alterations appear to offer superior vantage points if one were to enter or climb on top of the structure or alteration, regardless of whether the opportunity to do so really exists. If the landscape contains more than one man-made alteration or built structure then the alteration or structure which is most visually apparent dominates and is responsible for the effect of man-made alterations or built structures on the visual quality of the landscape.

a. The most visually apparent man-made alteration or built structure in this landscape appears to offer the "opportunity for prospect." Therefore, this alteration or structure has a positive influence on visual quality.

b. The most visually apparent man-made alteration or built structure in this landscape does not appear to offer the "opportunity for prospect."

Rule 13 - The potential for relaxation and contemplation in a landscape has a positive influence on the visual quality of the landscape. These are alterations or structures in the landscape which appear to be quiet and/or serene places, regardless of whether they actually are. If the landscape contains more than one man-made alteration or built structure then the alteration or structure which is most visually apparent dominates and is responsible for the effect of man-made alterations or built structures on the visual quality of the landscape.

a. The most visually apparent man-made alteration or built structure in this landscape appears to have "potential for relaxation and contemplation." Therefore, this alteration or structure has a positive influence on visual quality.

b. The most visually apparent man-made alteration or built structure in this landscape does not appear to have "potential for relaxation and contemplation."

May 19,1992

Visual Quality Component I-B: Natural Features
Explanatory Rules

Rule 1 -- The presence of natural features has a influence on visual quality. People are often fascinated or intrigued by visually significant natural features. A "visually significant" natural feature is one that sets a landscape apart from form a similar landscape which does not contain the natural feature. The presence of one or more visually significant natural features has a positive influence on the visual quality of the landscape that overrides other factors, as long as no negative man-made alterations of the landscape or built structures are present.

a. There are one or more natural features present in this landscape. The natural feature(s) present in this landscape is (are):

1. (The name and description of the first natural feature should be provided here.)

2. (The name and description of the second natural feature, if one exists, should be provided here.)

3. (The name and description of the third natural feature, if one exists, should be provided here.)

b. There are no natural features present in this landscape. Therefore, natural features have no effect on visual quality. Visual quality will be determined by other factors.

Rule 2 -- The degree of influence a natural feature(s) has on visual quality depends on how visible or visually apparent the feature(s) is (are). The greater the visibility or the more visually apparent, the greater the influence of the natural feature(s) on visual quality. If there are more than one feature with different ratings the feature or features with the highest visual apparentness rating determines the visual quality of the landscape.

a. This landscape was found to contain a **Dominant** natural feature(s). Due to the size or position in the natural feature(s) in the landscape the feature is (are) visually dominant and the viewer's attention is pulled to the feature.

b. This landscape was found to contain an **Important** natural feature(s). The natural feature(s) is (are) highly visible but does (do) not dominate the landscape. The viewer readily notices the feature(s) but it does not capture the viewer's attention.

c. This landscape was found to contain a **Visible** natural feature(s). The feature(s) is (are) visible, but does (do) not standout. The viewer can easily see the natural feature(s) if it (they) is (are) pointed, but may easily overlook it (them) if not.

d. This landscape was found to contain a **Not Important** natural feature(s): The feature(s) is (are)barely visible. It is difficult for the viewer to spot the natural feature(s) even when pointed. Natural features have no effect on the visual quality of this landscape. Visual quality will be determined by other factors.

Rule 3 -- When more then one natural feature has the same highest visual apparentness rating then there is an additional or cumulative influence on visual quality. If two natural features have the same highest apparentness rating the visual quality will be greater than if just one feature had the same

rating. If three natural features have the same highest apparentness rating the visual quality will be greater than if just one or two features had the same rating.

a. This landscape was found to have one natural feature with the highest visual apparentness rating.

b. This landscape was found to have two natural features with the same highest visual apparentness rating.

c. This landscape was found to have three natural features with the same highest visual apparentness rating.

Rule 4 – The degree of influence that natural feature(s) which are visually dominant, visually important or visible have on visual quality depends on how interesting or fascinating the feature(s) is (are). The greater the interest or the more fascinating, the greater the influence of the natural feature(s) on visual quality.

a. This landscape was found to have natural features of **High Interest**. People would be very fascinated by this feature and would be very interested in exploring the feature further or finding out more about it.

b. This landscape was found to have natural features of **Moderately/High Interest**. People would find this feature rather interesting and would be interested in exploring the feature further or finding out more about the feature.

c. This landscape was found to have natural features of **Moderate Interest**. Most people would find this feature somewhat interesting and would only be moderately interested in exploring or finding out more about the feature.

d. This landscape was found to have natural features of **Low Interest**. Most people would find this feature not to be of much interest and would not be interested in exploring the feature further or finding out additional information about the feature..

Visual Quality Component II: Spatial Organization
Explanatory Rules

Rule 1 -- Visibility is one factor important to the sense of mystery in a landscape. Mystery has positive effect on visual quality. Mystery is "the promise of additional information if one could proceed further into the landscape."

a. This landscape was judged to be visible. Little or none of the landscape is obscured. One can easily observe most of the landscape. Since the entire landscape is visible, there is little potential for additional information and mystery is low. Therefore, mystery does not enhance the visual quality of this landscape.

b. This landscape was judged to be partially obscured. The landscape is partially obscured by trees and/or topography. A portion of the landscape is hidden by intervening obstacles. The sense of mystery would be high since part of the landscape is obscured, but glimpses of the landscape beyond are still available. Therefore, mystery does enhance the visual quality of this landscape.

c. This landscape was judged to be completely obscured. The view of the landscape is entirely or almost entirely obscured by trees and/or topography. Only the immediate foreground is visible. The rest is hidden by intervening obstacles. If the landscape is obscured there is little hint of potential information that the landscape may contain and mystery would also be low. Therefore, mystery does enhance the visual quality of this landscape.

Rule 2 -- Potential accessibility is one factor important to the sense of mystery in a landscape. Mystery is the promise of additional information if one could proceed further into the landscape. Mystery is an important aspect of the visual quality of the landscape.

a. This landscape was judged to offer complete access: One could walk relatively unimpeded over the entire landscape Access over the entire landscape means there is little opportunity for the landscape to offer additional information that is not already apparent. Thus, mystery is low and there is no affect on visual quality.

b. This landscape was judged to offer access through and around obstacles in the landscape. While one or more obstacles exist, it appears as if one or more routes are available to easily walk through the landscape. Easily passable routes through the landscape and around obstructions increases the potential to acquire additional information, and thus, mystery is also be high. Therefore, the element of mystery causes this landscape to have high visual quality.

c. This landscape was judged to offer access somewhat impeded through the landscape. While there appeared to be one or more routes through the landscape, one would have to step and/or climb over ground level obstacles such as fallen logs and vegetation. If there appears to be passable routes through and around the landscape, but considerable effort would be required in negotiating obstacles then the potential for obtaining additional information would be reduced, thus reducing

the sense of mystery. Therefore, mystery does enhance the visual quality of this landscape.

d. This landscape was judged to offer impeded access. There does not appear to be any routes for proceeding further into this landscape or potential routes appear to be blocked by a sizable obstacles such as a solid mass of vegetation or stream. Since access into the landscape appears not to exist or be blocked the potential for obtaining additional information is low, thus the sense of mystery would also be low.

Rule 3 – Open spaces are places in the landscape which are not filled with trees or topographic features. they may be completely open, partially enclosed or completely enclosed by vegetation and land forms. The presence of enclosed and well defined open spaces in the landscape has a positive influence on visual quality.

a. This landscape was judged to be completely or mostly open: The landscape is open with few or no vegetative or land form features surrounding and creating a space. Therefore, the lack of spatial enclosure would have a negative influence on visual quality.

b. This landscape was judged to have no open spaces: The landscape is being viewed from within a stand of vegetation and appears to be continuous, lacking openings in the vegetation. Therefore, the lack of spatial enclosure would have a negative influence on visual quality.

c. This landscape was judged to be partially enclosed: The landscape contains space that is contained by vegetative or topographic features on one or two sides, while remaining open on the other sides; or space that is more less surrounded by broken or widely spaced vegetative or topographic features. Therefore, spatial enclosure would have a slight positive influence on visual quality.

d. This landscape was judged to be mostly enclosed or enclosed: This landscape contains space that is contained by vegetative or topographic features on at least all but one side; or space that is more less surrounded by closely spaced vegetative or topographic features, but may be broken in a few places. Therefore, spatial enclosure would have a positive influence on visual quality.

Rule 4 – Spaces with well defined edges are more imageable or memorable. Imageability has a positive influence on visual quality.

a. This landscape was judged to have spaces with well defined edges. Therefore, these spaces will have a positive influence on visual quality.

b. This landscape was judged not to have well defined spaces. Therefore, these spaces will have a less positive influence on visual quality than a spaces with well defined edges.

Rule 5 – Landscapes which contain comfortable spaces for people, or pleasant places for people to be in, have higher visual quality. People feel comfortable in landscapes that have the following attributes: (1) appear to offer opportunities for prospect and/or refuge, or (2) contain spaces which are human scale and provide places for reflection. Landscapes which contain more of these attributes have higher visual quality.

a. This landscape was not found to contain spaces with attributes that cause them to be comfortable spaces or pleasant places for people.

b. This landscape was found to contain spaces with attributes that cause them to be comfortable spaces or pleasant places for people.

(1) The landscape contains human scale spaces. "Human scale" spaces are those that are small enough that people feel comfortable or secure when located in the space. Thus, they have the potential to increase visual quality, particularly if they also offer prospect or refuge. A space the size of an average camp site is human scale. A football field or most clear-cuts are not human scale. Thus, there is a positive influence on the visual quality of this landscape.

(2) This landscape was judged to have spaces which offer "potential" for refuge. Landscapes which offer the "potential" for refuge "appear" to have places that one could safely locate (hide or defend one's self). This has a positive influence on visual quality. Note, that this is regardless of whether the need or opportunity to seek refuge really exists. For example a rocky hillside may appear to be a place where one would likely find refuge in a small cave, thus increasing visual quality. In actuality it may not provide refuge because no cave exists. Thus, there is a positive influence on the visual quality of this landscape.

(3) This landscape was judged to offer opportunities for prospect. Landscapes, which appear to offer opportunities for prospect, appear to have opportunities to survey the surrounding landscape visually. The spaces in the landscape appear to be a place where one could position one's self in the landscape so that one would be able to see any approaching danger, regardless of whether the opportunity position one's self or potential danger really exist. Opportunities for prospect in the landscape have a positive influence on visual quality. Therefore, there is a positive influence on the visual quality of this landscape.

(4) This landscape was judged to offer opportunities for solitude and reflection. The potential for solitude and reflection has a positive influence on visual quality. Opportunities for solitude and reflection increase the potential for relaxation and contemplation in the landscape, regardless of whether the opportunity for relaxation and contemplation actually exits. For example, a space adjacent to a stream may appear to be a tranquil and peaceful place, Thus, the visual quality of the landscape would be enhanced, regardless of the fact that in actuality highway noise may destroy the tranquillity and peacefulness of the space. Opportunities for solitude and reflection in the landscape have a positive influence on visual quality. Therefore, there is a positive influence on the visual quality of this landscape.

Visual Quality Component III: Visual Composition
Explanatory Rules

Rule 1. Naturally occurring visual patterns increase the visual quality of the landscape, particularly if they are visually striking or eye catching. A visual pattern is created when visual elements (lines, forms colors and textures) are repeated in the landscape in a structured or regular manner. These visual elements result from the occurrence and/or repetition of common landscape features, such as tree trunks and geologic formations.

a. This landscape was found to have the following visual pattern:

_____
_____
_____
_____.

b. This landscape was found to have no evident visual patterns. Striking visual patterns have on influence on the visual quality of this landscape.

Rule 2. The extent to which a landscape contains striking visual pattern has a direct influence on the visual quality of the landscape. A pattern which immediately catches and holds the attention of the viewer is very striking. The more striking a pattern is the higher the visual quality.

a. The pattern observed in this landscape was found to be visually striking and immediately catches and holds the attention of the viewer, The visual pattern dominates the visual composition. The pattern causes the landscape to have high visual quality.

b. The pattern observed in this landscape was found to be moderately visually striking. It is apparent, but does not dominate the visual composition. The visual pattern attracts the viewer's attention, but the composition is complex enough that the viewer's attention is not dominated by the visual pattern. The visual pattern has a slight positive influence on visual quality.

c. The pattern observed in this landscape was found not to be striking or eye catching. The viewer's attention would not be attracted to the pattern. and it has no effect on visual quality.

Rule 3. Visual complexity is directly related to visual quality in natural landscapes. The more visually complex the landscape the greater the visual quality.

a. This landscape was judged as having a high degree of visual complexity. There is large number of different visual elements in the landscape or high degree of variation in the lines, forms, colors and textures which comprise the visual landscape. The landscape is very intricate or visually rich.

b. This landscape was judged as having moderately high visual complexity. There is a moderately large number of different visual elements in the landscape or moderately high degree of variation in the lines, forms, colors and textures which

comprise the visual landscape. The landscape has a moderate degree of intricateness or visual richness.

c. This landscape was judged as having moderate visual complexity. There is a moderate number of different visual elements in the landscape, or a moderate degree of variation in the lines, forms, colors and textures which comprise the visual landscape. The landscape has a moderate degree of intricateness or visual richness.

d. This landscape was judged as having moderately low visual complexity There is a moderately low number of different visual elements in the landscape, or moderately low degree of variation in the lines, forms, colors and textures which comprise the visual landscape. The landscape has a moderately low degree of intricateness or visual richness.

e. This landscape was judged as having low visual complexity. There is a low number of different visual elements in the landscape or a low degree of variation in the lines, forms, colors and textures which comprise the visual landscape. The landscape has a low degree of intricateness or visual richness.

Rule 4. Vividness is directly related to visual quality of the natural landscapes. The more vivid a landscape the greater the visual quality.

a. This landscape was judged to be highly vivid. The individual landscape features and/or patterns within the landscape are very visually distinct and readily visible. This landscape is very distinctive and memorable.

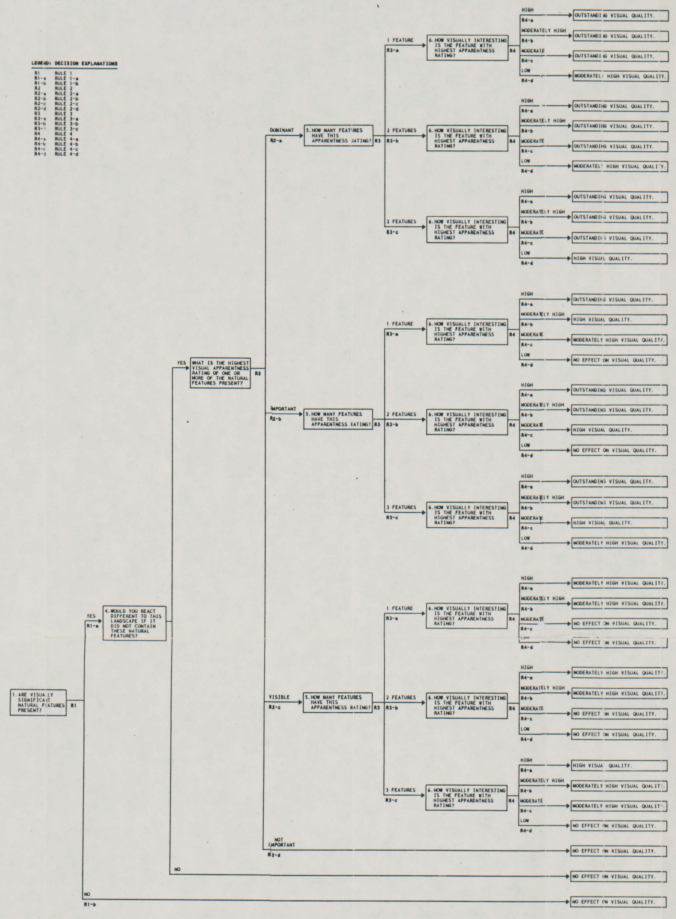
b. This landscape was judged to have moderately high vividness. The individual landscape features and/or patterns within the landscape have a moderately high degree of visual distinctiveness and visibility. This landscape is moderately high in terms of distinctiveness and memorability.

c. This landscape was judged to have moderate vividness. The individual landscape features and/or patterns within the landscape have a moderate degree of visual distinctiveness and visibility. This landscape is moderate in terms of distinctiveness and memorability.

d. This landscape was judged to have moderately low vividness. The individual landscape features and/or patterns within the landscape have a moderately low degree of visual distinctiveness and visibility. This landscape is moderately low in terms of distinctiveness and memorability.

e. This landscape was judged to have low vividness. The individual landscape features and/or patterns within the landscape have a low degree of visual distinctiveness and visibility. This landscape is low in terms of distinctiveness and memorability.

The vita has been removed from the scanned document

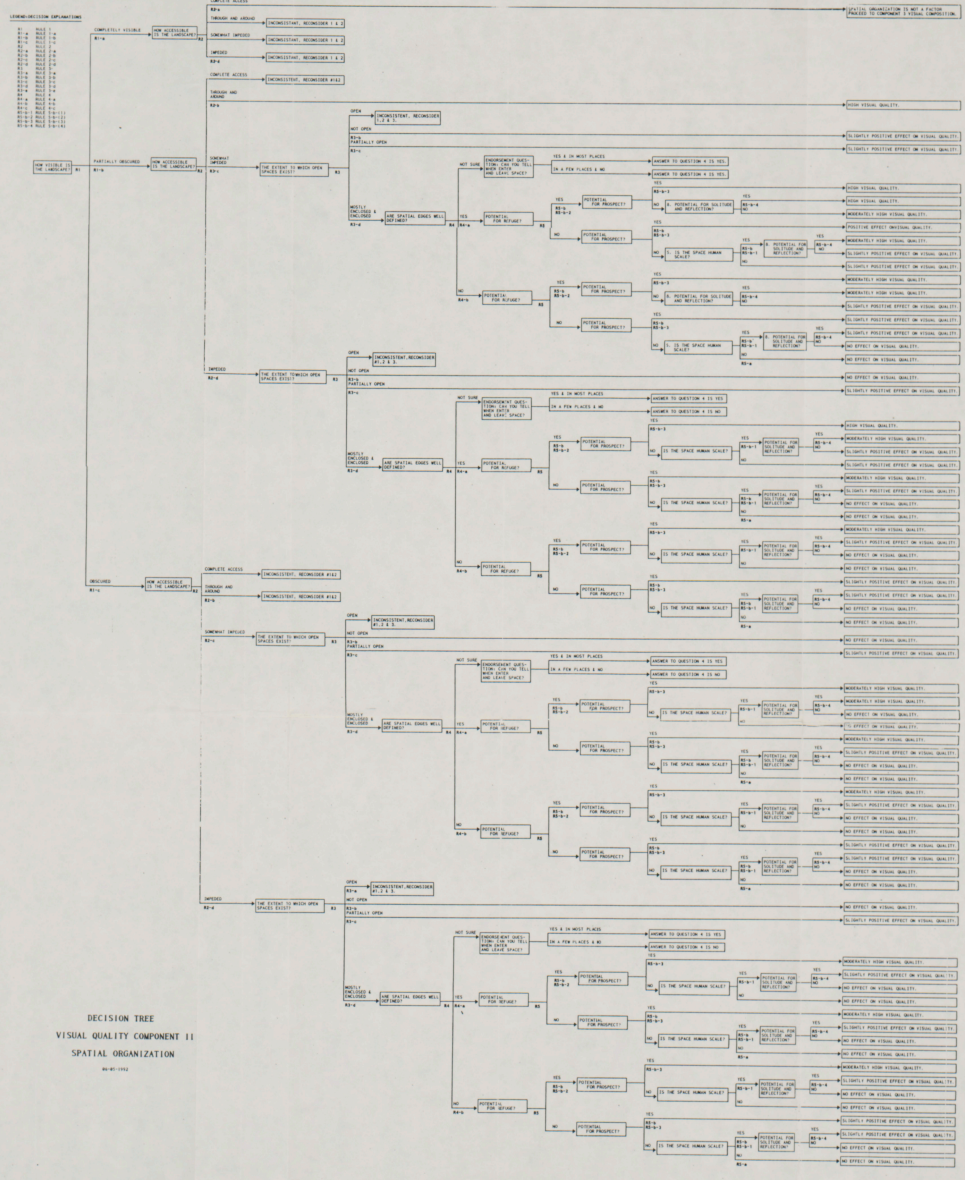


DECISION TREE

VISUAL QUALITY COMPONENT 1-B

NATURAL FEATURES

06-04-1982

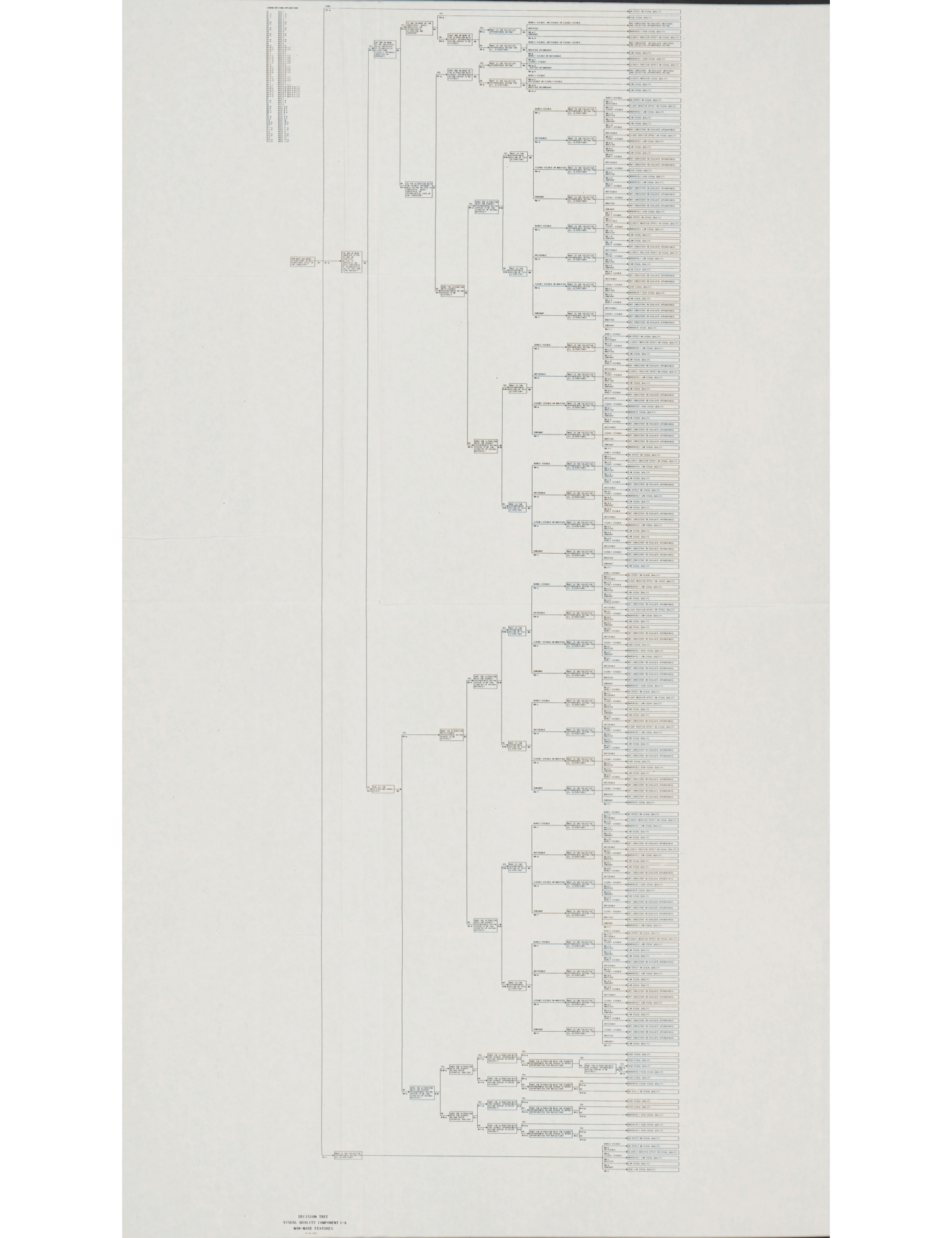


DECISION TREE

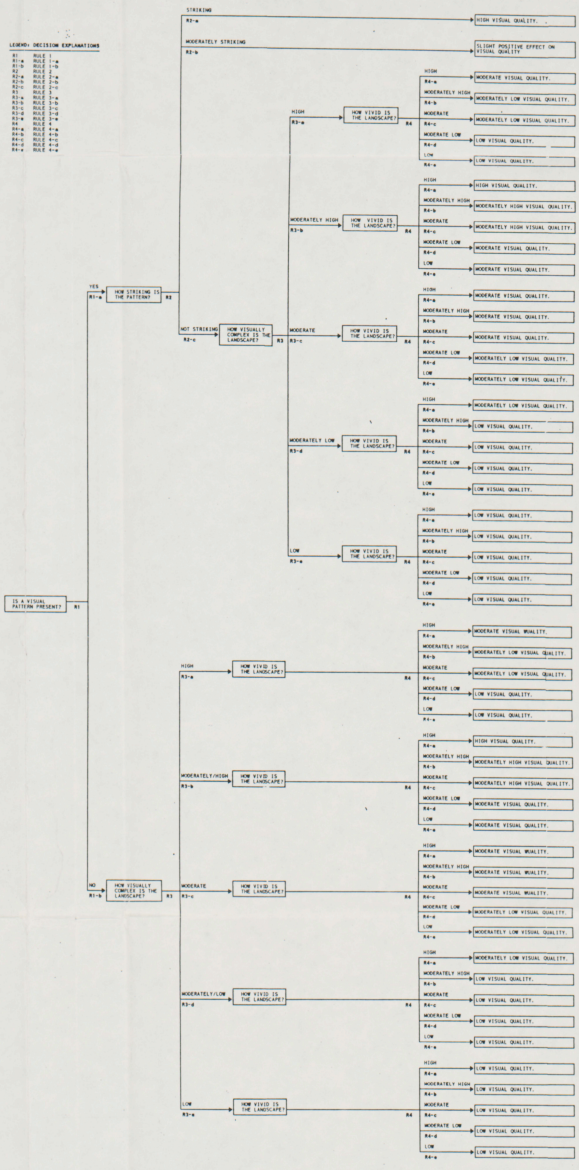
VISUAL QUALITY COMPONENT II

SPATIAL ORGANIZATION

06/07/1992



DECISION TREE
VISUAL QUALITY COMPONENT 1-A
MAN-MADE FEATURES



DECISION TREE

VISUAL QUALITY COMPONENT III

VISUAL COMPOSITION

06-04-1992