INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI
University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700  800/521-0600
A decision support system for highway embankment design using FGD by-products

Kim, Sung Hwan, Ph.D.
The Ohio State University, 1994
A DECISION SUPPORT SYSTEM FOR HIGHWAY EMBANKMENT DESIGN
USING FGD BY-PRODUCTS

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the Graduate
School of The Ohio State University

by

SUNG HWAN KIM, B.S. and M.S.

* * * * * *
The Ohio State University
1994

Dissertation Committee:
William E. Wolfe
Tien H. Wu
Fabian C. Hadipriono

Approved by
Adviser
Department of Civil Engineering
DEDICATION

To my mother, my wife and my son
ACKNOWLEDGMENTS

The author wishes to express his sincere gratitude to Dr. William E. Wolfe, his advisor, for his guidance and continuous encouragement throughout the course of this study. His patient and careful evaluation of the manuscript was of great assistance in the preparation of this dissertation. Thanks go to other members of his advisory committee, Dr. Fabian C. Hadipriono and Dr. Tien H. Wu for their valuable advice and review of the manuscript.

The author also wishes to thank Korea Highway Corporation and the sponsors of the project "Land Application Uses for Dry FGD By-products" RF768342. Deep appreciation is expressed for these sources of financial support. The author expresses his sincere gratitude to Mr. Eugene Geiger, Barry Thompson, and J. J. Yee from ODOT, Pedro Amaya from AEP, Sal Nodjomian from the US Air Force, Chien Tan Chang from Federal Highway Administration and Dr. Shin Tower Wang from ENSOFT, Inc. who participated in the validation process so necessary to the successful development of the work described in this dissertation.
# VITA

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
</table>
| February 15, 1958 | Born,  
Kyungbook, Korea                                                        |
| 1984       | Bachelor of Science,  
Seoul National University,  
Seoul, Korea                                                                 |
| 1983 - present | Section Chief  
Highway Research Center  
Korea Highway Corporation  
Seoul, Korea                                                                 |
| 1986 - 1988 | Master of Science,  
Department of Civil Engineering  
The Ohio State University  
Columbus, Ohio                                                                 |
| 1990 - 1994 | Research Associate  
Department of Civil Engineering  
The Ohio State University  
Columbus, Ohio                                                                 |

# PUBLICATIONS


FIELDS OF STUDY

Major Field: Civil Engineering

Study in geotechnical engineering

Minor Field: Mathematics
# TABLE OF CONTENTS

DEDICATION ........................................ ii
ACKNOWLEDGMENTS .................................. iii
VITA .............................................. iv
LIST OF TABLES ................................... ix
LIST OF FIGURES .................................. x

CHAPTER

I. INTRODUCTION ................................ 1
   1.1 Background ................................ 1
   1.2 Objectives ................................ 6
   1.3 Scope and Limitations ..................... 7

II. EMBANKMENT DESIGN USING FGD BY-PRODUCTS .... 10
   2.1 General ................................... 10
   2.2 Operating Conditions in the Power Plant .... 10
      2.2.1 FGD Systems ......................... 11
         2.2.1.1 Dry Scrubbing System ............. 11
            2.2.1.1.1 Sodium-Based Process ........ 11
            2.2.1.1.2 Calcium-Based Process ....... 13
         2.2.1.2 Wet Scrubbing System ............. 20
   2.3 FGD Waste Characterization ............... 23
      2.3.1 Leachate Properties .................. 24
      2.3.2 Chemical Properties .................. 25
      2.3.3 Engineering Properties ............... 28
   2.4 Embankment Design ......................... 31
      2.4.1 Standard ODOT Design Procedures ...... 33
      2.4.2 Engineered Design Procedures ...... ... 34
   2.5 Decision Support System in Civil Engineering 38
      2.5.1 Decision Support System ............. 38
      2.5.2 Expert System ...................... 40
      2.5.3 Comparison of DSS and ES .......... ... 42
<table>
<thead>
<tr>
<th>III. KNOWLEDGE ACQUISITION AND KNOWLEDGE REPRESENTATION</th>
<th>46</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 General</td>
<td>46</td>
</tr>
<tr>
<td>Knowledge Acquisition Process</td>
<td>49</td>
</tr>
<tr>
<td>3.2.1 Unstructured Knowledge Acquisition</td>
<td>49</td>
</tr>
<tr>
<td>3.2.2 Structured Knowledge Acquisition</td>
<td>50</td>
</tr>
<tr>
<td>3.2.3 Organized Phase of Knowledge Acquisition</td>
<td>51</td>
</tr>
<tr>
<td>3.3 Knowledge Representation</td>
<td>54</td>
</tr>
<tr>
<td>3.4 Chapter Summary</td>
<td>57</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IV. DEVELOPMENT OF INTELLIGENT DECISION SUPPORT SYSTEM</th>
<th>59</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 General</td>
<td>59</td>
</tr>
<tr>
<td>4.2 A Proposed Model</td>
<td>59</td>
</tr>
<tr>
<td>4.3 Knowledge Base</td>
<td>63</td>
</tr>
<tr>
<td>4.3.1 Objects</td>
<td>63</td>
</tr>
<tr>
<td>4.3.2 Rules</td>
<td>65</td>
</tr>
<tr>
<td>4.4 Inference Engine</td>
<td>66</td>
</tr>
<tr>
<td>4.5 User Interface</td>
<td>73</td>
</tr>
<tr>
<td>4.5.1 Screen Objects</td>
<td>74</td>
</tr>
<tr>
<td>4.5.2 Text</td>
<td>74</td>
</tr>
<tr>
<td>4.5.3 Graphics</td>
<td>75</td>
</tr>
<tr>
<td>4.6 External Files</td>
<td>76</td>
</tr>
<tr>
<td>4.7 Chapter Summary</td>
<td>76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>V. SUBSYSTEMS</th>
<th>78</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 General</td>
<td>78</td>
</tr>
<tr>
<td>5.2 Data Base Subsystem</td>
<td>78</td>
</tr>
<tr>
<td>5.2.1 Data Base of FGD By-products and Underlying Soils</td>
<td>79</td>
</tr>
<tr>
<td>5.2.1.1 Data Base for FGD By-products</td>
<td>80</td>
</tr>
<tr>
<td>5.2.1.2 Data Base for Underlying Soils</td>
<td>84</td>
</tr>
<tr>
<td>5.2.2 Data Base Management System</td>
<td>85</td>
</tr>
<tr>
<td>5.2.2.1 Relational Data Model</td>
<td>86</td>
</tr>
<tr>
<td>5.2.2.2 Data Base Inference Engine</td>
<td>91</td>
</tr>
<tr>
<td>5.2.2.3 Data Analysis in Data Base</td>
<td>92</td>
</tr>
<tr>
<td>5.3 Numerical Processing Subsystem</td>
<td>95</td>
</tr>
<tr>
<td>5.3.1 Slope Stability</td>
<td>95</td>
</tr>
<tr>
<td>5.3.2 Settlement</td>
<td>96</td>
</tr>
<tr>
<td>5.3.3 Integration of This Subsystem into the Shell</td>
<td>98</td>
</tr>
<tr>
<td>5.4 Graphical Design Subsystem</td>
<td>102</td>
</tr>
<tr>
<td>5.4.1 Interactive Graphics with Data Base Subsystem</td>
<td>102</td>
</tr>
<tr>
<td>5.4.2 Interactive Graphics with Numerical Programs</td>
<td>103</td>
</tr>
<tr>
<td>5.4.3 Interactive Design of Embankment</td>
<td>103</td>
</tr>
</tbody>
</table>
VI. TESTING, VALIDATION AND CASE STUDY . . . . 107

6.1 General . . . . . . . . . 107
6.2 Testing . . . . . . . . . 108
6.3 Validation By Experts . . . . . 112
6.4 Case Study of An Implementation of The System 116
   6.4.1 State Route 541 Background . . . . 116
   6.4.2 Soil Properties . . . . . . . 117
   6.4.3 Design of Reconstruction Embankment . 118

VII. DISCUSSION AND CONCLUSION . . . . 125

7.1 General . . . . . . . . . 125
7.2 Design of Fill Using FGD By-products . . 126
7.3 Knowledge Acquisition and Representation . 127
7.4 Development of IDSSHED . . . . . 128
7.5 Validation of IDSSHED . . . . . 131
7.6 Future Study . . . . . 132

LIST OF REFERENCES . . . . . . . . 134

APPENDICES

A. OPERATING MANUAL FOR IDSSHED . . . . 141
   A.1 Structure of IDSSHED . . . . 142
   A.1.1 Data Base . . . . 142
   A.1.2 Numerical Analyses . . . . 144
   A.1.3 Embankment Design Consultation . 145
   A.2 Installation . . . . . . . . 145

B. SOURCE CODES FOR IDSSHED . . . . . 149

C. DATA BASE FOR FGD BY-PRODUCTS AND SOILS . 287

D. FORTRAN PROGRAM FOR SETTLEMENT CALCULATION . 298

E. SOURCE CODES FOR GRAPHICAL DESIGN . . . 307
   E.1 Drawing for Leachate Properties . . . 308
   E.2 Drawing for Chemical Properties . . . 309
   E.3 Drawing for Engineering Properties . . 310
   E.4 Drawing of Failure Surfaces . . . . . 319
   E.5 Drawing of the Relationship of Time and Settlement . . . . . . 323
   E.6 Drawing of Embankment Shape . . . . 325

F. SCREENS OF CONSULTATION . . . . . . . . 330

G. EVALUATION RESULTS OF IDSSHED . . . . 355
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Allowable limits of leachate components</td>
<td>25</td>
</tr>
<tr>
<td>2.2 Total metal analysis obtained in both leachate and runoff</td>
<td>26</td>
</tr>
<tr>
<td>2.3 Guide for earth slope design</td>
<td>33</td>
</tr>
<tr>
<td>2.4 The difference between ES and DSS</td>
<td>45</td>
</tr>
<tr>
<td>5.1 An example of PCODE file</td>
<td>83</td>
</tr>
<tr>
<td>5.2 An example of MCODE file</td>
<td>83</td>
</tr>
<tr>
<td>5.3 An example of LEADB file</td>
<td>84</td>
</tr>
<tr>
<td>5.4 An example of soil data</td>
<td>85</td>
</tr>
<tr>
<td>6.1 General evaluation of IDSSHED by experts who were involved in the system development</td>
<td>114</td>
</tr>
<tr>
<td>6.2 General evaluation of IDSSHED by experts who were not involved in the system development</td>
<td>114</td>
</tr>
<tr>
<td>6.3 Properties of soils</td>
<td>117</td>
</tr>
<tr>
<td>6.4 Properties of Tidd FGD by-product</td>
<td>119</td>
</tr>
<tr>
<td>6.5 Validation result by field engineers</td>
<td>121</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figures</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Distribution of the production and use of coal combustion by-products in U.S.</td>
<td>3</td>
</tr>
<tr>
<td>2.1 A diagram of a typical dry process</td>
<td>12</td>
</tr>
<tr>
<td>2.2 A diagram of a typical wet process</td>
<td>21</td>
</tr>
<tr>
<td>2.3 A structure of DSS</td>
<td>43</td>
</tr>
<tr>
<td>2.4 A structure of ES</td>
<td>44</td>
</tr>
<tr>
<td>3.1 Knowledge acquisition process</td>
<td>53</td>
</tr>
<tr>
<td>3.2 Knowledge representation in IDSSHED</td>
<td>58</td>
</tr>
<tr>
<td>4.1 The proposed model of IDSSHED</td>
<td>62</td>
</tr>
<tr>
<td>4.2 Knowledge tree</td>
<td>68</td>
</tr>
<tr>
<td>4.3 An example of an object FGD_UCS</td>
<td>70</td>
</tr>
<tr>
<td>4.4 Class hierarchy in data base</td>
<td>71</td>
</tr>
<tr>
<td>4.5 Class heterarchy in IDSSHED</td>
<td>72</td>
</tr>
<tr>
<td>5.1 A structure of data base</td>
<td>80</td>
</tr>
<tr>
<td>5.2 The relational scheme with E-R diagram for FGD material evaluation process</td>
<td>88</td>
</tr>
<tr>
<td>5.3 Alternative symbols for E-R diagram</td>
<td>89</td>
</tr>
<tr>
<td>5.4 Hierarchical representation of SELECTION process in PCODE</td>
<td>90</td>
</tr>
<tr>
<td>5.5 Integration of IDSSHED and IDSSDB inferences</td>
<td>92</td>
</tr>
<tr>
<td>5.6 An example of a calculation in the data base</td>
<td>94</td>
</tr>
</tbody>
</table>
5.7 A schematic diagram of the communication between
the shell and the FORTRAN program . . . 100

5.8 An example of creation of sequential file
in the shell . . . . . . . . . . . . . 101

5.9 An example of assigning topics from a sequential
file created in FORTRAN program . . . 101

5.10 A schematic diagram of the communication between
the shell and the graphical design program . 105

6.1 Decision tree in IDSSHED . . . . . . 111

6.2 Validation tree . . . . . . . . . . . . . 115

6.3 Map of the construction site . . . . . . 122

6.4 Site condition at failure . . . . . . . . 123

6.5 Design of Embankment in case study . . . . 124

A.1 Program structure of IDSSHED . . . . . . 142

A.2 Structure of FGDDBD . . . . . . . . . . . . . 143

A.3 Structure of SOILDB . . . . . . . . . . . . . 144

A.4 Structure of STABLE_W . . . . . . . . . 144

A.5 Structure of SETTLE_W . . . . . . . . . . . . . 145

A.6 Structure of EMBKB . . . . . . . . . . . . . 146

A.7 New program project . . . . . . . . . . . . . 148

A.8 Program group properties . . . . . . . 148

A.9 Program item properties . . . . . . . 148

F.1 Introduction screen to IDSSHED . . . . . 331

F.2 Procedures of embankment design . . . . . 331

F.3 User's information screen . . . . . . . 332

F.4 Site information screen . . . . . . . 332

F.5 Selection of material type for fill . . . . . 333

F.6 Data base for material properties . . . . . 333

xi
F.7 Stress analysis methods  . . . . . 334
F.8 Loading conditions in embankment  . . . 334
F.9 Methods of slope stability analysis . . . 335
F.10 Profile for slope analysis  . . . . . 335
F.11 Material properties for slope analysis . . 336
F.12 Boundary conditions for slope analysis . . 336
F.13 Possible failure surfaces in soil fill . . 337
F.14 A critical failure surface in soil fill . . 337
F.15 Drainage types . . . . . . 338
F.16 Number of layers in embankment and foundation 338
F.17 Average stress history profile  . . . 339
F.18 Inputs for settlement calculation . . . 339
F.19 Relationship of time and settlement . . 340
F.20 Summary of consultation in path I . . . 340
F.21 Conclusion in Path I  . . . . . 341
F.22 Selection of FGD material from data base 341
F.23 Operating information of a material selected . 342
F.24 Leachate property of a material selected . 342
F.25 Chemical property of a material selected . 343
F.26 Engineering property of a material selected . 343
F.27 Selection of default values . . . . 344
F.28 Inputs of leachate properties . . . . 344
F.29 Inputs of chemical properties . . . . 345
F.30 Inputs of engineering properties . . . . 345
F.31 Intermediate conclusion . . . . . 346
F.32 Methods of slope stability analysis xii
(with FGD material) . . . . . . 346

F.33 Profile for slope analysis (with FGD material) 347

F.34 Material properties for slope analysis (with FGD material) . . . . . . 347

F.35 Boundary conditions for slope analysis (with FGD material) . . . . . . 348

F.36 Possible failure surfaces in FGD fill . . . . . . 348

F.37 A critical failure surface in FGD fill . . . . . . 349

F.38 Number of layers in embankment and foundation (with FGD material) . . . . . . 349

F.39 Inputs for settlement calculation (with FGD material) . . . . . . 350

F.40 Relationship of time and settlement (with FGD material) . . . . . . 350

F.41 Summary of numerical analysis . . . . . . 351

F.42 Conclusion in Path III . . . . . . 351

F.43 Factors of embankment design . . . . . . 352

F.44 Design of FGD-based embankment . . . . . . 352

F.45 Selection of side slope cover . . . . . . 353

F.46 Summary of design information . . . . . . 353

F.47 Construction information . . . . . . 354

F.48 Report of consultation results . . . . . . 354

xiii
CHAPTER I
INTRODUCTION

1.1 Background

The burning of coal in electric power plants results in the production of sulfur dioxide gas (SO$_2$), which is considered to be a primary cause of acid rain. As the 1970 Clean Air Act and its subsequent amendments were established, the reduction of power plant SO$_2$ emissions and solid waste disposal have become issues. To reduce the emission of SO$_2$ gas, sorbents such as lime or limestone are injected in the flue gas control systems. As a consequence of this SO$_2$ emissions control process, a solid waste material is generated. This solid waste is known as a flue gas desulfurization (FGD) by-product.

Various types of FGD systems have been in operation since 1967. Generally, two major types of SO$_2$ control systems are available today: dry and wet. Uses of the dry FGD by-product are discussed in the work presented here. The typical dry FGD by-product contains fly ash and unreacted lime as well as the sulfur reaction products. It is the free lime as well as to some extent the fly ash that has led researchers in the area to assume that the by-product might
possess desirable engineering properties. FGD waste has a relatively low unit weight and good shear strength characteristics which make it a desirable fill material, particularly over weak or compressible soils. However, the specific characteristics of the dry FGD by-product produced during stack gas cleaning operations vary with several factors including the sorbent used, where the sorbent is injected into the system, the sulfur content of the coal and the mechanics of the scrubbing process, particularly the handling and storage systems. Furthermore, the properties of FGD by-product can be modified during construction and, once placed, may be affected by environmental conditions. Therefore, the dry FGD by-product is not a single material, but a range of products displaying a high degree of variability in chemical, physical, and engineering properties.

Figure 1.1 shows the distribution of the production and use of the primary coal combustion by-products in the U.S. In 1991, 18.1 million tons of FGD waste were produced and over 98% of it was disposed in landfills [ACAA, 1992]. As the amount of these wastes will continue to increase, the economic and environmental considerations affecting disposal must receive more attention. It would be desirable to find large-scale uses for this waste product. Some of the uses for the waste being investigated include as a structural fill, cement replacement, aggregate production, agricultural
uses, mine reclamation and placement in road bases, subbases, and embankments [Rosenberg, 1992, Shepard, 1985, and Stein, 1991]. A logical and potentially promising area of the large-volume use of FGD by-product is the construction of highway embankments [Kim et al., 1993].

![Figure 1.1 Distribution of the production and use of coal combustion by-products in the U.S (data obtained from ACAA)](image-url)

Figure 1.1 Distribution of the production and use of coal combustion by-products in the U.S (data obtained from ACAA)
Geometric designs for modern highways often require that the roadways be constructed at elevations above existing ground. The embankment structure is the major component for an above ground highway. The function of an embankment is to support a pavement system above the natural ground. Therefore, the embankment has failed when it causes roughness or damage to the roadway. The failure may be massive, as in the case of slides resulting from instability of the embankment or the underlying foundation, or more subtle as when consolidation of the embankment or the underlying foundation materials produces failure by the gradual development of excessive differential settlements of the pavement surface, causing rutting, faultings or cracks. Thus, embankment performance is dependent upon the stability and deformation of both the embankment and the underlying foundation materials [NCHRP, 1971]. Failures in the embankment itself may be caused by poor fill materials, substandard construction methods, or inadequate quality control. Designing an embankment using an FGD by-product is a complex undertaking due to the range in the material properties of the FGD material itself, environmental limits that might be placed on using such materials, and the traditional geotechnical problems associated with characterization of embankment designs and existing ground conditions. The design process requires the geotechnical engineer to make numerous decisions to optimize the
selection of the material types with the embankment design. Traditionally, much of the design process is guided by the experience of experts in the field of geotechnical engineering. Recently, these experiences have begun to be collected and organized into formal rules permitting the knowledge of recognized experts to be utilized by practicing geotechnical engineers in a systematic manner [Adams, 1990; Baker et al., 1989; Kim et al., 1992a; Maher, 1991]. However, the lack of experts in the area of the embankment design using FGD by-products makes this domain attractive for the development of a knowledge-based system. It is the goal of this research to develop rules and procedures that would assist geotechnical engineers in the design of earth structures in which FGD by-products can be used as replacements for soil with a high degree of confidence.

An intelligent decision support system for highway embankment design (IDSSHED) is proposed as a solution to this problem. This system integrates the better components of both a decision support system and an expert system. It contains the six essential components selected from both systems i.e. a knowledge base, an inference engine, a database, an analytic model, a design model, and a user interface.
1.2 Objective

The objective of this study is to introduce a knowledge-based decision support system to assist the project engineer in designing a reliable embankment using FGD by-products. This system will 1) evaluate an FGD by-product which the designer has available for use in a specific highway embankment application, 2) perform the required design calculations, 3) provide detailed design information to the user and then 4) encourage the utilization of FGD by-product by providing the technical information necessary to incorporate FGD materials into highway applications.

To achieve this objective, we propose to develop an Intelligent Decision Support System for Highway Embankment Design (IDSSHED). This system is a user-friendly computer program which encodes expert experiences, heuristic judgments, and calculation results obtained from analytical programs. The core of IDSSHED is a knowledge base which identifies likely material properties and determines design requirements. This system is an interactive tool able to manage the dynamic data exchange with external programs. In the interface with the data base it is capable of evaluating the FGD material properties and appending or retrieving the records of the material properties for FGD by-products and any underlying soils. The values obtained from the data base will be transferred to numerical processing programs for
slope stability and settlement calculations. This system also serves as an educational tool to introduce the characteristics of FGD by-products and the procedures of embankment design to the designers.

1.3 Scope and Limitations

There is a variety of material types used in geotechnical structures including natural soil, steel, concrete, geosynthetics, and industrial production by-products. A significant by-product of industrial production is generated during the combustion of coal. Coal combustion by-products are actually several different materials including, fly ash, bottom ash, slag, and FGD wastes. The FGD by-product is the result of attempts to reduce air pollution caused by stack gasses. Today these FGD by-products come from two sources: a high water content sludge produced by wet processes and a soil-like waste produced by dry processes. This research investigates the potential of using the dry FGD waste in high volume earth structures such as highway embankments.

For this application it is necessary to characterize the FGD material, to analyze the stability and settlement of the structure to be made from the FGD materials, and to determine a satisfactory design of the embankment based on the previous outputs. In this research the author proposes to develop a knowledge-based decision support system which
can integrate heuristic and algorithmic techniques to increase designer's confidence in a practical model. In this system, a data base provides the designer with the material properties of FGD by-products and underlying soils. The system combines several numerical processing programs to analyze the stability of the embankment slopes and settlement of the embankment. The large volumes of outputs are integrated into interactive graphical design program.

In the following chapter, we review the literature relating to FGD waste utilization, highway embankment design, and intelligent decision support systems as applied to geotechnical engineering. Chapter 3 provides the knowledge acquisition and knowledge representation for the system. Knowledge acquisition models and knowledge representation methods are discussed. Chapter 4 contains the development of an intelligent decision support system. The knowledge base and the inference engine of the system as well as the system shell are discussed. Chapter 5 discusses the subsystems included in the decision support system. A numerical analysis subsystem for slope stability and settlement, a data base subsystem for the material properties of FGD by-products and foundation soils, and an interactive graphics subsystem are presented. Chapter 6 presents the testing and validation of IDSSHED. A case study is also shown and its results are discussed. In Chapter 7, the study is summarized and discussion and
conclusions drawn from this research are presented. A user’s manual for IDSSHED is attached in Appendix A. Source programs are in Appendix B. The data base for FGD by-products and soils is in Appendix C. The source programs for settlement calculations and graphical design are in Appendix D and Appendix E, respectively. Screens of the consultation for case study are presented in Appendix F. Evaluation results of IDSSHED conducted by experts are included in Appendix G.
CHAPTER II
EMBANKMENT DESIGN USING FGD BY-PRODUCTS

2.1 General

The goal of the research presented in this thesis is the development of a comprehensive decision support system to evaluate the potential for using Flue Gas Desulfurization wastes in highway embankments and then to recommend a suitable embankment design. Reviews of what the author considers the three primary aspects of the problem i.e. characterization of the properties of the FGD waste, highway embankment design, and decision support systems in geotechnical engineering will be reviewed in the following sections.

2.2 Operating Conditions in the Power Plant

As the 1970 Clean Air Act and its subsequent amendments were established, reducing emissions of SO$_2$ became an important part of power plant operations. A number of advanced SO$_2$ control processes have been developed to address increased concerns over the impact on the environment of SO$_2$ emissions.
The major factors of the operating conditions in the advanced SO₂ control processes affecting the characteristics of FGD wastes can be summarized as follows [Vuceta, 1980]: flue gas characteristics (coal ash content, coal sulfur content, coal heating value, and boiler type), FGD technology applied (FGD systems and processes), sorbent characteristics (sorbent type, stoichiometric ratio, and sorbent utilization), and the type of solid collection apparatus. In a brief overview of the different systems, major processes with an emphasis on the dry FGD system are discussed.

2.2.1 FGD Systems

The FGD systems used to remove SO₂ from power plant emissions are classified as either dry or wet.

2.2.1.1 Dry Scrubbing System

Dry systems were developed as a way to eliminate the inherent problems of the established wet-scrubbing system. The basic dry system consists of a powdered sorbent that is fed into gas stream and reacts with sulfur dioxide (SO₂) in the flue gas. A generalized flow diagram of this process is shown in Figure 2.1. The injection point of the sorbent can be varied from the boiler furnace area all the way to the flue gas entrance to the electrostatic precipitator (ESP) or bag collector. The by-product, which
Figure 2.1 A diagram of a typical dry process (after Goodwin, 1983)
contains mostly fly ash and sulfur compounds, is removed as a dry waste with normal bag cleaning. Studies have shown that dry scrubbing is capable of removing from 30% to 95% of the SO$_2$ in the gas stream, making it most suitable for low to medium sulfur coals [Midkiff, 1984; EPRI, 1988a].

With regard to waste disposal, dry FGD systems have an inherent advantage over wet systems in that they produce a dry, solid waste product that can be handled by conventional fly ash handling systems, eliminating requirements for a sludge handling system. Dry systems, therefore, do offer potential advantages over wet systems, especially in the areas of energy savings and handling costs.

There are several dry FGD processes typically classified by types of sorbent used and the location of the sorbent injection. The FGD processes can be distinguished by the following [Vuceta, 1980]:

(1) the type of sorbent used: most common sorbents are calcium-based alkali, sodium-based alkali, calcium-magnesium alkali (dolomite), and ammonia;

(2) the location between the burners and the particulate control device where the sorbent is injected. There are eight basic types of dry SO$_2$ processes that were studied in conjunction with this research; These include lime injection multistage burner (LIMB), spray dryer (SPD),
duct injection (DUCT), Coolside, and fluidized bed combustion (FBC), SOx-NOx-ROx Box (SNRB), lime injection dry scrubbing (LIDS), rotary cascading bed boiler (RCBB).

2.2.1.1 Sodium-Based Process

Sodium-based sorbents are nahcolite, sodium bicarbonate, soda ash, trona, and fly ash. Many sodium-based processes used nahcolite as a sorbent. When added to a hot flue gas, the raw nahcolite (NaHCO₃) decomposes to Na₂CO₃ and then reacts with SO₂ and O₂ to form Na₂SO₃ and Na₂SO₄ solids. Sodium bicarbonate, soda ash and trona also react similarly to nahcolite in that Na₂CO₃ is first formed, then the Na₂SO₃ and Na₂SO₄ salts. Fly ash may remove some SO₂ due to the reaction of sodium salts in the fly ash. A typical reaction of the sorbent to SO₂ gas is shown in the following:

$$2\text{Na}_2\text{HCO}_3 + \text{SO}_2 + \frac{1}{2}\text{O}_2 \Rightarrow \text{Na}_2\text{SO}_4 + \text{H}_2\text{O} + 2\text{CO}_2$$

Wastes contain fly ash, unreacted sorbent, and sorbent-reacted products which are predominantly sodium sulfate (Na₂SO₄). These solid wastes are highly water-soluble and can lead to leachability and waste-stability problems.
2.2.1.1.2 Calcium-Based Process

The most common calcium-based sorbents are lime and limestone. Sorbents react with $\text{SO}_2$ and $\text{O}_2$ to form $\text{CaSO}_3$ and $\text{CaSO}_4$ solids. In the case of using limestone as a sorbent, the limestone, $\text{CaCO}_3$, calcines:

$$\text{CaCO}_3 \Rightarrow \text{CaO} + \text{CO}_2$$

and the resulting quick lime, $\text{CaO}$, subsequently reacts with $\text{SO}_2$ then forms $\text{CaSO}_3$ or $\text{CaSO}_4$:

$$\text{CaO} + \text{SO}_2 + \frac{1}{2}\text{O}_2 \Rightarrow \text{CaSO}_4$$

The relative ratio of $\text{CaSO}_3/\text{CaSO}_4$ depends upon oxygen availability. Lime is inherently more effective as a scrubber reagent than limestone.

Wastes contain fly ash, unreacted sorbent, and sorbent reaction products which are primarily calcium sulfite ($\text{CaSO}_3$) and calcium sulfate ($\text{CaSO}_4$). The following subsections describe the processes using calcium-based sorbents

Limestone Injection Multistage Burner (LIMB) LIMB is a retrofit technology in which a calcium-based sorbent is injected into the boiler for direct capture of $\text{SO}_2$ in the flue gas [Nolan, 1988a and 1988b]. It may be combined with the use of low $\text{NO}_x$ burners for reduction of $\text{NO}_x$. 

emissions. In this process, the sorbent (hydrated lime) is calcined to active calcium oxide, and then reacts with sulfur dioxide and oxygen in the flue. The reacted lime remains suspended in the flue gas stream until it is collected just prior to the outlet stack in ESP. Sorbent utilization efficiency in this process is low. Therefore, the by-product contains up to 30% unreacted lime (CaO) [Nolan, 1988a].

A full-scale demonstration of a LIMB process was begun at the 104 MW Unit 4 boiler at Ohio Edison's Edgewater Station in Lorain, Ohio in 1986.

**Spray Dryer** The spray dryer process introduces the sorbent in the form of a slurry. Fine droplets of the reagent slurry come in close contact with the flue gas in the spray dryer chamber and combine with sulfur dioxide to form calcium sulfite. Some of the lime and sulfur dioxide also combine with oxygen and form calcium sulfate. In the spray dryer chamber, the water absorbed in the slurry evaporates. Some of the dry waste produced through the processing drops out of the bottom of the spray dryer, but most of the waste is collected downstream of the spray dryer in either a baghouse or an ESP. \( \text{SO}_2 \) removal rates of 65-95% can be achieved with spray dryer absorption [Donnelly, 1985].
Fluidized Bed Combustion (FBC)  The two desulfurization processes described in the previous sections are utilized primarily as retrofits to existing boilers. FBC uses a completely new furnace/boiler system. FBC desulfurization is a process of burning crushed coal in a bed of sorbent particles held in suspension by the upward flow of combustion air entering a chamber through a perforated air distribution grid below the fluidized bed. During operation, coal and sorbent are fed continuously into the bed. The sorbent reacts with the SO$_2$ released from the burning coal and prevents the SO$_2$ from escaping into the atmosphere.

There are three types of FBC boilers normally discussed. These are atmospheric FBC (AFBC), circulating FBC (CFBC), and pressurized FBC (PFBC) [Manaker, 1992]. Both of the AFBC and CFBC processes operate at atmospheric pressure. FBC units can burn low-grade fuels under stringent emissions standards. In the U.S. large number of independent power producers have been using AFBC and CFBC technology.

PFBC technology has lagged AFBC and CFBC technology. At this time, only demonstration scale plants are underway, a few in Europe and one in the U.S. The U.S. PFBC demonstration unit is operated by AEP at their Tidd plant in Brilliant Ohio. The PFBC process involves burning crushed coal under high pressure in a sorbent bed of
either dolomite or limestone [McClung, 1990].

**Other Processes** Other demonstration scale processes to remove SO\(_2\) gas include Coolside, duct injection, SOx-NOx-NOx Box (SNRB), lime injection dry scrubbing (LIDS), rotary cascading bed boiler (RCBB) processes.

The Coolside process [Yoon, 1988; Wu, 1991] is an SO\(_2\) control technology involving dry sorbent injection followed by flue gas humidification through evaporation of water sprays. This technology is particularly well-suited for retrofit applications in coal burning power stations that lack adequate space for other FGD installations. A demonstration test (10 MW scale) at the Ohio Edison Edgewater Station under the DOE clean coal technology program began in 1988.

Duct Injection process [PETC, 1990] controls the SO\(_2\) emission using a calcium-based reagent. It is suitable for the retrofit application at older coal-fired power plants. This process can remove 50 - 60% of the SO\(_2\) in the stack. The duct injection technology has been demonstrated at a 12 MW unit at Ohio Power Company Muskingum River in Beverly, Ohio. In this power plant three types of dry sorbent injection tests were performed.

LIDS process [Amrhein, 1992] combines the technologies of furnace limestone injection and dry scrubbing to achieve high levels of sulfur dioxide removal. The process begins by injecting pulverized
limestone into the upper furnace where it calcines and
reacts with the SO$_2$ in the flue gas. A pilot-scale
demonstration (1 MW scale) of the LIDS process sponsored
by the Ohio Coal Development Office (OCDO) and Babcock and
Wilcox (B&W) was installed at the B&W research center in
Alliance, Ohio.

The SNRB process [Holmes et al., 1992] is a combined
SOx, NOx, and particulate removal air pollution control
technology in which all three pollutants are removed from
the flue gas stream in a high-temperature, pulse-jet
baghouse. The SNRB technology incorporates dry injection
of a sorbent such as hydrated lime or sodium bicarbonate
for SOx emission reduction, ammonia injection with
selective catalytic reduction for NOx removal, and a high-
temperature baghouse for particulate control. The SNRB
technology is being demonstrated at a 5 MW unit at Ohio
Edison's R.E. Burger plant near Shadyside, Ohio.

RCBB is designed to burn diverse fuels and waste
ranging from high-sulfur coal to municipal solid wastes in
an economical and environmentally acceptable manner at
capacity levels suitable for industrial, institutional,
and commercial use. This technology is being demonstrated
at the Pedco plant operated by North American Rayon
Corporation in Elizabethton, Tennessee.
2.2.1.2 Wet Scrubbing System

Wet scrubbing systems are the most prevalent flue gas desulfurization systems in use today, accounting for over 80% of the coal-fired generating capacity. In wet scrubbing technology, removal efficiencies can approach 90% by carefully balancing the many chemical-reaction parameters involved [Midkiff, 1984].

In the operating process as shown in Figure 2.2, a solid sorbent is pulverized and mixed with water to form a scrubbing liquor, which contacts the flue gas containing SO₂ in an absorption tower. In this tower, the SO₂ is converted to an insoluble sludge product. The resulting slurry is removed from the system, treated and disposed in the field. In the case of a lime sorbent, the following chemical reactions occur. The by-products include gypsum (CaSO₄·2H₂O), calcium sulfite hemihydrate (CaSO₃·1/2H₂O) and calcium carbonate (CaCO₃).

CaO + SO₂ + H₂O \Rightarrow x\text{CaSO}_3\cdot\frac{1}{2}\text{H}_2\text{O} + y\text{CaSO}_4\cdot2\text{H}_2\text{O} + z\text{Ca(OH)}_2

Ca(OH)$_2$ + SO$_2$ \Rightarrow x'\text{CaSO}_3\cdot\frac{1}{2}\text{H}_2\text{O} + y'\text{CaSO}_4\cdot2\text{H}_2\text{O} + z'\text{CaCO}_3

Hagerty et al. [1977] reported that a typical chemical composition of FGD sludge created by a lime scrubbing operation was 21% CaSO₄·1/3H₂O, 12% CaSO₄·2H₂O, 39% unreacted sorbent,
Figure 2.2 A diagram of a typical wet process
[after Goodwin, 1983]
Typical components of sludge leachate are arsenic 0.0139 ppm, chromium 0.8 ppm, cadmium 0.07 ppm, lead 0.3 ppm, mercury 0.00025 ppm, selenium 0.09 ppm, and silver 0.01 ppm [Goodwin, 1983].

Most waste produced in this system is in slurry form, containing only 10% - 15% solids. The major concern in the operation of wet FGD systems is the handling and disposal of a large volume of liquid waste products. Therefore, the first step in disposing of these sludges is to stabilize them if they are to be placed in landfills. Several methods (forced oxidation, fly ash blending, and lime fixation) [Goodwin, 1983] have been applied to stabilize the wet FGD sludge, yielding a transportable material. Since these stabilized materials exhibit soil-like characteristics, adequate engineering properties can often be attained by proper field compaction. The characteristics of FGD sludge are summarized as follows [Goodwin, 1983; Hagerty and Ullrich, 1977]:

1. particles of sludge are very uniform in size and morphology,
2. compacted unit weight varies from 1.2 g/cm$^3$ to 1.69 g/cm$^3$,
3. shear strength exhibited falls within the range of inorganic silts and fine sands ($c' = 0$ psf and $\phi' = 25^\circ - 33^\circ$), and
4. the coefficient of permeability ranged from $10^{-4}$ to
10^-6 cm/sec.

In addition to the handling and disposal problems of the sludge, utilities have experienced numerous problems in operating the massive scrubbers, many of which are related to the use of a liquid slurry medium [Shepard, 1985]. A major problem area is the slurry nozzle which often plugs. Other problems are common in equipment which is prone to corrosion.

2.3 FGD Waste Characterization

The by-products of the dry processes are typically self hardening because the processes result in a material having a considerable amount of unreacted lime. They should possess desirable engineering properties as either a soil amendment or as a soil replacement. It is the development of design procedures using the dry FGD that is the subject of this thesis. As was described earlier, the dry waste product is an easier material to landfill. However, Goodwin [1983] and Stein [1991] reported that the dry processes generate a greater volume of by-product than a typical wet FGD scrubber. Therefore, the dry processes require larger areas for disposal. Most by-products are now being placed in landfills. Due to the limited area and high costs for new landfill development, it is desirable to find alternative uses for the waste products. For proper utilization of large quantities of
FGD wastes, chemical and geotechnical evaluations of these by-products are required.

2.3.1 Leachate Properties

A primary concern with the use of FGD materials in any construction activity is the potential for leaching hazardous chemical components, and the consequent pollution of ground and surface water near FGD by-product structures. The US Environmental Protection Agency (USEPA) has identified the concentrations of certain heavy metals that, if present in sufficient quantities in the leachate, would result in the FGD by-products being classified as hazardous according to the Resource Conservation and Recovery Act (RCRA). In that act, a waste is categorized as hazardous when the extract obtained by the extraction procedure (EP) test has concentrations of any of the regulated metals in excess of 100 times Primary Drinking Water Standards. Table 2.1 lists these drinking water maximum values.

In the Phase I report published by OSU [Bigham et al., 1993], the measured concentrations of heavy metals in the leachate of FGD by-products is much lower than the limit and the material would be classified as nonhazardous according to the characteristics of EP toxicity. Wolfe and Beeghly [1993] constructed a small truck ramp on the OSU campus using a dry FGD waste. At this ramp site samples
of water were obtained from both the runoff water and the leachate. They reported that measured levels of metals as shown in Table 2.2 in both leachate and runoff have always been significantly below the concentration accepted by the EPA limit for the leachate. In most of the engineering applications, the material would be well compacted, have a low permeability, and have a low potential for leachate generation. As a result, leaching characteristics of the dry FGD by-products should not be a problem for most types of utilization. The most likely problem would be the potential for leaching sodium and alkaline species in large-scale utilizations, such as structural fill or roadbase construction. Understanding the leaching potential is critical to the designer since methods for minimizing or eliminating leachate would then have to be employed in the design.

2.3.2 Chemical Properties

Chemical properties of FGD wastes directly affect the quality of leachate and long term strength development [EPRI, 1988a]. The properties are similar to those of conventional fly ash since fly ash makes up a major fraction of the product [EPRI, 1988d and 1989]. The difference in properties when compared to fly ash can be attributed to the sorbent products in the waste [EPRI, 1988a]. Bigham et al. [1993] reported that a typical dry
waste consists of 52% fly ash, 18% CaSO\(_4\), 13% CaCO\(_3\), 10% Ca(OH)\(_2\), and 2% others.

Table 2.1 Allowable limits of leachate components
(unit: mg/L)

<table>
<thead>
<tr>
<th>Compound</th>
<th>EPA Limit for Drinking Water</th>
<th>Leachate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arsenic</td>
<td>0.05</td>
<td>1.5</td>
</tr>
<tr>
<td>Barium</td>
<td>1.00</td>
<td>30.0</td>
</tr>
<tr>
<td>Cadmium</td>
<td>0.01</td>
<td>0.3</td>
</tr>
<tr>
<td>Chromium</td>
<td>0.05</td>
<td>1.5</td>
</tr>
<tr>
<td>Lead</td>
<td>0.05</td>
<td>1.5</td>
</tr>
<tr>
<td>Mercury</td>
<td>0.002</td>
<td>0.06</td>
</tr>
<tr>
<td>Selenium</td>
<td>0.01</td>
<td>0.3</td>
</tr>
<tr>
<td>Silver</td>
<td>0.05</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Table 2.2 Total metal analysis obtained in both leachate and runoff [Wolfe and Beeghly, 1992]

<table>
<thead>
<tr>
<th>Compound</th>
<th>Surface Runoff</th>
<th>Leachate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arsenic</td>
<td>0.073</td>
<td>0.044</td>
</tr>
<tr>
<td>Barium</td>
<td>&lt; 0.2</td>
<td>&lt; 0.2</td>
</tr>
<tr>
<td>Cadmium</td>
<td>0.04</td>
<td>0.04</td>
</tr>
<tr>
<td>Chromium</td>
<td>&lt; 0.06</td>
<td>&lt; 0.06</td>
</tr>
<tr>
<td>Lead</td>
<td>0.31</td>
<td>0.49</td>
</tr>
<tr>
<td>Mercury</td>
<td>&lt; 0.001</td>
<td>0.004</td>
</tr>
<tr>
<td>Selenium</td>
<td>0.014</td>
<td>0.014</td>
</tr>
<tr>
<td>Silver</td>
<td>&lt; 0.03</td>
<td>&lt; 0.03</td>
</tr>
</tbody>
</table>
The primary components of FGD by-products have been evaluated in some detail by many researchers. The Electric Power Research Institute [EPRI, 1988a] reported that calcium, silicon, aluminum, and hydroxide components increase cementitious behavior which is related to strength development. On the other hand, sodium and sulfate components are known to promote expansion reactions that cause a deterioration of long-term strength in the wastes. The sodium and sulfate compounds are also soluble and affect the resulting leachate quality and the strength in terms of dissolved solids. These soluble materials may show high strength after initial curing, but lose strength after soluble components dissolve. Calcium components are much less soluble than sodium components.

FGD by-products yield a highly alkaline leachate in the range of pH 9 - 12.7 [Bigham et al., 1993]. The pH affects leachable species. Many species of FGD wastes are leachable primarily at low pH, while high pH deters solubility of these species [EPRI, 1988a]. The higher pH results in increased strength development and lower permeability of the disposed materials. The main potential for corrosion exists in areas where the solid wastes might encounter moisture. In general, high pH values in FGD wastes are of concern because the alkaline solutions formed when the wastes are hydrated, are corrosive.
2.3.3 Engineering Properties

A number of potential applications for FGD materials have been considered including the following: soil and sludge stabilization, agricultural uses, structural fill, stabilized road base, and cement production [EPRI, 1988c]. The potential of FGD by-products in high volume field applications such as a highway embankment is currently under study at OSU.

The most important properties of embankment materials are compressibility, shear strength, swell potential and permeability. The basic engineering properties of similar materials such as fly ash and bottom ash have been extensively investigated. This has provided general information on properties of ash-based materials. Some of engineering properties of FGD wastes are directly related to the properties of the ash. However, there has been very little work done on the engineering properties specifically related to dry FGD by-product and their use in construction projects. There have been a few papers [Adams et al., 1992; Donnelly, 1991; Kim et al., 1992b; Wolfe and Wu, 1992; Bigham et al., 1993] that have published the engineering properties obtained from laboratory work.

Unit Weight. It is oftentimes advantageous for a material to have a low unit weight in high embankments or where fills are placed on relatively weak subsoils. Most
natural soils have dry unit weight of 1.6 - 2.1 g/cm³. Bigham et al. [1993] reported that the range of compacted dry unit weights for FGD by-products is 0.8 - 1.8 g/cm³ with an average value of 1.14 g/cm³. The low unit weight of FGD materials may represent a considerable reduction in the surcharge placed on in-situ materials.

**Compressibility** To evaluate the settlement of an FGD embankment, the most appropriate measure of compressibility can be made using data from one-dimensional consolidation tests. Low compressibility is a desirable quality in a highway embankment material for a smooth roadway profile and for minimizing differential settlement between highway structures. Wolfe and Wu [1992] showed that the FGD materials are much less compressible than typical natural materials that might be used for embankment fill. Bigham et al. [1993] reported the virgin compression index, $C_v$ ranged from 0.011 to 0.051 and the recompression index, $C_r$ ranged from 0.002 to 0.004 for compacted FGD materials.

**Shear Strength** One of the most important characteristics of the FGD by-product in its use as roadway embankment material is its strength. In the FGD by-product, strength develops over time due to the cementitious behavior of the material. Wolfe and Wu [1992] observed that the FGD materials possess strengths considerably in excess of what would be required in a
structural fill. Maximum compressive strength ranging from 2.4 kg/cm² to 155.1 kg/cm² after 28 days of cure were measured. The high strength property of this material should be expected to result in low compressibility and good stability characteristics [Meyers, 1976].

**Permeability** The permeability is, in general, a function of grain size, degree of compaction, and curing period [EPRI, 1988a]. Kim and Wolfe [1992b] conducted falling head tests to determine the permeability of saturated FGD samples at densities typical of those specified for engineered fills. They observed that: (1) measured permeability coefficients ranged from $4.3 \times 10^{-5}$ cm/sec to $9.1 \times 10^{-10}$ cm/sec, which is typical of the range expected for silts and impervious clays; (2) the permeability gradually decreased as the material aged; and (3) the minimum permeability was obtained at moisture contents wet of optimum. It was observed that the permeability of the disposed material also help determine drainage design and how much leachate is generated. The relatively low permeability of compacted FGD materials means that seepage will move around a FGD material body.

**Swelling Potential** Swelling can be a very important consideration because of its potential effect on the rate of serviceability loss [AASHTO, 1986]. Adams and Wolfe [1992] observed that some compacted FGD samples possess swelling potentials that would impact on the suitability
of these FGD materials in certain zones within the fill. They conducted a series of laboratory tests for the purpose of identifying the swelling potential of the compacted samples. In these tests, the volume of the compacted specimen was measured over an extended period of time. The volume increases ranged from as little as 0.4% to as high as 90% of the original sample volume. FGD waste products which used dolomitic sorbent typically contained CaSO$_3$ and showed low swelling potential, while the waste materials in which calcitic sorbent was used, typically contained CaSO$_4$ and showed a high swelling potential. EPRI [1988a] reported that the magnitude of swelling depends on the curing age of the FGD by-product and on its original chemical components [EPRI, 1988a; Meyer, 1976]. The swelling potential of FGD by-products should be considered in the design of high volume applications. A drainage system can be effective in minimizing embankment swelling if it reduces the availability of moisture for absorption [AASHTO, 1986].

2.4 Design of Embankment

Roads often need to be constructed at elevations above existing ground levels. When the roads are built above the ground, either a bridge or an earth embankment can be specified. In most cases, where it is feasible, the more economical choice is the embankment. Therefore,
embankments are a major component of the road support system [NCHRP 8, 1971].

A complete highway project progresses through the following stages (NCHRP 33, 1978): (1) corridor study, site investigation, (2) route selection, (3) preliminary design, (4) final design, (5) advertising and bidding, and (6) construction. Transportation agencies realize the importance of geotechnical engineers' involvement in all the above stages. When the engineers are properly involved in the projects, the benefits may include: avoiding problem sites, selecting the alignments at minimum foundation treatment cost, and identifying potential construction problems. Development of the most economical and satisfactory design requires proper acquisition and use of geotechnical information supplied by the geotechnical engineers.

The design of an embankment involves a careful evaluation of many factors including the properties of foundation soils, the proposed alignment, available right-of-way and construction materials, and construction schedule. This study is primarily concerned with the embankment design in geotechnical engineering practice, once the alignment is decided.
2.4.1 Standard ODOT Design Procedures

When the project conditions meet a set of standard requirements, handbook design procedures are often considered adequate. In this procedure, the design of an embankment fill is affected primarily by geometric design criteria. The engineering properties of the embankment and its foundation are assumed to be acceptable as long as the standard specifications are followed. In this case most state highway agencies designate only density of the compacted fill, moisture control, lift thickness and type and size of compaction equipment. Implicit in this specification is the assumption that for the standard design, side slope, strength and compressibility of a properly compacted fill will not be a problem [NCHRP 8, 1973]. Table 2.3 shows the required standard design for slopes. According to this guideline and ODOT design manual [1991], most embankments will be designed with slopes that have a ratio of vertical change to horizontal of 1 to 4, or only 14°.

The design of the embankment, especially in the case of extensive fills, poor foundation conditions, and limited right-of-way, must be individually evaluated because in these situations, the specifications listed in the standard design are not adequate. The following section describes the engineered design procedure.
Table 2.3 Guide for earth slope design
[after AASHTO, 1984]

<table>
<thead>
<tr>
<th>Height of Cut or Fill (m)</th>
<th>Earth Slope, Horizontal to Vertical, for Type of Terrain</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Flat or Rolling</td>
</tr>
<tr>
<td>0.0 - 1.2</td>
<td>6 : 1</td>
</tr>
<tr>
<td>1.2 - 3.0</td>
<td>4 : 1</td>
</tr>
<tr>
<td>3.0 - 4.5</td>
<td>4 : 1</td>
</tr>
<tr>
<td>4.5 - 6.0</td>
<td>2 : 1</td>
</tr>
<tr>
<td>Over 6.0</td>
<td>2 : 1</td>
</tr>
</tbody>
</table>

2.4.2 Engineered Design Procedures

The first design requirement for any embankment is stability. The embankment must be stable during and at the end of construction and throughout its design life. When the slope angle falls outside the design guide given in Table 2.3 stability without an analysis cannot be assumed. Ladd [1977] considered three basic components for a stability analysis: the evaluation of the embankment and the foundation conditions, the determination of the appropriate strength parameters, and the selection of the analysis method. The adequacy of the factor of safety depends on the accuracy of all the three components. The factor of safety against failure is determined by conventional slope stability analyses, which are based on
limiting equilibrium concepts. Methods of stability analyses are widely available in the literature, and the computations required for these methods may be accomplished with a wide variety of available computer software. For example, the programs ICES LEASE, STABL, SSTABL (subsequent versions are SSTAB2, UTEXAS, UTEXAS2, UTEXAS3), PCSLOPE, and STABR are available for use on mainframe and personal computers [Abramson, 1993]. One of the more popular stability programs is STABL, which was developed at Purdue University. Subsequent versions that are a superset of the original are available as PCSTABL5, PCSTABL6, XSTABL, and GEOSLOPE. PCSTABL6 was adopted for use in this study. This program will be discussed in detail in Chapter 5.

Traditional factors of safety for embankment at the end-of-construction condition are 1.3 to 1.5 [NCHRP 147, 1986]. Ladd [1977] assumed reasonable factors of safety to be above 1.4 to 1.5. Chirapuntu and Duncan [1976] required minimum safety factor of 1.4. If soil improvement by foundation consolidation is being done, then the minimum factor of safety may be less than 1.3. Low safety factors are appropriate only where detailed and careful investigations, testing, analyses, and monitoring are performed. In these cases, Ladd suggested minimum factors of safety of 1.2 to 1.3.
When the stability analysis indicates an acceptable factor of safety, the next step is to estimate settlement of the embankment. Settlement analysis is an integral part of any embankment design. Terzaghi (1925) developed a theory of one-dimensional consolidation for saturated conditions. Later Biot [1941] extended this theory to three-dimensional situations. Sandhu and Wilson [1969] solved the nonlinear differential equation of consolidation by the finite element method. Wu et al. [1983] applied an elastic plastic, work-hardening stress-strain model to analyze the displacements of an embankment on clay. In that study, they simplified their constitutive model to be applicable to practical deformation analyses in the highway embankment. Good agreement between computed and measured settlements at the center of the embankment was observed.

Ladd [1977] considered primary and secondary compression at vertical loading conditions on varved clays. He modified Terzaghi's assumptions by allowing the variation of $C_v$ with depth and included the effect of lateral drainage on consolidation time. Bergado et al. [1990] compared the measured settlement on highway embankments with the calculated results by several predicative methods. They observed that those methods, which were developed based on existing consolidation theory including the secondary settlement, showed a
reasonable settlement prediction for highway embankments. Urzua [1993] described a microcomputer program EMBANK for computing one-dimensional compression settlement due to embankment loads. The vertical compression was calculated by the equations presented by Lambe & Whitman, Ladd, and Poulos & Davis.

NCHRP Synthesis 29 [1977] has noted that post-construction settlement of a roadway of as much as 1 ft to 2 ft is generally considered tolerable provided: (a) it is reasonably uniform, (b) does not occur adjacent to a pile-supported structure, and (c) occurs slowly over a long period time. Differential settlements are always a problem due to the variability of soil properties and thickness of the compressible layers [AASHTO, 1986]. When long-term settlement is anticipated, flexible pavement also often are used to facilitate future maintenance.

Perhaps more important than settlement of the embankment material itself is the effect of the weight of the fill on underlying natural soils. Settlement and stability of an embankment foundation depend on the weights of embankment material. Forsyth [1976], Moore [1966], and Hartlen [1985] discussed various types of lightweight materials that have been used in or considered for highway embankment. Because the compacted densities of lightweight materials are significantly less than the unit weights of soils ordinarily used in embankment
construction, settlement can be reduced and stability increased by using lightweight embankment fill. Consequently, required sideslopes can be steepened by the use of lightweight materials. Especially where the embankment would pass through an urban area, it may be necessary to reduce the rights-of-way. Sideslope steepening can be accommodated more easily by the use of lightweight materials such as FGD by-products.

2.5 Decision Support System in Civil Engineering

In recent years there have been rapid developments in two technologies aimed at improving decision-making: decision support system (DSS) and expert system (ES) [Gottinger and Weimann, 1988].

2.5.1 Decision Support System

The concepts of DSS began to evolve in the late 1960s to early 1970s by Morton and Gorry [Ford, 1985]. They introduced DSS to support managers in solving specific semi-structured problems using computer-based routines. The decision support system was defined as an interactive system that helps decision-makers utilize data and models to solve unstructured or semi-structured problems [Keen and Morton, 1978]. The DSSs do not provide answers to structured problems; rather, they emphasize direct support for professional judgments required in their decision-
making. Therefore, the purpose of a DSS is to support the user in making a decision by providing quick and applicable knowledge to that decision. The data and models are at the user's disposal, to be used as desired. The structure of a DSS can be described by identifying the major subsystems or components of the system. As illustrated in Figure 2.3, Sprague and Carlson [1982] proposed three basic subsystems for DSS: data, models, and dialog. The data subsystem performs all data-related tasks. The model subsystem contains the library of models and routines to maintain and manage them. The dialog subsystem acts as the interface between the system and the user giving the system the power, flexibility, and usability that are characteristic of a DSS. These three major subsystems provide a convenient scheme for identifying the technical capability.

The concept of DSS has been widely applied in the business area. However, a few papers related to DSS in the civil engineering field have been published. Sieh [1988] used the system to analyze the seepage from earth dams. Kao [1990] applied the system in the field of environmental engineering.
2.5.2 Expert System

The expert system has also become established in the mainstream of decision-making with the DSS during the same period. ES has grown dramatically in different application fields from the DSS. An ES is a problem-solving program that simulates the thought process of a human expert to solve complex decision problems in a specific domain [Badiru, 1992]. The ES is designed to provide the user with a conclusion or decision that is correct all the time. The ES provides conclusions or decisions to the non-expert user that are more likely to be correct than that one the user could provide. A basic framework used to describe the structure of an ES is shown in Figure 2.4. This system contains the three essential components: the knowledge base, the working memory and the inference engine. The knowledge base consists of problem solving rules, procedures and facts about the problem domain. The working memory refers to data structure for the problem domain under consideration. The inference engine is a control mechanism which derives new facts from known facts to arrive at some decision.

In civil engineering, there has been considerable interest in developing decision-making systems in the fields of design, planning, diagnosis and management. Examples can be found in the work of Baker et al.[1989], Ritchie [1987] and Hadipriono [1992] as well as many
others. Baker et al. developed a hybrid design expert system for highway bridge dynamics. This program integrates an expert system with numerical methods. In the areas of planning and diagnosis, Ritchie developed an expert system to help highway engineers in maintenance planning and selection of rehabilitation strategies for asphalt concrete pavements. An expert system has been developed for the planning of the precast concrete construction operation. Hadipriono has applied expert systems in the area of construction management. In many of his works, fuzzy logic and fault tree systems were used in conjunction with the expert system. In geotechnical engineering, Adams et al. [1990] developed a knowledge-based system to diagnose retaining wall failures and suggest preliminary rehabilitation strategies. Chameau and Santamaria [1987] have developed a knowledge based expert system which, based upon user input of site characteristics, suggests the best method of soil improvement. Maher et al. [1991] have developed a hybrid expert system for design with geosynthetics which assists the designer in selecting the most appropriate type of geosynthetic material, performs the required design calculations and provides detailed design information to the system user.

The early works in expert systems in civil engineering developed at a conceptual level and have been implemented
only in prototype systems. Most of the applications to decision making still depend on the system composed by the expert system alone.

2.5.3 Comparison of DSS and ES

Table 2.4 illustrates the differences between DSS and ES. These concepts of both DSS and ES began to evolve in the late 1960s to early 1970s. With fast development of hardware and software in the late 1970s and early 1980s, the development and application of DSS and ES increased rapidly [Ford, 1986]. Most of DSS applications were used to support managerial decision making. The concepts of expert systems were greatly applied in civil engineering.

In recent years the number of applications of decision-making have tried to combine the advantages of DSS and ES. Raghavan [1991] and McGovern et al. [1991] proposed a paradigm for intelligent decision support system which combined the DSS and ES in decision making. The concept of this system is an advanced variation and refinement of the traditional DSS philosophy. In this view, this system is an interactive tool for decision making for well-structured decision and planning situations. The system uses expert system techniques as well as specific decision models to make it a model-based expert system.
Grivas and Reagan [1991] proposed a prototype of decision support system for evaluation and treatment of earth slope instability. In their paper, the expert system was used as a decision support system. The expert system was combined with analytical and data base components to form complete system.

![Diagram of DSS Structure](after Sprague and Carlsons, 1982)
Figure 2.4 A structure of ES [after Badiru, 1992]
Table 2.4 The difference between DSS and ES  
(after Sprague and Watson, 1986)

<table>
<thead>
<tr>
<th>Decision Support System</th>
<th>Expert System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective</strong></td>
<td></td>
</tr>
<tr>
<td>Assist human</td>
<td>Mimic human and Replace him</td>
</tr>
<tr>
<td><strong>Decision Maker</strong></td>
<td></td>
</tr>
<tr>
<td>The human</td>
<td>The system</td>
</tr>
<tr>
<td><strong>Problem Area</strong></td>
<td></td>
</tr>
<tr>
<td>Wide domain</td>
<td>Narrow domain</td>
</tr>
<tr>
<td><strong>Knowledge Structure</strong></td>
<td></td>
</tr>
<tr>
<td>Unstructured</td>
<td>Wall-Structure</td>
</tr>
<tr>
<td>Flexible</td>
<td>Little flexible</td>
</tr>
<tr>
<td><strong>Users</strong></td>
<td></td>
</tr>
<tr>
<td>Individual and/or Group users</td>
<td>Individual user</td>
</tr>
<tr>
<td><strong>Query Direction</strong></td>
<td></td>
</tr>
<tr>
<td>Human queries the machine</td>
<td>Machine queries the human</td>
</tr>
<tr>
<td><strong>Manipulation</strong></td>
<td></td>
</tr>
<tr>
<td>Numerical</td>
<td>Symbolics</td>
</tr>
<tr>
<td><strong>Data Area</strong></td>
<td></td>
</tr>
<tr>
<td>Factual knowledge</td>
<td>Procedural and factual knowledge</td>
</tr>
</tbody>
</table>
CHAPTER III

KNOWLEDGE ACQUISITION AND KNOWLEDGE REPRESENTATION

3.1 General

Knowledge of a specific field (domain) takes many forms. When the knowledge is well defined by a series of equations within a problem domain, algorithmic computer programs, which solve a series of equations by inserting input data into appropriate slots within the equations, are more appropriate than heuristic ones [Buchanan, et al., 1983]. However, when the domain knowledge to be solved is ill-defined or when judgment and experience are necessary tools in finding the solution, decision-making systems embodying a heuristic approach are more appropriate. Knowledge obtained from various domains is usually distributed in a variety of fashions that do not permit simple translation into a program. Thus, the process of extracting knowledge from sources of expertise and transferring it to a program is an important but difficult problem. This process is termed knowledge acquisition [Buchanan, et al. 1983], and it may be
performed by a knowledge engineer or by an automated computer program. Knowledge can be acquired from many sources. The sources are generally human experts but can also be data bases, text books, reports, standard specification, and case studies. In this research, knowledge acquisition was conducted by a knowledge engineer who is the author of the study.

In this research, the domain knowledge is classified into four groups consisting of 1) environmental properties, 2) material characteristics of dry FGD by-products, 3) design and construction work of the embankment, and 4) geotechnical background of the foundation soils. The group of environmental issues refers to the limitations placed by regulatory agencies on the utilization of dry FGD by-products in any construction project of the size and type of a highway embankment. Most of this knowledge was acquired from experts at the Ohio Environmental Protection Agency (OEPA) and its publications. The second group, knowledge about the material characteristics, covers the effects of the chemical components on the engineering properties. Herein, we obtained the expertise from case studies, laboratory tests, and human experts including operating engineers at the power plants, chemical engineers, and geotechnical engineers. The knowledge obtained from this group was stored in the data bases. The knowledge in the third group is related to design and construction of the
embankment using dry FGD by-products as soil replacements. Knowledge in this field was obtained from engineers with the Ohio Department Of Transportation (ODOT) and American Electric Power (AEP) and author's experiences with the Korea Highway Corporation. More information was also obtained from the design manuals and standard specifications published by ODOT and AEP. Embankment performance is associated with the stability and the deformation of both the embankment and the underlying foundation materials. Thus, a considerable amount of geotechnical background knowledge on the foundation is necessary. The author through his training in geotechnical engineering provided this expertise.

Knowledge is distributed in a variety of forms depending on the nature of the subject with which it is concerned. Each individual piece of knowledge in the problem domain is not sufficient for solving the entire problem. However, the partial knowledge becomes valuable in the solution of the domain problem by synthesizing sources of expertise obtained from different fields. It is necessary that knowledge be collected and managed in a convenient and systematic manner and that the system be easily available to all users. Thus, the author developed a knowledge acquisition process to effectively gather and collect this distributed knowledge. The gathered knowledge is organized in the knowledge base which will be discussed in the next section.

The following sections present the knowledge acquisition
process and knowledge representation used in a computer program (IDSSHED) written by the author.

3.2 Knowledge Acquisition Process

The knowledge engineer usually acquires the knowledge through direct interaction with the sources of knowledge. In acquiring the knowledge from various sources, the knowledge engineer proceeds through several phases before producing a knowledge base. Buchanan [1983] and Waterman [1987] classified the knowledge acquisition process into five stages consisting of problem identification, conceptualization, formalization, implementation, and validation. They focused on the domain experts to acquire the sources of expertise. Badiru [1991] also proceeded to acquire knowledge in multiple phases such as finding a good knowledge engineer, establishing the characteristics of the knowledge to be acquired, choosing the domain experts and the transfer of knowledge. In this study, the knowledge acquisition process was comprised of three phases which have been identified as unstructured, structured, and organized phases as shown in Figure 3.1. The three phases of knowledge acquisition are defined in the following sections.

3.2.1 Unstructured Knowledge Acquisition

The first phase of the knowledge acquisition process, the unstructured phase, is designed to identify and characterize
the important aspects of a domain knowledge. This involves establishing the characteristics of the knowledge, identifying the sources of knowledge and collecting related facts and procedures. This phase is generally performed through reviewing the literature on a subject, attending seminars (e.g. on FGD by-products utilization) as well as unstructured interviews with experts. For this work, the domain experts were selected through describing the problem to individuals and groups of experts at ODOT, AEP, and OEPA. Through a general literature review and these interviews with the experts, an initial knowledge base was built. In this process, much of the knowledge of geotechnical applications as well as FGD material characteristics were obtained from the author himself and his advisor. In this phase, the author identified the key concepts and their relationship to each other and established a preliminary framework of IDSSHED. After completing this phase, the information was investigated and summarized to prepare more detailed questions used in the next phase. Thus, a strategy for additional knowledge acquisition was built. This phase has the advantage of not prejudicing the gathering of information.

3.2.2 Structured Knowledge Acquisition

The structured phase of the acquisition of knowledge was performed through structured interviews and focused
literature surveys to obtain detailed information. In the structured knowledge acquisition, the knowledge engineer prepared detailed questions about each field for each domain expert. During this phase the collected knowledge was filtered again and studied to prepare even more detailed questions to be used in another round of interviews. In this phase, a set of design requirements was identified. The knowledge engineer started to develop a data base to store the material properties and numerical programs to analyze the slope stability and settlement. This process continued until the knowledge engineer was satisfied that sufficient knowledge was collected to build the knowledge base of IDSSHED.

3.2.3 Organized Phase of Knowledge Acquisition

Finally, the organized phase of knowledge acquisition focused on mapping the structured knowledge from the previous phases and then developing a prototype system. In this phase, the knowledge engineer refined the structured knowledge and started to formulate the production rules for the prototype program. These production rules usually formed IF and THEN statements. As an example,

IF factor of safety is greater than 1.5
    and total settlement is less than 1.5 ft
    and side slope is not stiffer than 2.0

THEN steepen side slope using lightweight FGD material and show a graphical design.
To create a more workable set of production rules, the knowledge engineer took an active role in discussions with the domain experts about knowledge representation, results of embankment analysis, or correction of a design method for the embankment. This phase was continuously performed to refine and extend the prototype program until a satisfactory system was obtained.

We usually used the interview method to extract the knowledge from the experts. In the unstructured phase, the knowledge engineer sat with the expert and discussed the process of solving a problem through an informal meeting. This interview was used in the preliminary stage of the knowledge acquisition to obtain a large amount of general information. Later in the structured phase, a follow-up interview was conducted to gain detailed information about one particular aspect of the expert's technique. The interview was recorded on audio tape. For the interview in this phase, we usually asked the expert a set of formal questions prepared to extract the expertise. In the next phase, the knowledge engineer repeated this task to build the production rules and refine them.
Figure 3.1 Knowledge acquisition process
3.4 Knowledge Representation

The Knowledge Representation paradigm is to organize the collected knowledge into a form such that the decision-making system can readily access it for decision-making purposes. A number of different knowledge representation methods exists. Selection of the appropriate knowledge representation method is critical to the development, functionality and extension of a decision-making system. The most common way to represent knowledge is through logic, rules, semantics nets, frames and object-oriented programming (OOP) [Dillon and Tan, 1993].

Logic is a representational formalism for expressing knowledge and reasoning about such knowledge, and deducing consistent facts from existing knowledge. It is used to represent and manipulate mathematical relationships in various disciplines of science and engineering [Jain, 1992]. Rules are defined using an IF-THEN syntax that connects one or more antecedent (premise) clauses with one or more consequent (conclusion) clauses. The antecedents and consequents of rules refer to specific facts that describe the state of the world. The rules and facts make up the knowledge base and are analyzed by the inference engine. The term semantic net is used to describe a knowledge representation method based on a network structure. It is a scheme for representing abstract relations among objects in a problem domain. The frames are also organized in a network
structure much like a semantic net. However, frames are different from semantic networks in the sense that frames contain a subset of the properties used in a semantic net. In a semantic network, information about an object can be randomly distributed throughout the knowledge base. By contrast, in a frame the information is grouped together into a single unit. We consider both semantic nets and frames to be frame-based systems [Waterman, 1987]. OOP uses a special organization consisting of objects that represent entities capable of exhibiting behavior. Like rules, objects provide a natural way for handling processes where the wealth and variety of data configurations determine the reasoning paths.

Much of the early work in engineering focused on rule-based systems because the problems typically concentrated on the development of classification schemes. However, rules alone are not enough for the formulation and solution of complex engineering programs, principally because they do not allow sufficient expressive power for describing complex objects and multifaceted concepts [Dym, 1991]. We can now look forward to solving harder and more interesting problems by using OOP. OOP is a relatively new style of knowledge representation which is being increasingly used to develop decision-making systems. In artificial intelligence (AI) programming environments, object-oriented programming is mixed with rule-based and frame-based representations
[KPWIN, 1991]. In this research, OOP is selected as a knowledge representation paradigm.

OOP paradigm represents knowledge through inheritance. Inheritance is a mechanism through which the properties of one class are passed to other classes [Keirouz, 1992]. The hierarchy shown in Figure 3.2 defines the structure through which inheritance takes place. The design of an embankment in IDSSHED is composed of five groups of classes. First is a site survey class, which represents the attributes of the various types of information obtained from site survey. This class inherits four following classes (subclasses) including route location, field exploration, laboratory test, and geometric conditions. Second is the data base class, which has subclasses for determining a type of FGD material defined by the user. Third is the standard specification class, which defines the standard requirements for embankment design. Fourth is a prediction class, which determines the engineering behavior of the embankment based on the three above classes (superclasses). Fifth is the embankment design subclass, which combines all superclasses defined above to design the embankment. All five parts of the classes are required to ensure a clear, flexible, and efficient representation of the design procedures and to enable dynamic data exchanges through the object messages.
3.5 Chapter Summary

In this chapter, knowledge acquisition and knowledge representation of the computer program IDSSHED were discussed. Knowledge in IDSSHED was obtained from documents, the author's experiences, and experts from ODOT, AEP, and OEPA. To collect the knowledge from different fields, a knowledge acquisition process was developed. The process consisted of three steps including an unstructured phase, a structured phase, and an organized phase.

The gathered knowledge was organized into a knowledge base for decision-making. OOP paradigm was used as a knowledge representation method. The OOP represents the knowledge using inheritance mechanisms. In the research described in this dissertation, the knowledge was grouped into five levels of classes.
3.2 Knowledge representation in IDSSHED
CHAPTER IV
DEVELOPMENT OF INTELLIGENT DECISION SUPPORT SYSTEM

4.1 General
The knowledge collected through the knowledge acquisition process was organized into the knowledge representation paradigm as illustrated in Chapter 3. It is necessary to have a system to store and transfer the gathered knowledge to users. There are two systems to improve decision-making such as expert system and decision support system. Each system has its own characteristics to be applied to engineering field. In this research, knowledge-based decision support system is proposed to integrate components of both DSS and ES. The next section describes the proposed model in detail. Basic components of the model will be presented in the following sections.

4.2 A Proposed Model
An expert system (ES) is a problem-solving system that achieves good performance in a specific problem domain. The system processes the knowledge of experts and attempt
to mimic their thinking, skill, and intuition. A fundamental component of the expert system is its knowledge base, which is created by extracting and then presenting the knowledge of experts. With this gathered knowledge, an expert system provides the non-expert user with a conclusion or decision that is correct all the time. An expert system is an independent consultation process that does not integrate with any other external programs. A major challenge facing an expert system is the need for better integration with other decision support models.

On the other hand, a decision support system (DSS) is an interactive system that helps decision-makers utilize data and models to solve unstructured or semi-structured problems. An important function of DSS is the creation of an environment in which a decision-maker can apply his/her quantitative analysis techniques. This system gives the user the ability to approach a problem analysis in a flexible, personal way. Therefore, DSS supports rather than makes decision. The knowledge base in DSS may be in the decision maker's head, in references, or user's manuals, and/or in a series of help command available upon request.

In this research, an intelligent decision support system (IDSS) is proposed to better integrate the components of both decision support system and expert system. The basic idea of this approach is that expert systems can be added onto a DSS to extend its capabilities for performing
functions that the regular DSS cannot perform such as heuristic judgment. Some expert system concepts may be transferable to the area of DSS, which may include inference capabilities as well as heuristic decision structures. In the IDSS, the expert system component provides a base solution based on expertise which could either be accepted or implemented by the decision maker.

Figure 4.1 shows the proposed model of IDSS. The model contains the six essential components of an intelligent decision support system: the knowledge base, inference engine, user interface, data base, analytic model, and design model. The next sections describe the first three components of the model. The remaining components will be presented in Chapter 5.

The operating environment for the decision support system development is a major factor in the success of this development. Many MS-DOS-based systems are available. But they generally suffer from the limited power offered by the PC environment. The introduction of Microsoft Windows 3.1 has added a significant boost to knowledge-based systems implementations. Many software developers have now introduced Windows-based development tools for knowledge-based systems. Knowledge Pro Window (KPWIN) from Knowledge Garden, Inc. was one of the first Windows-based inference engines available on the market. In this study, the shell KPWIN [1993] is used. This system shell uses fully object-
oriented programming mixed with rule and frame-based representations.

Figure 4.1 The proposed model of IDSSHED
4.3 Knowledge Base

Once the desired knowledge has been acquired and a representation method has been selected (as presented in Chapter 3), the next step is the development of the knowledge base. The knowledge base codifies the gathered knowledge about the problem domain. For this research the knowledge tree shown in Figure 4.2 was developed. This tree formulates a structure for the knowledge base and defines the various paths that the knowledge base must permit to reach conclusions.

In IDSSHED the knowledge base was coded using objects and rules. Objects are used to represent the stereotyped items and rules are used to represent heuristics.

4.3.1 Objects

Object-oriented approaches are based on physical or logical entities called objects [Khoshafian, 1990]. Objects combine the properties of both the data (the object's attributes) and the procedures (the object's method). Figure 4.3 shows an example of an object FGD_UCS. In the object, the attributes characterize the object's state and the methods define its behavior. An attribute can be called by several other names, such as slots and instance variables. These attributes (Material_Code, FGD_system, FGD_process etc.) associated with the object can only be directly accessed by the methods (UCS_Average,
Methods represent the actions that can be performed by an object. The method could result in a change of the attribute values of the object. It, therefore, could be treated as a state transformer [Dillon and Tan, 1992]. A method can be a function or procedure that causes an action or change of state to occur. For example, the function UCS_Average and the procedure Write_in in the object are methods (see Figure 4.3).

Objects communicate with other objects using messages. A message is associated with a method within the object. An object with attributes and methods is inactive until it receives a message (the object is then called receiver) sent by other objects (called senders). When an object receives a message, the associated method is activated. As an example, the message FGD_UCS.UCS_Average invokes the method UCS-Average which performs the calculation of the mean of unconfined compressive strength, and returns the values to the sender object.

A class is a set of objects that represents similar characteristics such as the same attributes and methods. Therefore, class shares attributes, methods and associated messages through an inheritance mechanism. Classes with similar properties can be grouped into a higher level class called a superclass or into lower level classes called subclasses. A superclass represents a generalization of the
subclasses, whereas a subclass of a given class represents a specialization of the class above. Classes and objects provide a computable representation of the physical world that is convenient for engineers.

4.3.2 Production Rules

In IDSSHED, there are three types of IF-THEN statements. They are production rules to handle the expertise of the design criteria for an embankment, to select subclasses for the decision support, and to give an explanation to user. The first type of IF-THEN statement can make inferences and arrive at a final conclusion. An example of this rule follows,

IF Factor of safety is less than 1.5 and Total settlement is greater than 1.5 ft

THEN It is necessary to increase the stability and to reduce the settlement using lightweight material like FGD by-product and show a graphical design.

The IF portion in the rule has two conditions of stability and settlement. The THEN portion indicates a type of design method to improve the status of the embankment and provide a graphical design.

Some rules are stored in the knowledge base, but cannot refine the conclusion. They are executed for a value of one attribute. The last two types of the production rules are included in this group. Two examples of rules are given
below:

IF default_value is "mean plus standard deviation"
THEN load a topic Meanplus().

IF Total height of embankment is not equal to the sum of each layer
THEN show error message and make a correction.

In the first example, the rule loads a subclass Meanplus related to the superclass. The subclass shares attributes, methods, and messages from the superclass. This rule determines a decision path. In the second example, the rule simply checks the numerical inputs before the following topics are executed. Most of error statements were handled by this kind of rule.

4.4 Inference Engine

Inference is the derivation of new procedures or conclusions from known facts. An inference engine is the processor of these known facts. In OOP, Knowledge flows through certain relationships in the inference engine known as inheritance links. Inheritance enables the descriptive information to be shared among multiple classes. The principal feature of inheritance is that any class inherits all the properties from its superclass and may also have other properties.

There are three forms of inheritance to infer the knowledge in this system including simple inheritance,
multiple inheritance, and selective inheritance. Most of the inheritance examples have used simple inheritance in which each subclass has only one immediate superclass. In this simple inheritance, a class inherits the properties from its superclass in a hierarchy chain. Figure 4.4 shows an example of class hierarchy with simple inheritance. This example shows how the subclass Avg+Std might be inherited by the parent Engineering_Property. In many situations, it is desirable for a class to inherit from more than one parent class. This is called multiple inheritance. With multiple inheritance, we can combine several existing classes to form combination classes that utilize each of their multiple superclasses in a variety of usages and functionalities. Figure 4.5 gives an example of the class hierarchy (a class structure for multiple inheritance). In this example, the class Stability has three superclasses, Embankment_Material (EM), Foundation_Material (FM) and Geometric_Conditions (GC). Thus, class Stability inherits the properties of classes EM, FM and GC.

Selective inheritance means that only some properties of parent classes are inherited by the child class. For example as shown in Figure 4.5 class Stability may inherit some properties of superclass EM (E1 and E2), some of superclass FM (S1 and S2) and some of superclass GC (G1 and G2).
Figure 4.2 Knowledge tree of IDSSHED
Figure 4.2 (continued)

Figure 4.2 A knowledge tree of IDSSHED
Object . FGD_UCS

Attributes:
Material_Code = "OSU-SPD-03".
FGD_System = "Dry".
FGD_Process = "Spray Dryer".
Power_Plant = "McCracken".

Methods:
UCS_Average = sum/num
UCS_Standard_Deviation =
   sqrt[sum(X_i^2)/n - (UCS_Average)^2]
Write_In = ('ucs.in', UCS_Average,
             UCS_Standard_Deviation).

Figure 4.3 An example of an object FGD_UCS
Dry System

SPD LIMB AFBC PFBC Others

Plant1 Plant2 Others

Engineering Property Chemical Property Leachate Property

Avg+Std Avg Avg-Std Manual Specific

Figure 4.4 Class hierarchy for IDSSHED
Figure 4.5 Class heterarchy for IDSSHED
4.5 User Interface

A computer-aided system requires good communication between the user and the computer. This communication system is called a user interface. The goal of the user interface is to simplify data entry and function choices and to transfer the computer outputs to the user effectively. The interface must be designed so that even though many users of this system are not programmers, direct and effective communication link is established. For example, setting up an input format for FORTRAN programs is generally time-consuming for somebody who is not familiar with the programs, and it may be difficult to transfer an output from the numerical program to become an input to the shell. A user interface is therefore suggested as a bridge to help data transfer among the shell and users. It is assumed that the users would be engineers but not necessarily computer experts.

The characteristics of a good user interface are: (1) ease of use; (2) minimization of error making opportunities; (3) efficiency of decision management, (4) flexibility of modification [Kao, 1991]. A good user interface is expected to require a minimum of learning time. To achieve the characteristics of a good interface IDSSHED employs screen objects, text, and graphics. A menu driven interface with a pointing device (mouse) to make most option selections in the program is developed. By
moving a mouse and clicking a button on a desired menu item, text object or graphic object, a function can be easily selected with the response displayed immediately after selection.

4.5.1 Screen Objects

The shell provides a collection of predefined objects. These are the most common usage of screen objects. The predefined screen objects employed in the IDSSHED user interface are windows, menus, buttons (push button and radio button), boxes (check box, list box, and combo box) and edit lines. A detailed description of each object can be found in the reference KPWIN [1991].

4.5.2 Text

The primary way the computer communicates with the user is by displaying text on the screen. When the user needs more information on the given topic, the hypertext function is used.

Hypertext can be identified on screen by changing the fonts or colors of the text linked to other topics. The function of hypertext is to provide non-sequential access to textual or graphical information stored in the knowledge base. This function supplies help statements to assist users in the system.
4.5.3 Graphics

Graphics can be a very powerful way to communicate information. In the shell KPWIN, we use graphics not only to display information, but also to receive information. The two types of graphics used in IDSSHED are fixed-image graphics and dynamic graphics. The first type is drawn using a graphics package such as Paintbrush and stored in the external files in the form of bit-mapped files. This style of graphics was used in the help statements and hyperregion functions. In the second type, figures are drawn dynamically using the data created by the dynamic data exchange among subsystems. These graphics can summarize large volumes of data and clearly transfer the meaning of the data to user. The dynamic type of graphical user interface will be discussed in detail in Chapter 5.

Hyperregion has the function to receive the information through a designated region on a graph. The first type is usually used for this function. This function is usually defined over a picture displayed on the screen. The content of the picture along with screen text lets the user know what can be selected. The search for topics linked to hyperregions is the same as for hypertext.
4.6 External Files

When large amounts of text, data, or graphics are used in an application, it is most efficient to keep them in external files. In IDSSHED all three types of external files are used. Text files are used to store descriptions such as design procedures for embankments and help statements. The text is stored in seven different files according to objects. The contents of these files are displayed on screens by the use of hypertext function described in the previous section. Data files such as input and output data created from numerical processing and data base are also included in the system. Graphic files are stored in the form of bitmapped images. These bitmap files require large amount of memory. In IDSSHED 80 bitmapped graphics were used which required 2.8 MB of memory in external files. These graphics are displayed on the screen by the function bitmap.

4.7 Chapter Summary

In this chapter, an intelligent decision support system was proposed to integrate the components of both DSS and ES. The proposed model consists of six components: knowledge base, inference engine, user interface, and three subsystems (data base, numerical processing, and interactive graphics). The first three components were discussed in this chapter. The shell KPWIN was selected as
the operating environment for the model.

The knowledge base stores the gathered knowledge in the form of objects and production rules. Objects are physical or logical entities. Rules are IF-THEN statements. The inference method was performed by the inheritance of the objects. The user interface was designed to simplify data entry and function choices and to transfer the computer outputs to the user effectively. A menu-driven interface with mouse was developed. Hypertext and hyperregion functions were used to send and receive information through texts and graphics. To enhance the system performance, large text, data, and graphic files were stored as external files.
CHAPTER V
SUBSYSTEMS

5.1 General

A major challenge facing computer-based design systems is the integration of the expert system with the decision support system. System integration involves the linking of subsystems to form composite systems. The integration system facilitates a more dynamic and responsive system for making decisions. The system described in this thesis uses subsystems such as the data base to manage the information relating to the material properties of the FGD wastes and foundation soils, numerical programs to calculate slope stability and settlement for the embankment, and interactive graphics programs to present the design of the embankment.

5.2 Data Base Subsystem

The typical geotechnical project begins with an effort to collect basic properties data. Data for the project may come from several sources, including field instrumentation,
site investigations, laboratory tests, and model studies. A collection of these data is called data base. Often these data need to be correlated and interpreted to give users better access to the range of data. The data base manages data and provides the information to be used in decision making. In this way, the usefulness of engineering data bases can be extended to include support of engineering design decisions.

The data base is an especially important component of a decision support system, as was shown in Figure 4.1. The data base in this study provides information on what range of values is typical for the properties used as embankment materials and provides the designer with default values to be used as needed in the numerical calculations.

5.2.1 Data Base of FGD By-products and Underlying Soils

The data base (IDSSDB) used in the present system consists of two major parts: 1) the data base for FGD by-products (FGDB) and 2) the data base for foundation soils (SOILDB). A schematic of the system data base is shown in Figure 5.1. The following sections will discuss how these data bases are built.
5.2.1.1 Data Base for FGD By-products

Knowledge of the characteristics of the FGD by-product is needed in the planning, design and construction of both disposal and utilization projects. Although FGD by-products have been produced in large quantities for years, the amount of properties information available is relatively small. Much of the work that has been done with FGD by-products has been limited to a particular FGD material and a specific project. As such, the information developed is usually not of general interest and is seldom published. The data that are available are often incomplete or limited.

In some cases, data are available only on FGD that has been mixed with other materials for utilization or disposal. These materials include lime, soil, bottom ash, and fly ash. In these cases, data on the FGD alone was not determined or presented.
All of these factors contribute to the fact that the available information on the properties of FGD by-products is limited. This situation makes both conceptual design and evaluation of test results difficult. Furthermore, engineers working with FGD by-products are faced with not having information on the values and range of values typical of physical and engineering properties. In many cases, design engineering with FGD by-products proceeds as though the FGD was a totally new material rather than one that is very common.

To provide a better understanding of the typical values and range of values for FGD material properties, a data base of values was established from published and unpublished sources. Mainly, the data base was first built with data from laboratory tests conducted at the Ohio State University on 59 separate samples from 14 different power plants located in the U.S.

The specific characteristics of the FGD by-product produced as a result of stack gas scrubbing depend on several factors including the sorbent used, the sulfur content of the coal and the mechanics of the scrubbing process, and the handling and storage systems. Furthermore, the properties of FGD by-product can be modified during construction and, once placed, may be affected by environmental conditions. In the data base, these factors influencing the properties of FGD by-product and the
properties themselves were grouped in five separate tables: 1) information on the power plants (PCODE), 2) sample information (MCODE), 3) leachate properties (LEADB), 4) chemical properties (CHMDB), and 5) engineering properties of the FGD by-products (ENGDB).

PCODE File: This file contains the relationships between power plant facilities and operating systems and processes. A sample of the information contained in the PCODE file is shown in Table 5.1. The table consists of 7 columns. The first column, PCODE#, is the primary key of this table. The second through fifth columns store the general information for particular power plants. The sixth and seventh columns contain information concerning the overall type of cleaning system used and the specific process used to remove the SO₂ gas, respectively.

MCODE File: Each power plant will produce samples which have different material properties. This file contains a set of particular attributes related to each sample. A portion of the MCODE file containing several factors influencing each sample is shown in Table 5.2. MCODE# is the primary key for this table. PCODE# is a foreign key which is obtained from the PCODE table.

Data Files: There are three data files (LEADB, CHMDB, ENGDB), which inherit from MCODE File, to store the properties of FGD samples. LEADB file includes leachate properties, CHMDB file stores chemical properties, and
ENGDB file contains engineering properties. As an example, Table 5.3 shows the leachate data files (LEADB) in tabular format.

Table 5.1 An example of PCODE file

<table>
<thead>
<tr>
<th>Plant Code</th>
<th>Plant Name</th>
<th>Plant Location</th>
<th>State</th>
<th>Capacity</th>
<th>FGD System</th>
<th>FGD Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>100 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>P2</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>635 MW</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>P5</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>70 MW</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
</tbody>
</table>

Table 5.2 An example of MCODE file

<table>
<thead>
<tr>
<th>Material Code</th>
<th>Plant Code</th>
<th>Sampling Point</th>
<th>Sampling date</th>
<th>Sorbent Type</th>
<th>Coal Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSU-SPD-03</td>
<td>P1</td>
<td>Baghouse</td>
<td>4/01/91</td>
<td>Slaked lime</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-06</td>
<td>P2</td>
<td>ESP*</td>
<td>11/16/90</td>
<td>Dolomitic lime</td>
<td>Bituminous</td>
</tr>
<tr>
<td>TID-FLB-01</td>
<td>P4</td>
<td>Cyclone</td>
<td>8/28/91</td>
<td>Dolomitic lime</td>
<td>Bituminous</td>
</tr>
</tbody>
</table>

* electrostatic precipitator
<table>
<thead>
<tr>
<th>Leachate Code</th>
<th>As</th>
<th>Ba</th>
<th>Pb</th>
<th>Sc</th>
<th>Others</th>
<th>Material Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>.005</td>
<td>.29</td>
<td>.012</td>
<td>.005</td>
<td>--</td>
<td>OSU-SPD-03</td>
</tr>
<tr>
<td>L5</td>
<td>.005</td>
<td>.23</td>
<td>.001</td>
<td>.005</td>
<td>--</td>
<td>EDG-LIM-14</td>
</tr>
<tr>
<td>L12</td>
<td>.005</td>
<td>.17</td>
<td>.001</td>
<td>.005</td>
<td>--</td>
<td>TID-FLB-03</td>
</tr>
</tbody>
</table>

### 5.2.1.2 Data Base for Underlying Soils

In any project involving structures placed on the ground the engineering properties of the foundation soils must be determined. To provide a better understanding of the typical values and range of values for underlying soils properties, a data base was established from published and unpublished sources. The main source for the data base came from engineers at ODOT. The data base was expanded by including data published in journals and text books [e.g. Das, 1985; Ladd, 1977].

The data base was stored in a table, an extract of which is shown in Table 5.4. The soil properties represent average values which were obtained from a variety of published sources. They were arranged according to the group numbers using the AASHTO soil classification system. This system was chosen because it is widely used in state and county highway departments. According to this system, a
soil is identified as belonging to one of seven major groups, A-1 through A-7. Soils classified under groups A-1, A-2, and A-3 are granular materials. Silt and clay-type materials are classified under groups A-4, A-5, A-6, and A-7. Because the AASHTO system does not include classification for organic soils, OL, OH, and Pt used in the Unified system for organic soils were added to the AASHTO system.

Table 5.4 An example of soil data

<table>
<thead>
<tr>
<th>Soil Code</th>
<th>Soil Class</th>
<th>Moisture Content [%]</th>
<th>Strength [kg/cm²]</th>
<th>Permeability [cm/sec]</th>
<th>Compression Index</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>A-1-a</td>
<td>12</td>
<td>2.3</td>
<td>1.1x10⁻³</td>
<td>0.12</td>
<td>*</td>
</tr>
<tr>
<td>S5</td>
<td>A-7-5</td>
<td>35</td>
<td>0.6</td>
<td>2.3x10⁻⁶</td>
<td>0.18</td>
<td>*</td>
</tr>
<tr>
<td>S13</td>
<td>Pt</td>
<td>92</td>
<td>0.3</td>
<td>1.8x10⁻⁷</td>
<td>0.34</td>
<td>*</td>
</tr>
</tbody>
</table>

* ODOT field records

5.2.2 Data Base Management System

A data base contains a structure that defines the relationships among its records. This is called a data model. Notable data models are the network, hierarchical, and relational models [Date, 1987]. Software that implements any of these data models is called a data base management system (DBMS). A DBMS is used for the creation, manipulation, and maintenance of information by users. The
structure of the data base in this system is referred to as a relational model.

5.2.2.1 Relational Data Model

In a relational data base, tables are used to store the data, each table being called a relation. The entire data base can be imagined as a collection of relations. Examples of relations are shown in Tables 5.1-5.4. The individual rows of a table are called records or tuples. The columns in the table are referred to as attributes or fields. The relational model is very simple because it is built up of only tables. However, it does not give a very good visual overview of the data base. The model describes relationships in the same way as entities. In general, it is important to have a comprehensive overview of the data base, and it therefore may be useful to supplement a relational model with an Entity-Relationship diagram (E-R diagram). Figure 5.2 shows a relational scheme with an E-R diagram developed for the FGD material evaluation process. In this figure, the relational scheme is represented by hierarchical levels to give better visual overview and understanding. Relationships between entities come in three flavors: one-to-one (1:1), one-to-many (1:m), and many-to-many (n:m) (see Figure 5.3).

To setup a relationship between two tables - for example, the PCODE and MCODE tables, a one-to-many relationship is used. In this relationship, a record in the
table PCODE can have more than one matching record in table MCODE. In this case, the column PCODE# is used as the primary key, which is a unique identifier for a table. To find the relationship between MCODE and the three property tables - LEADB, CHMDB, and ENGDB, a one-to-one relationship is applied. In a one-to-one relationship, a record in the table MCODE can have no more than one matching record in the tables - LEADB, CHMDB, and ENGDB. This type of relationship is unusual because in many cases, the information in the two tables can simply be combined into one table, but the information of the material properties of FGD by-products is separated from the material-code file, MCODE. By separating the data into four tables, we can easily manipulate the data base because the four tables have their own group characteristics.
Figure 5.2 The relational scheme with E-R diagram for FGD material evaluation process
Figure 5.3 Alternative symbols for E-R diagram

Relations can be illustrated by means of tables such as the previously shown Tables 5.1 - 5.3. In this relational data base, two operators (SELECTION and JOIN) are used to create a relation. A SELECTION is a subset of the set of the rows in a table. In the system, a SELECTION is made on the basis of the contents of three attributes FSYST, FPROC, and PNAME in the table PCODE. In the computer program, the selection proceeded in a tree walk. Figure 5.4 shows a hierarchical representation of the selection process. The following code shows a statement of selection process in the table PCODE.

FILTER = (FSYST = "FGD_system" and FPROC = "FGD_process" and PLANT = "Plant_Name")
After creating a new relation using the SELECTION in a table, the JOIN operator was used to establish a relation with other tables. The JOIN statement requires the two tables to have one or more fields in common over which the join can be made. For each entry in the first table which was created by the SELECTION operator we search through all the entries in the second table with the same value in the
key column as the entry from the first table file. After this, the entry from the first table is concatenated with each of the entries found in the second table. The following code illustrates the function of JOIN in the data base.

JOIN PCODE and MCODE
where PCODE (PCODE#) = MCODE (PCODE#)

5.2.2.2 Data Base Inference Engine

The data base approach in IDSSHED is to have a tight integration of the host program (shell) and the data base subsystem. Figure 5.5 illustrates conceptually how the integration works. The inference mechanism of the data base is integrated into the inference engine of IDSSHED. This means that data base queries and data base updates are all under the same transaction control mechanism. It also means that the IDSSHED has a single optimizer and the data traffic between the two engines is minimized. The inference engine handles the flow of control over inferencing during execution of the programs. The inference engine also keeps track of the explanations to provide the user reasons, and the flow of control on goals that succeed or fail.
5.2.2.3 Data Analysis in Data Base

The data base management system (DBMS) creates, reads, writes, and maintains data files. KPWIN by itself does not support the ability for DBMS. However, KPWIN provides an efficient mechanism for accessing external routines using dynamic link libraries (DLL). DLLs are libraries of
procedures that applications can link to perform their activities. A data base DLL adds data base access capabilities, while a Math DLL supports mathematical functions to KPWIN. In IDSSHELD, DBMS requires managing individual records and also calculating averages and standard deviations of field values from the records assigned by the SELECTION and JOIN operators. Figure 5.6 shows an example of parameters calculation using DLLs. In the example, @KPDBLIB.TPX in database DLL and Load_Library in math DLL load the libraries of database and mathematical topics into KPWIN and make all the functions accessible within the application. For example, the square root of \[ \frac{?squcs/?num - ?aucs*?aucs}{?aucs} \] is calculated by calling the math function \texttt{sqrt} in the library using the KPWIN function user.

The average \texttt{(AUCS)} and the standard deviation \texttt{(STDUUCS)} are automatically assigned to topics accessible as the parameters to be shown and the default values.
mathLibrary is load_library('KPMATH.DLL').
* Partial summation of field values *
squcs = 0.
repeat
    squcs is ?squcs + (first(?kkk)*first(?KKK))
    and kkk is rest(?kkk)
until ?kkk is [ ].
* Total number of field values *
x = total(?tot).
topic total(list).
    num = 0.
    apply (addto, ?list).
    total = ?num.
topic addto (x).
    num = ?num + 1
end.
end.
* Summation of field values *
sum = 0.
repeat
    sum is ?sum + first(?tot) and
    tot is rest(?tot)
until ?tot is [ ].
* Average and standard deviation of field values *
AUCS is ?sum/?num.
STDUCS = user(?mathlibrary, sqrt,[?squcs/?num -
?aucs*?aucs]).

Figure 5.6 An example of a calculation in the data base using DLLs
5.3 Numerical Processing Subsystem

Once the appropriate material properties and design requirements have been identified, the determination of structural performance is made by comparing loads with strengths or other performance criteria. This comparison is generally made numerically. The numerical analysis for the highway embankment is primarily concerned with the stability and settlement calculations necessary to evaluate the behavior of the foundation and fill materials. In the present system, we use recognized methods for slope stability and settlement calculations. At this time, the system uses the FORTRAN programs PCSTABL6 [Humphrey, 1986] for slope analysis and SETFGD developed by the author for settlement.

5.3.1 Slope Stability

PCSTABL6 which was developed at Purdue University was selected as the program to determine slope stability. The version of STABL, which has been modified in many ways since first introduced, implemented in IDSSHED is called PCSTABL6. The program has been adopted by many agencies. Favorable comparisons of the factor of safety values generated by STABL with those for the same surfaces by other methods of slices were reported by Boutrup (1977).

PCSTABL6 has the capability of analyzing potential failure surfaces by including isotropic or anisotropic soil
parameters, piezometric water surfaces, tieback, surcharge, earthquake and reinforcement loads. The program is a general purpose limit equilibrium slope stability analysis program with solution options that use the simplified Janbu and Spencer methods with circular, sliding wedge, or irregularly shaped sliding surfaces or the simplified Bishop method with circular surfaces.

In IDSSHED, the Simplified Bishop method was used as the default method in the stability analysis. This method satisfies the conditions of overall moment and vertical force equilibrium, but does not satisfy horizontal force equilibrium. The basic assumption of this method is that the side forces are horizontal. Whitman and Bailey (1967) have shown that numerical difficulties may occur because the normal force on the base of a slice may become very large or negative when the failure surface is steeply inclined. If these difficulties do not arise, the method gives a result shown to be within about 5% of the results calculated by the more complete methods which satisfy all conditions of equilibrium.

5.3.2 Settlement

Loadings cause soil layers to undergo a certain amount of settlement. This compression is due to compression of the soil skeleton caused by the rearrangement of soil particles accompanied by a dissipation of water pressure
and expulsion of water from the soil. Settlement caused in soil due to loading may be divided into three parts: (1) immediate settlement due to elastic deformation taking place at constant volume; (2) a settlement due to primary consolidation wherein volume change is controlled by the change in effective stresses as result of the dissipation of excess pore water pressure; and (3) settlement due to the so-called secondary compression wherein volume change continues after excess pore water pressures have essentially dissipated.

In this research, a program SETFGD was developed based on time rate of one-dimensional consolidation theory proposed by Terzaghi but was modified so that the theory properly represents actual consolidation conditions in the laboratory and in the field. In the program, primary and secondary consolidation were considered in the calculation of total settlement of an FGD embankment. The primary consolidation was estimated by assuming one-dimensional compression at the center of embankment. The appropriate equation for the calculation was shown in Appendix D. The most important factor affecting amount of primary settlement is the stress history of the foundation soil. Therefore, the program considered the effect of maximum past stress on the compression. It also considered that with a finite embankment width the vertical stress increment decreases with depth. The secondary consolidation
was estimated by the equation shown in Appendix D. For highly compressible soils and organic deposits it may be a large portion of the total settlement.

Rate of consolidation was estimated by Terzaghi's consolidation theory. The key to estimating the rate of settlement is to select a proper value of coefficient of consolidation \( C_v \). The Terzaghi theory assumes that \( C_v \) is constant with depth. But it is usually not the case in practice. SETFGD considers that \( C_v \) varies with depth. Six different types of vertical drainage conditions were considered in the calculation (see Figure F.15). The equations to evaluate the rate of settlement are given in Appendix D. The average degree of consolidation \( (U_v) \) for vertical drainage was computed at any depth in the foundation. The average degree is applied to the total settlement \( (\rho_f) \) to obtain the consolidation settlement \( (U_v. \rho_f) \) to that time and to obtain the remaining settlement \( [(1-U_v).\rho_f] \) at that time.

The detailed descriptions of the program development are included in Appendix D.

5.3.3 Integration of This Subsystem into the Shell

In the current computer application, it is important to be able to share data among several programs, thereby enabling data generated by one program to be processed by another.
A file is a collection of data that is physically located in an external file rather than the main program. There are two kinds of files in FORTRAN: direct access files and sequential access files. In a direct access file, it is possible to reference any record at any time (access order is random). In a direct access file, all records must be of identical length. A sequential access file has the property that its records must always be processed serially, starting with the first. At any point, the next record to be processed is the record following the last one processed. The files used in the programs were created as sequential access files because the shell KFWIN can manage sequential access files more easily than random access files.

In order to create a file for later use, a sequential file in the shell is generated and the data are written onto a file for the appropriate FORTRAN program. Figure 5.7 illustrates how to interact with the shell and the FORTRAN programs. For example as shown in Figure 5.8, to create the input parameters to be used in STABL, the shell creates a sequential file SLOPE.IN and stores those parameters in it.
Figure 5.7 A schematic diagram of the communication between the shell and the FORTRAN programs.
new_file('slope.in').
write('slope.in',?comcardo,?titleo,?nunbdo,
'#$s',?numsufo,
'$n$#$s',?x1,?y1,?x2,?y1,1,'#n',
'#$s',?x2,?y1,?x3,?y2,2,'#n',
'#$s',?x3,?y2,?x4,?y2,2,'#n',
'#$s',?x2,?y1,?x4,?y1,1).

Figure 5.8 An example of creation of sequential file in the shell

If the file SLOPE.IN does not exist, it is created. If the file already exists, the file is opened and the new text is appended to the file. To overwrite a file, the function NEW_FILE destroys the existing contents of the file, leaving an empty file for new writing. After executing the FORTRAN program, the FORTRAN program generates a sequential access file (SLOPE.OUT) to store the output. The shell accesses these files and uses them as a knowledge base. As shown in Figure 5.9, the shell captures the parameters in the file (SLOPE.OUT) and assigns them to the appropriate topics (OFOS and FFOS in this illustration) in the knowledge base.

set_file_pos('slope.out',10,beginning).
SFOS is read_line('slope.out').
set_file_pos('slope.out',-10,end).
FFOS is read_line('slope.out').

Figure 5.9 An example of assigning topics from a sequential file created in FORTRAN program
5.4 Graphical Design Subsystem

Computer graphics have proven to be a powerful method for presenting visual information and allowing interaction with users in developing models and designs. Interactive graphs have become extremely useful not just for evaluating the material properties but also for the synthesis tasks of the slope stability and settlement in embankment design. Figure 5.10 shows how the graphical design subsystem communicates with data base and numerical processing subsystems through the shell. The shell runs two subsystems and results in input files for the graphic subsystem. Then, the shell runs the executable programs to draw screen objects. In IDSSHED, the graphic subsystem has capabilities to summarize large volumes of numerical data obtained from both subsystems and to allow the engineers to visualize the trends in the data. The following sections will describe relationships among the subsystems.

5.4.1 Interactive Graphics with Data Base

Large volume of records for material properties of FGD by-products were summarized graphically. The graphics provide better understanding of the characteristics of the material properties. Eleven executable programs were developed to present the distributions of leachate, chemical, and engineering properties of FGD by-product. Leachate properties were drawn using bar charts showing
percent of EPA leachate criteria. The distribution of chemical components was presented using pie chart in percent over total weight. To plot a wide range of data points for engineering properties, various ranges of X and Y axes were defined. In addition to individual values, maximum, minimum, average, and standard deviation values of each property are included in the graphs. The source codes for the graphs are attached in Appendix E.1 through E.3.

5.4.2 Interactive Graphics with Numerical Programs.

Two executable programs were created to plot the output data obtained from stability and settlement calculations. Ten possible failure surfaces and a critical failure surface were drawn (see Figures F.13, F.14, F.36 and F.37). The progressive settlement was plotted as a function of consolidation time (see Figure F.19 and F.40). Source codes are attached in Appendix E.4 and E.5.

5.4.3 Interactive Design of Embankment

In this system, the performance requirements for an embankment design are produced after evaluation of fill materials and numerical calculations have been executed. The system shell creates input files using the data obtained from the data base and numerical subsystems and exports the files to a graph program as described in the previous section. The graph program is automatically
activated and produces the final designs of the embankment. The design considered different side slope angles and fill types for the embankment. The areas resulting from different side slopes configurations were calculated. The fill types are determined from geometric conditions and properties of fill material. Figure 5.11 shows the four types of fill considered in the system.
Figure 5.10 A schematic diagram of the communication between the shell and the graphical design programs.
Figure 5.11 Types of embankment fill
CHAPTER VI
TESTING, VALIDATION AND CASE STUDY

6.1 General

Knowledge-based systems, which are less assertive than conventional programs, depend mainly on flexible and reliable heuristics that may not have been subjected to rigorous scrutiny. Therefore, any knowledge-based system must be tested and validated before being applied to actual decision making [Badiru, 1992].

In this research, three levels of quality control procedures were conducted: testing, validation, and case study. The first level of procedure consisted of testings done by the knowledge engineer (the author) and his advisor. The second was additional testing (validation) conducted by area experts who were involved in the development of the system as well as several who did not participate in the system development. In the third level, a case study supplied by ODOT field engineers was examined. The following sections will discuss the procedures in detail.
6.2 Testing

Testing is the process of examining whether or not the system is functioning as intended. Thus, testing checks the semantics of the systems. This process was done through repeated examination of the system performance. The discussion may include checking for decision procedures, subsystem linking, error analysis, and maximizing system performance.

In order to test the system, decision procedures in IDSSHED were first evaluated. In this process several decision trees were created and compared in usability, efficiency and run time. Figure 6.1 shows a decision tree developed for IDSSHED. The tree conceptually illustrates the decision procedures of embankment design in IDSSHED. This tree consists of two steps: standard ODOT design procedure using soil as a fill material and engineered design procedure using either soil or FGD by-products as the fill material.

Once the decision tree was tested, subsystem linking was tested. Because there are several hundreds of objects and attributes for dynamic data exchanges (messages) among the subsystems, it is essential to send and receive messages consistently. For example, numerical processing subsystems STABLE.KB and SETTLE.KB often repeat the calculation process when input errors occur or when a new result is required. In this step, some of the previous
inputs are held and others are replaced. Through this process there are many conflicting rules to update the input files for other subsystems. To test the consistency of the messages, a calculation result by the system was compared with hand calculations using known examples.

The system has many places where errors may occur because it requires the input of numeric data from the user. Through execution of the system, possible places for errors to occur either mechanically or logically were investigated. To reduce the possibility of errors, pop-up windows and help statements using the hypertext function were liberally employed. For example, if the soil foundation thickness is divided into three layers, the summation of the three layers should be equal to the foundation thickness. In this case, if the user's input does not meet this rule, then a pop-up window will appear with a message for violations and ask the user to re-input the soil foundation thickness. This kind of logical mistake is hard to find in the results of calculations later, but with the system of checks included, many general mistakes are preventable at the early stages. The chance of using the wrong input values in the numerical calculations is therefore significantly reduced.

Finally, the system was tested to maximize its performance. This is very important for large application programs like IDSSHED. As subsystems were integrated and
text and graphic files were attached, the system required a large working memory, which causes slow execution. IDSSHED requires 1.6 MB of execution memory. In the test process, the execution time was measured for each subsystem. Several techniques were applied for maximizing system performance. First, the size of individual text files was reduced to less than 20 KB in memory. Second, texts, data, and graphic files were saved in external files instead of keeping them all in the main program. Topics were listed hierarchically. Unused resources such as fonts, bitmaps, and hyper-regions were removed.
Figure 6.1 Decision tree in IDSSHED
6.3 Validation By Experts

After the testing sequence by the author and his advisor, the validation process was initiated. Validation concerns a diagnosis of how closely the IDSSHED's decision supports a real application. This validation was performed by experts who participated in the development of the system. IDSSHED was also validated by other experts who were not involved in the development of the system.

A validation tree was developed based on the decision tree. Figure 6.2 shows the validation tree for IDSSHED. The tree consists of major paths and local loops.

The major paths consist of three paths identified as Path I, Path II, and Path III. Path I designates the procedure to design the fill using soil alone. This path includes six steps including the data base for fill and underlying soils, numerical analysis of stability and settlement, conclusion of the path, graphical design of the embankment and construction procedure. Path II considers FGD by-products as a fill material instead of soil. This path adds two more steps of FGD materials to Path I. The steps are an interrogation of the data base for FGD by-products and an intermediate conclusion. Path III combines the above two paths to design the fill. On this path, the user will first consider soil as a fill material. If the fill is unstable or requires improvement of geometric conditions, the system recommends the use of better quality
materials like FGD by-products. In the following steps, the system proceeds to the final design process using FGD by-product as the fill material.

The local loops describe repeated processes within each path. These loops can repeat analysis processes without starting the consultation again when input errors are detected or when unsatisfactory results are encountered during the analysis process.

Several known cases were evaluated through three major paths by the experts. After the cases were run, the experts evaluated the system. During this validation process, the factors of primary interest in the validation process are completeness, efficiency, user-friendliness and usability.

The completeness refers to the thoroughness of the system and measures if the system can address all desired problems within its specified domain. The efficiency evaluates on how well the system makes use of the available knowledge, data, software, and time in solving problems within its problem domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data. User-friendliness is a measure of the ease of use of the system. Here evaluations are made of, among other features, on-screen help, illustrations, and explanations given to conclusions. The usability involves an evaluation of how the system might meet users' needs.
Table 6.1 and 6.2 show the results obtained for general evaluation of IDSSHED by the participating experts and non-participating experts, respectively. The expert graded the factors of completeness, efficiency, user-friendliness and usability on a scale of 0 to 4 with 4 representing very good and zero representing very poor.

Table 6.1 General Evaluation of IDSSHED by experts who were involved in the system development

<table>
<thead>
<tr>
<th>Measure</th>
<th>Expert 1 *</th>
<th>Expert 2 **</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>User-Friendliness</td>
<td>Very Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Usability</td>
<td>Good</td>
<td>Good</td>
</tr>
</tbody>
</table>

* Expert from ODOT  
** Expert from AEP

Table 6.2 General Evaluation of IDSSHED by experts who were not involved in the system development

<table>
<thead>
<tr>
<th>Measure</th>
<th>Expert 3*</th>
<th>Expert 4 **</th>
<th>Expert 5 ***</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very Good</td>
<td>Very Good</td>
<td>Good</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Good</td>
<td>Very Good</td>
<td>Good</td>
</tr>
<tr>
<td>User-Friendliness</td>
<td>Very Good</td>
<td>Good</td>
<td>Very Good</td>
</tr>
<tr>
<td>Usability</td>
<td>Very Good</td>
<td>Good</td>
<td>Very Good</td>
</tr>
</tbody>
</table>

* Expert from Federal Highway Administration, Washington D.C.  
** Expert from US Air Force  
*** Expert from Engineering Software of ENSOFT Inc., Texas
Figure 6.2 Validation tree
6.4 Case Study of An Implementation of the System

A case study was included in the evaluation of IDSSHED to demonstrate how the decision support system might work to design a highway embankment. In this illustration, a repair project performed on Ohio State Route 541 was selected as a case study of an implementation of the system.

6.4.1 State Route 541 Background

A section of the embankment of SR 541 located between Station 55 and 58 began to slip in early summer 1993. The map of site location is shown in Figure 6.3. Cracks in the embankment that were nearly parallel to the roadway appeared first. Then small slips occurred on the embankment slope. The small slips gradually grew and combined to form a failure surface. The Ohio Department of Transportation (ODOT) decided to repair the embankment. During repair ODOT work crews from the District 5 Special Project Branch discovered that a large amount of water from the upper hillside was trapped above an impermeable layer of shale, and that most likely this water entered the embankment saturating the bottom several feet of the embankment. The high water table reduced the effective stress of the embankment soil most likely leading to the observed embankment failure.
According to the drawings provided by ODOT, the height of the embankment beneath the existing roadway was 10.7 m. The thickness of the soil layer beneath the embankment was approximately 2.7 m at the center of the embankment. Shale underlay this soil layer. The slope was designed at 2 horizontal to 1 vertical to accommodate limited right-of-way. Figure 6.4 illustrates the boundary conditions of the rebuilt embankment section. Approximately 7,000 m$^3$ of compacted borrow material was needed to reconstruct the embankment.

### 6.4.2 Soil Properties

The properties of the embankment fill and foundation materials were obtained from ODOT. According to the results of Atterberg limits and particle-size distribution tests conducted by the author, the soil for embankment and foundation was classified as A-6a (silt and clay). The representative properties of the soil are summarized in Table 6.3.

#### Table 6.3 Properties of Soil (Undrained test)

<table>
<thead>
<tr>
<th>Plastic Index (PI)</th>
<th>Soil Type</th>
<th>Total Unit Weight (g/cm$^3$)</th>
<th>Saturated Unit Weight (g/cm$^3$)</th>
<th>Cohesion (kg/cm$^2$)</th>
<th>Friction</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>A-6a</td>
<td>1.87</td>
<td>2.0</td>
<td>0.50</td>
<td>0.0</td>
</tr>
</tbody>
</table>
6.4.3 Design of Reconstruction Embankment

The failure mechanism described above was considered in choosing the remedial measures. The field engineer consulted IDSSHED to identify appropriate remedial measures for the given site conditions. In the system, stability and settlement were evaluated using existing soil properties. The results were used to support user's decision for embankment reconstruction. This system recommended a method for increasing embankment stability by placing stiff layers of FGD by-products within the soil embankment. The reinforcement was shown to increase substantially the factor of safety on potentially critical slip surfaces.

To insert stiff layers into the embankment required the properties of high strength, low permeability, and non-swelling potential. The data base subsystem first filtered all of the records in the data base and found several FGD materials having those material properties. Among the materials selected, the FGD by-product generated at the Tidd power plant in Brilliant, Ohio was chosen for the fill material. The material properties obtained from system data base are summarized in Table 6.4.
Table 6.4 Properties of Tidd FGD by-product

- **Leachate Properties (mg/L)**
  - pH: 10.5
  - Arsenic: 0.03
  - Barium: 0.16
  - Cadmium: 0.0017
  - Chromium: 0.014
  - Lead: 0.0317
  - Mercury: 0.0135
  - Selenium: 0.1033
  - Silver: 2.1685

- **Chemical Properties (%)**
  - Fly ash: 21.0
  - CaCO₃: 13.1
  - CaO: 3.9
  - Ca(OH)₂: 0.8
  - CaSO₄: 27.4
  - CaMg₂: 18.8
  - MgO: 14.8

- **Engineering Properties**
  - W₀nt (%): 22.8
  - Dry Unit Weight (g/cm³): 1.68
  - Cohesion (kg/cm²): 46.4
  - Permeability (cm/sec): 4.3x10⁻⁷
  - Swell potential (%): 0.9

The system next compared the EPA leachate limits with test results obtained for the Tidd material to determine the potential for ground water contamination. This material was categorized as nonhazardous since chemical analyses of the leachate showed all concentrations of any substance listed in the EPA criteria to be well below EPA threshold values. The Tidd material has a high unconfined compressive strength of 92.8kg/cm², a very low permeability of 4.3x10⁻⁷ cm/sec, and non-swelling potential at 28 days after
compaction. Then the system analyzed slope stability and settlement of the embankment. The program created the input files for two FORTRAN programs, PCSTABL6 and SETFGD. The safety factor at the end of construction and the relationship of settlement to elapsed time are determined in the program. During these steps, the system can load default values as design parameters from the data base.

After determining the design requirements, the user can activate the program for detailed design of the embankment. This was done and the system iterated to determine the size and location of the layers in the embankment. The size was adjusted within the limited right-of-way until the factor of safety was greater than 1.5. The design dimensions of the embankment are presented in Figure 6.5. The FGD-reinforced embankment shows that the side slope is stable at a ratio of 2 horizontal to 1 vertical instead of the more common requirement of 4 : 1. In addition, total settlement of the underlying soil is predicted to be approximately 37 cm in 306 days and 79 % consolidation will finish within the construction period.

The site required good drainage conditions at the bottom of the embankment to prevent saturation of the bottom layer. A geotextile was installed to improve the drainage conditions. All of the processes conducted in the system are attached in Appendix F.
In this case study, the system was evaluated by the field engineers, who were not involved in the development of the system. The validation using this case study proceeded the same way as the process described in the previous section. The same scale to grade the factors was used and the results are shown in Table 6.5.

Table 6.5 Validation results by field engineers

<table>
<thead>
<tr>
<th>Measure</th>
<th>Very Good(4)</th>
<th>Good (3)</th>
<th>Fair (2)</th>
<th>Poor (1)</th>
<th>Very Poor (0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Efficiency</td>
<td></td>
<td>√</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User-friendliness</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usability</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>√</td>
</tr>
</tbody>
</table>

* Field Engineers of District 5, ODOT
Figure 6.3 Map of the construction site
Figure 6.4 Site conditions at failure
Figure 6.5 Design of embankment by IDSSHED
CHAPTER VII
DISCUSSION AND CONCLUSION

7.1 General

The beneficial utilization of FGD wastes has not received the level of attention from highway departments and utility companies that it should, given the huge amount of material presently being generated. This is due in large measure to the lack of knowledge of the properties of the material, limited knowledge of the behavior of the material in engineered structures, and the absence of procedures to guide engineers to safe, economical designs. The system outlined in this paper attempts to assist the highway designer by:

(1) codifying the important engineering properties needed in a rational design procedure of embankments;

(2) providing a step by step analysis and design procedure that takes advantage of the experience of a number of people in the fields of material characterization, highway design and construction, and geotechnical engineering. Through these efforts, this research encourages
the utilization of FGD by-product by providing the technical information necessary to incorporate FGD by-product into various highway applications.

To achieve these purposes, a computer-aided system IDSSHED was proposed. The system makes possible the efficient implementation of decision making tasks from the geotechnical engineering field in the design of structural fills utilizing coal combustion by-products. This system is a user-friendly computer program which encodes expert experiences, heuristic judgments, and numerical calculations obtained from analytical programs. The interactive response time is quick and editing of input data is easy. The system also summarizes large volumes of data and transfers visible data format using graphical and tabular data preparation to users. Thus, the system enhances the productivity of the decision maker.

The discussions in the next sections focus on general issues in the design of decision support system for embankment design problems.

7.2 Design of Fill Using FGD By-Products

In this research, standard ODOT design procedures and engineered design procedures were discussed. Most highway agencies specify density of the compacted fill and follows the AASHTO standard slope design. The given standard requires too large factor of safety in slope design. In the
case of problem foundations, limited right-of-way, and high embankment, the design specification listed in the standard design are not adequate. The engineered design procedures added stability and settlement analyses to the standard design procedures.

Stability and settlement of an embankment often depend on the weight of embankment material. A lightweight embankment fill can reduce the settlement and increase stability. PCSTABL6 was adopted to analyze the slope stability. SETFGD was developed to calculate settlement. These two numerical analyses have required a knowledge of geotechnical engineering and computing. Field engineers have expected to be able to use the programs easily.

In this research, FGD by-products were selected as lightweight fill materials. FGD waste has a relatively low unit weight (average = 1.1 g/cm³) and good shear strengths (average Cu = 15.4 kg/cm²) making it a desirable fill material. In addition, the FGD materials are much less compressible than typical natural soils. Sideslope steepening and settlement reducing could be obtained more easily the use of FGD by-products.

7.3 Knowledge Acquisition and Representation

The knowledge required to achieve the objectives of the research was gathered from documents, the author, and engineers from ODOT, AEP, and OEPA. The knowledge was
scattered in a variety of fashions. This research developed a systematic process to collect the scattered knowledge. The proposed knowledge acquisition process consisted of unstructured phase, structured phase, and organized phase. The proposed process lead to refinement of the knowledge and elimination of redundancies and events.

OOP paradigm was selected as knowledge representation method. OOP approaches are based on objects. A set of objects is called a class. Objects and classes provide a computable representation of the physical world that is convenient for engineers.

7.4 Development of IDSSHED

The knowledge gathered by the knowledge acquisition process consists of subtasks with many independent or semi-independent sources, interacting to find a common solution. For example, the knowledge of embankment design requirements are obtained from experts and slope stability can be analyzed by a numerical program. Therefore, the knowledge consists of domain knowledge and process knowledge. The domain knowledge was managed using an expert system. The process knowledge was performed using a decision support system. Some solutions depend on the specific preferences of an individual or group of decision makers. The expert systems approach is not able to produce the general representation for the beliefs of these decision makers. On
the other hand, a decision support system directly supports the decision makers in order to enhance their professional judgment.

In this research, an intelligent decision support system was proposed to integrate the components of both a decision support system and an expert system. The basic idea of this approach is that expert system can be added onto a DSS to extend its capabilities for performing functions that the regular DSS can not perform such as heuristic judgment. The proposed model consists of six components: knowledge base, inference engine, user interface, data base subsystem, numerical processing subsystem, graphical design subsystem. KPWIN was selected as the operating system.

The knowledge base stores the gathered knowledge in the form of objects and production rules. Objects are physical or logical entities. Rules are IF-THEN statements. The inference method was performed by the inheritance of the objects. The user interface was designed to simplify data entry and function choices and to transfer the computer outputs to the user effectively. A menu-driven interface with mouse was developed. Hypertext and hyperregion functions were used to send and receive information through texts and graphics. To enhance the system performance, large text, data, and graphic files were stored as external files.

Three subsystems were used to enhance the quality of decision making. A data base subsystem was built for FGD by-
products and natural soils. The data base in the system provided the user with typical ranges of the material properties and with default values to be used as needed in the numerical calculations. A relational data model was used to define the relationships among records. The data base subsystem was integrated into the shell under the same transaction control mechanism. The average and the standard deviation were automatically assigned to topics accessible as the default values.

The numerical processing subsystem was primarily concerned with the stability and settlement calculations necessary to evaluate the behavior of the embankment and foundation materials. The system adopted two FORTRAN programs PCSTABL6 for slope stability analysis and SETFGD for settlement. The host program (KPWIN) communicated with the FORTRAN programs through sequential files generated by both programs. The shell captured the parameters in the files and assigned them to the appropriate topics in the knowledge base.

The interactive graphical design subsystem was used not just for evaluating the material properties but also for the synthesis tasks of numerical calculations. This subsystem communicates with data base and numerical processing subsystems through the shell. The subsystem summarized large volumes of numerical data obtained from both subsystems and visualizes the trends in the data.
The proposed model is a user-friendly computer program which integrates expert experiences, numerical calculations, data base and graphics within single package. The system can improve the efficient implementation of decision making tasks in the geotechnical engineering field. Thus, the system enhances the productivity of the decision maker.

7.5 Validations of IDSSHED

In this research, three levels of quality control procedures were conducted: testing, validation, and case study of an implementation.

The first procedure, testing, examined whether or not the system is functioning as intended. This process checked for decision procedure, system linking with subsystems, error analysis, and system maximization. This procedure was conducted by the author and his advisor.

The second procedure, validation, was conducted by area experts who were involved in the system development as well as experts who were not involved in development. The validation concerned a diagnosis of how closely the IDSSHED's decision supports a real solution. The validation was performed using a validation tree which consists of three major paths and six local loops. In the third procedure the program was applied to a case study to evaluate how IDSSHED would work in the design process for a highway embankment. A repair project performed on Ohio State
Route 541 was selected. The system performance was evaluated by the field engineer responsible for the design.

The IDSSHED was validated by seven experts from three different areas. In validation, completeness, efficiency, user-friendliness, and usability of IDSSHED were evaluated. The overall averages of each factors are:

Completeness : between very good and good (3.6),
Efficiency : between very good and good (3.3),
User-friendliness : between very good and good (3.9),
Usability : between very good and good (3.3), and
Overall average : between very good and good (3.5).

7.6 Future Study

This research applied a knowledge-based decision support system to highway embankment design using FGD by-products. In this application, the system has limitations in building data base and system integrations. A number of changes or extensions to IDSSHED would make it more complete and more usable. Several suggestions for improvements or potential extensions in future study are listed below:

• build a more complete data base for FGD by-products because the data base includes limited number of samples on dry FGD by-products.
• expand the data base for soils which can provide accurate engineering properties according to the site conditions.
• refine the program SETFGD for settlement calculation
• upgrade the package for the interactive graphical design (for example, CAD)

• provide easy-to-learn tutorials

• include foundation treatment methods in the system.

• add cost analysis of embankment design.

• upgrade shell limitations by adding DLLs of user-defined functions to the shell.

The program as it exists has been shown to be a useful tool which, for the first time, provides the highway engineer with the information and procedures necessary to design safe, economical structures using the by-products of clean coal technology. It also provides the regulators with a way of checking the environmental impact of this type of construction. Finally, by combining the knowledge currently available on these types of materials in an analysis and design package that allows easy updating, the research described in this thesis is a significant contribution to the goal of reuse of a material resource previously considered as waste product.
LIST OF REFERENCES

AASHTO, A policy on geometric design of highways and streets, American Association of State Highway and Transportation Officials, 1984.

AASHTO, AASHTO guide for design of pavement structures, American Association of State Highway and Transportation Officials, 1986.


Boutrup, Eva, "computerized slope stability analysis for Indiana highways", Joint Highway Research Project No 77-25 and 77-26, School of Civil Engineering, Purdue University, W. Lafayette, Indiana, 1977.


EPRI, Advanced SO$_2$ Control By-Product Utilization: Furnace sorbent injection wastes, Palo Alto, California, Electric Power Research Institute, EPRI CS-5783, 1988b.

EPRI, Advanced SO$_2$ Control By-Product Utilization: Laboratory Evaluation, Palo Alto, California, Electric Power Research Institute, EPRI CS-5783, 1988c.


Jain, Deepak, et.al., Knowledge representation with logic, Expert Systems for Civil Engineers: Knowledge Representation, ASCE, 1992.

Kao, Jehng-Jung, Computer aided systems for environmental engineering decision making, Ph.D. Dissertation, University of Illinois, Urbana_Champaign, 1990


Nolan, P.S. and R.V. Hendricks, "Initial test results of the limestone injection multistage burner (LIMB) demonstration project," Presented at the 81st annual meeting of the Air Pollution Control Association in Dallas, Texas, June 20-24, 1988b.

PETC, "Beverly duct injection test facility results," Duct injection development program, Department of Energy-Pittsburgh Energy Technology Center, Technical Update No. 16, October 1990, pp. 1-5.


Stein, H., "Best Way to Handle Waste is to Minimize it", Electric Light and Power 69(9), 1991, pp. 32-34


AL. STRUCTURE OF IDSSHED

This system consists of four parts which are stored in separate directories:

(1) Data base alone (IDSSDB)
(2) Stability analyses alone (STABLE_W)
(3) Settlement analyses alone (SETTLE_W)
(4) Embankment design consultation (EMDKB)

The first three parts are embedded into the last part. Each part will be helpful to consult the specified fields. The following Figure A1 shows the program structure of IDSSHED.

![Program structure of IDSSHED](image)

**Figure A.1 Program structure of IDSSHED**

A1.1 Data Base

This branch manages the data bases for the embankment and foundation materials in the system. The structure of the data base is shown in Figure 5.1 of Chapter 5. The data base (IDSSDB) consists of FGDDDB and SOILDB. FGDDDB manipulates the
data for FGD by-products used as embankment material. SOILDB evaluates the data for fill soil and underlying soils of embankment foundation. Figure A.2 and A.3 show structure of FGDDb and SOILDB.

Figure A.2 Structure of FGDDB

Figure A.3 Structure of SOILDB
A1.2 Numerical Analyses

Figure A.4 Structure of STABLE_W

Figure A.5 Structure of SETTLE_W
A1.3 Embankment Design Consultation (EMBKB)

Figure A.6 Structure of EMBKB

A2. INSTALLATION

The programs in IDSSHED are written for IBM-PC compatibles with 386 and 486 processors. A color monitor with VGA capabilities (or better) is required. The installation on a hard disk requires 6 MB of storage and 4MB of RAM.

IDSSHED system was created by integrating the software of
KPWIN, FORTRAN, Visual Basic, Access, and Windows. This system was compiled to distribution system by using the utility KPWIN Runtime. The source codes of the system were written by Knowledge Pro version 2.35 for Windows. Numerical programs PCSTABL6 and SETFGD were written by FORTRAN, database was built by Access, and Interactive graphics was drawn using Visual Basic.

Windows is the only one required to execute the distribution system. To install the distribution system of IDSSHED in the hard drive of the computer, create a subdirectory called IDSSHED and copy all the files in the subdirectory. To run the files, a button has to be created on Windows. There are two steps to create this button: creating program group name and program item as shown in Figure A.7. To create the program group name in Windows, choose New from the File Menu by using Program Manager and select the Program Group option, then the Program Group properties dialog box appears. Type a content for the group as shown in Figure A.8. After creating a program group name, create a Program Item for the application. To create the Program Item, open the program group again that you created to add the item. Select the Program Item option and the Program Item properties dialog box appears. Type the contents in each box as shown in Figure A.9. After typing the contents in the boxes choose the OK button. The dialog box closes and the program item appears in the group. When
you choose the button of the Program Item, Windows starts the specified application. For more information about creating a button of Program Item for an application, see "Program Manager" in Windows Manual [MS, 1991].

The program IDSSHED consists of six source programs which have different extensions (.kb, .ckb, .txt, .bmp, .exe, and .dbf). In addition, four more programs are required to run the program in runtime (kpdbbr.dll, kpmathr.dll and kpwinrun.exe from KPWIN and vbrun200.dll from Visual Basic). The source programs of .kb are not necessary to execute the distribution system.

(1) .kb files: source codes written by KPWIN; these files are not necessary for distribution system,

(2) .ckb files: compiled source codes of .kb files, (disk #2)

(3) .txt files: text files for help statements written by KPWIN or Notepad in Windows, (disk #3)

(4) .bmp files: graphics files to show the information and to receive the information by the user, (disk #4 and #5)

(5) .exe files: execution files created by FORTRAN and Visual Basic programs, (disk #3)

(6) .dbf files: data files for FGD by-products and underlying soils, (disk #3)

(7) kpdbbr.dll, kpmathr.dll and kpwinrun.exe: files for runtime obtained from KPWIN, (disk #1) and

(8) vbrun200.dll: file for runtime obtained from Visual Basic. (disk #1)

To install the distribution system in your computer, Groups (2) through (7) are stored in the subdirectory (C:\IDSSHED\). Group (8) is stored in the subdirectory of
Windows (C:\Windows\System\vbrun200.dll) when the Visual Basic was not installed in the computer.

Figure A.7 New program object

Program Group Properties

- Description: Program group for IDSSHED
- Group File: IDSSHED

Figure A.8 Program group properties

Program Item Properties

- Description: IDSSHED
- Command Line: kpwinrun.exe main.ckb
- Working Directory: C:\idsshed
- Shortcut Key: None
- Run Minimized

Figure A.9 Program item properties
APPENDIX B

SOURCE CODES FOR IDSSHED
(*============================================
Main Program

Developed by
Sung Hwan Kim
Dr. William E. Wolfe
4/5/94
Main.kb

==============================================*)

mainsetup( ).

bdll is [ ].
wtxxl is [ ].
Lmxxl is [ ].
Spxl is [ ].
keyvalue is 0.
key_ssettle is 0.
key_sstable is 0.
key_settle is 0.
key_stable is 0.
keyspec is 0.
fdsoll is [ ].
ocrl is [ ].
OFOS = 0.
seratio = 0.
sestot = 0.

use_font (?bigFont).
bitmap(?box5,3,2).
text(' #x7#y2.5 #fblue GEOTECHNICAL DESIGN MODEL #d').
use_font(?boldfont).
text('#x20#y5 ( IDSSHED )').
bitmap(?box8,6,8.5).
bitmap(load_bitmap ('title.bmp'),14,8.5).
button2('Press Here To Continue =>>',continue,
    55,26,30,1.5).
wait( ).

use_font( ).
text('#e').
sample_win is window(, 20,9,45,8,' Practice to close
    Window',[thinframe,showchildren,visible],?main, ,
    lightgray,close_event).
button2('Close Me',continue,1,1,11,1.3).
window (,1,2.3,45,6.5,, [child,siblings,visible,
    showchildren],?sample_win ).
text('#e
A Window with the button #fred Close Me #d should
be closed when you leave the Window.
Do practice now.').
wait( ).
close_window( ).
close_window( ).
Generally, the Design of Embankment is developed through following stages (NCHRP Synthesis 33): (1) Corridor Study, (2) Route Selection, (3) Preliminary Design, (4) Final Design, and (5) Construction. In this program, Geometric conditions (height of embankment, right-of-way width) are provided from stages (1) and (2). Foundation conditions (depth, underlying soil types, water table) are obtained from the preliminary stage. As an embankment fill material, FGD by-products is selected. Data base incorporated into this system supplies the required information for the embankment and foundation materials. Based on the above information, the engineering behavior of embankment (slope stability and settlement) is predicted. The design of the fill begins considering the Standard Specification, Geometric conditions, and the results of numerical analysis. Finally, construction procedures for the embankment are listed.'
button2('Press Here To Continue =>>',continue,55, 26,30,1.5,,mouse_up_event).
wait( ).
    project is get_text(?tit).
    date is get_text(?dat).
    user_name is get_text(?nam).
    Position is get_text(?posi).
    company is get_text(?comp).

text('#e').
bitmap( load_bitmap('emtitle.bmp'),15,4).
button2('Press Here To Continue =>>',continue,55, 26,30,1.5,,mouse_up_event).
wait( ).

text('#e#x3#y1.2 #fbrown Site Information Obtained from Corridor Study and Route Selection #d').
use_font( ).
bitmap(?box7,2.5,3).
bitmap(load_bitmap('emshape.bmp'),5,15).
text('#x5#y4 o Geometric Conditions by Proposed Grade Line
    #x7#y5.5 - Slope of Embankment (n):
    #x7#y7 - Height of Embankment (ft) (b):
    #x7#y8.5 - Depth of Foundation (ft) (c):
    #x7#y10 - Half Width of Top (ft) (a):
    #x7#y11.5 - Width of Right-Of-Way (ft)').

shit = concat(30,,0).
srat = concat(4,,0).
sdep = concat(20,,0).
stop = concat(50,,0).
srow = concat(200,,0).

st1 is set_focus( edit_line (?srat,,38,5.3,7)).
st2 is edit_line(?shit,,33,6.8,7).
st3 is edit_line(?sdep,,38,8.3,7).
st4 is edit_line(?stop,,38,9.8,7).
st5 is edit_line(?srow,,38,11.3,7).

button2('Press Here To Continue =>>',continue,55, 26,30,1.5,,mouse_up_event).
wait( ).
    ratio is get_text(?stl).
    height is get_text(?st2).
    depth is get_text(?st3).
    width is get_text(?st4).
    row is get_text(?st5).

use_font(?boldfont).
text('#e #fbrown Embankment Materials #d').
use_font( ).
text('#x5#y3 Please select a Material type for the Embankment Fill.').
check_box('Natural Soil', soil_all, 10, 4.5).
check_box('FGD By-products', fgd_all, 10, 6.5).
bitmap(box6, 5, 12).
bitmap(load_bitmap('dnflow.bmp'), 6, 13).

**topic soil_all.**

#include sfound.ckb
#include sstab.ckb
#include ssettle.ckb
#include soilcon.ckb
#include sdesign.ckb

end.

**topic fgd_all.**

#include dbase.ckb
#include dbpropt.ckb
#include default.ckb
#include property.ckb
#include found.ckb
#include stab.ckb
#include Settle.ckb
#include fgdcon.ckb
#include fdesign.ckb

end.

**topic mark(item).**

intwin is window(, 45, 12, 46, 14,'Explanation of Embankment Design Stages', [thinframe, showchildren, visible], ?main, , lightgray, close_event).
button2('Close Me', close, 1, 1, 11, 1.3).
inttwin is window (, 1, 2.3, 46, 11.5, [child, siblings, visible, showchildren], ?intwin).
set_file_pos('introd.txt', 0, beginning).
introd is read('introd.txt', concat('//', ?item), '//').

text('#e', Pintrod).

end.

**topic close.**

close_window(?intwin).

end.

**topic mainsetup.**

main_title is 'IDSSHED - CONSULTATION'.
main is window(, 1, 1, 91, 30, ?main_title, [thickframe, visible, controlmenu, maximizebox, minimizebox, showchildren] , , , close_event).

menu([[&File, &Save, &Print, &Close],[&IDSSHED, &About Program], &IDSSHED, &KB_Structure,
&Acknowledgements],[['&Data Base','&DB Structure','&FGD General','&FGD Chemical','&FGD Engineering',
'&Foundation Soil']],[&Help, '&Terminology',
'&References']],[menutopic].

boldFont is create_char_font ( [1.2,1.2,700, 'F','F','F',0,1,34,'Helv']).

hyperFont is create_char_font ( [1,1,500,'F','T','F',0,1,34,'Helv']).

boldFont is create_char_font ( [1.3,1.2,700,'F','F','F',0,1,34,'Helv']).

bigFont is create_char_font ( [2,1.5857,700,'F','F','F',0,1,34,'Helv']).

mainFont is create_char_font ( [1,1,400,'F','F','F',0,1,34,'Helv]).

hypcolor is green2.

hyper_display (?hypcolor,,?hyperFont).

box1 is load_bitmap('box1.bmp').
box2 is load_bitmap('box2.bmp').
box3 is load_bitmap('box3.bmp').
box4 is load_bitmap('box4.bmp').
box5 is load_bitmap('box5.bmp').
box6 is load_bitmap('box6.bmp').
box7 is load_bitmap('box7.bmp').
box8 is load_bitmap('box8.bmp').
box9 is load_bitmap('box9.bmp').
box10 is load_bitmap('box10.bmp').
box11 is load_bitmap('box11.bmp').
box12 is load_bitmap('box12.bmp').
box13 is load_bitmap('box13.bmp').
box14 is load_bitmap('box14.bmp').
box15 is load_bitmap('box15.bmp').
box16 is load_bitmap('box16.bmp').
box17 is load_bitmap('box17.bmp').
box18 is load_bitmap('box18.bmp').
box19 is load_bitmap('box19.bmp').

ContentsList is' IDSSHED Help Index

The index lists help topics available to IDSSHED.
Use the scroll bar to see entries not currently visible.

Button Functions Used in IDSSHED:
Coal Combustion By-Product:
#mFly Ash#m
#mBottom Ash#m
#mFGD By-Product#m
#mWet System#m
#mDry System#m
- #mSpray Dryer#m (SPD)
- #mLime Injection Multistage Bunner#m (LIMB)
- #mDuct Injection#m (DUCT)
- #mSOx-NOx-ROx BOx#m (SNRB)
- #mCoolside#m (CLS)
- #mLimestone Injection Dry Scrubbing## (LIDS)
- #mAtmospheric Fluidized Bed Combustion#m (AFBC)
- #mPressurized Fluidized Bed Combustion#m (PFBC)
- #mRotary Cascading Bed Boiler#m (RCBB)

Embarkment Design:
#mHighway Embankment#m
#mFoundation#
#mSlope Angle#
#mUnderlying Soil#

Intelligent Decision Support System:
#mAlgorithmic Program#
#mAi#
#mDecision Support System#
#mIntelligence Decision Support System#
#mExpert System#
#mKnowledge Base#
#mInference Engine#
#mObject-Oriented Programming#
#mSystem Shell#

ReferenceList is 'fbrown References #d'

#mData Base#
#mDecision Support System#
#mExpert System#
#mEmbarkment Design#
#mFGD By-Property#
#mObject Oriented Programming#

```
topic menutopic(item).
   do(?item).

topic &Save.
   write('current.top').
end.

topic &Close.
   clear( ).
end.

topic '&About Program'.
   prowin is window(, 25,4.5,60,25, 'System Introduction',
   [thinframe,showchildren,visible],?main, ,lightgray,
   close_event).
   button2('Close Me',proclose,1,1,11,1.3).
```
IDSSHED is a complete design model for the Highway Embankment Application using FGD By-products.

Scope
Application Field: Highway Embankment.
Materials Used: Natural Soils, Dry FGD By-products.
Analysis Method: Decision Support System.

Computer-Integrated Design Model
System Shell: Knowledge Pro 2.3 for Windows
DBMS: DBaseIV or MS Access
Numerical Analysis: MS FORTRAN 5.1
Computer Aided Drawing: Visual Basic

If you are uncertain of the questions in this system, you may press or click the Hyper Text for a help statement.

end.
This system integrates a decision support system with expert system. The following Figure shows the Integrated Design Model for IDSSHED.'

bitmap(load_bitmap('model.bmp'),8,10).

hyper_region(numhyper,28,17,9,3).

hyper_region(dbhyper,19,20,9,3).

topic idsshed.
helpwin is window(, 10,8,75,16, 'IDSSHED Structure', [thinframe,showchildren,visible],?main, ,lightgray, close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2,3,75,15,, [child,siblings,visible, showchildren],?helpwin).
text('#x3#y1.2 A Schematic Diagram for the KB').
bitmap(load_bitmap('dnflow.bmp'),2,3).

end.

topic dbhyper.
helpwin is window(, 25,5,45,21.5, 'IDSSHED Help', [thinframe,showchildren,visible],?main, ,lightgray, close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2,3,45,20,, [child,siblings,visible, showchildren],?helpwin).
bitmap(load_bitmap('stable6.bmp'),2,1.2).

end.
bitmap(load_bitmap('dbhir.bmp'),2,1.2).
end.
end.(*idsshed*)
end.(*&IDSSHED*)

topic &KB_Structure.
kbwin is window(, 40,2.5,50,27, 'KB Structure',
    [thinframe,showchildren,visible],?main, ,lightgray,
    close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,24.5,, [child,siblings,visible, showchildren],?kbwin ).

use_font( )
text('#e KB Structure').
bitmap(load_bitmap('class.bmp'),3,3).
end.

topic &Acknowledgements.
helpwin is window(, 40,4,50,24, 'ACKNOWLEDGEMENTS',
    [thinframe,showchildren,visible],?main,
    ,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,21.5,, [child,siblings,vertscroll, visible, showchildren],?helpwin ).
use_font( )
text(
    #fbrown ACKNOWLEDGEMENTS #d

    #fgreen The program was developed by #d
    Sung-Hwan Kim
    William E. Wolfe
    Fabian C. Hadipriono,
    Tien H Wu.
    The work described in this system forms part
    of a project titled "Land Application Uses of Dry
    FGD By-Product," being performed in the Civil
    Engineering Department at The Ohio State Univ-
    ersity.

    #fgreen The sponsors are #d
    Dravo Lime Co
    Ohio Coal Development Office
    U.S. Department of Energy,
    Morgantown Energy Technology Center
    American Electric Power
    Ohio Edison

    #fgreen The first author was partially supported by #d
    Korea Highway Corporation, Seoul, Korea.

    #fgreen The experts cooperated in the system were #d
    Eugene C. Geiger, ODOT


end.

topic ' &DB Structure'.
helpwin is window(, 35,5,52,21.5, 'Data Base',
[thinlineframe,showchildren,visible],?main, ,lightgray,
close event).
button2 (FGD,fgdt,1,1,11,1.3 ).
button2 (Soil,soilt,12,1,11,1.3 ).
button2 ('Close Me',close,23,1,11,1.3).
window (,1,2.3,52,19,,
[child,siblings,visible,showchildren],?helpwin ).
bitmap(load_bitmap('dbtitle2.bmp'),2,2).

topic fgd.
helpwin is window(, 25,7.5,50,22, 'Data Base for FGD By-
Products',[thinlineframe,showchildren,visible],?main,
,lightgray,close_event).
button2 ('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,19.5,, [child,siblings,visible,
showchildren],?helpwin ).
bitmap(load_bitmap('fgddb.bmp'),4,2).
end.(*dbhir*)

topic soil.
helpwin is window(, 25,7.5,50,22, 'Data Base for Soil',
[thinlineframe,showchildren,visible],?main, ,lightgray,
close_event).
button2 ('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,19.5,, [child,siblings,visible,
showchildren],?helpwin ).
bitmap(load_bitmap('soildb.bmp'),4,2).
end.(*dbhir*)
end.(*structure*)

topic '&FGD General'.
helpwin is window(, 25,5,65,21.5, 'Energy Production',
[thinlineframe,showchildren,visible],?main, ,lightgray,
close_event).
button2 ('More',fgdgen,1,1,11,1.3 ).
button2 ('Close Me',close,12,1,11,1.3).
window (,1,2.3,65,19,, [child,siblings,visible,
showchildren],?helpwin ).
use_font(?boldfont).
text('#e #x9#yl.5 Energy Production by Energy Source').
energy is load_bitmap('energy.bmp').
prouse is load_bitmap('prouse.bmp').
bitmap(energy,1,2.5).

topic fgdgen.
helpwin is window(, 20,8,65,22, 'Production & Use of Coal Combustion Wastes', [thinframe,showchildren, visible],?main, ,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).

window (,1,2.3,65,19.5,, [child,siblings,visible, showchildren],?helpwin ).

text('#e #yl.5 Production & Use of Coal Combustion By-Products').

use_font( ) .

bitmap(?prouse,2,2.5).
end.(*fgdgen*)
end.


topic ' &FGD Chemical'.

helpwin is window(, 25,5,65,21.5, 'FGD Chemical', [thinframe,showchildren,visible],?main, ,lightgray,close_event).
button2 ('More',continue,1,1,11,1.3 ).
button2('Close Me',close,12,1,11,1.3).

window (,1,2.3,65,19,, [child,siblings,visible, showchildren],?helpwin ).

topic ' &FGD Engineering'.

helpwin is window(, 25,5,65,21.5, 'FGD Engineering', [thinframe,showchildren,visible],?main, ,lightgray,close_event).
button2 ('More',continue,1,1,11,1.3 ).
button2('Close Me',close,12,1,11,1.3).

window (,1,2.3,65,19,, [child,siblings,visible, showchildren],?helpwin ).

topic ' &Foundation Soil'.

helpwin is window(, 25,5,65,21.5, 'Foundation Soil', [thinframe,showchildren,visible],?main, ,lightgray,close_event).
button2 ('More',continue,1,1,11,1.3 ).
button2('Close Me',close,12,1,11,1.3).

window (,1,2.3,65,19,, [child,siblings,visible, showchildren],?helpwin ).

text('#e This is Foundation Soil').

end.


topic ' &Terminology'.

helpwin is window(, 35,2.5,55,27, 'IDSSHED Help', [thinframe,showchildren,visible],?main, ,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
subwin is window (,1,2.3,55,24.5, [vertscroll,child, siblings,visible,showchildren],?helpwin ).

use_font( ).
text('#e',?ContentsList).
use_font( ).

button2(Continue,contin,5,8.3,11,1.4).
button2('Close Me',conme,5,9.8,11,1.4).
button2(Graph,graph,5,11.3,11,1.4).
button2('Press Here To Continue',press,5,12.8,23,1.4).
button2('Table',table,5,14.3,11,1.4).

helpwin is window(, 20,10,50,18, ?item,
[thinframe,showchildren,visible],?main, , lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,16, [child,siblings,vertscroll, visible,showchildren],?helpwin ).
set_file_pos('help.txt',0,beginning).
helpctxt is read('help.txt',concat('//',?item), '//').
text('#e',?helptxt).
end.

helpwin is window(),20,10,50,18, ?item,
[thinframe,showchildren,visible],?main, , lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,16, [child,siblings,vertscroll, visible,showchildren], ?helpwin ).

helpwin is window (), 10,10,50,5, [thinframe,popup, child, showchildren,visible],?main, , lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,5, [child,siblings,vertscroll, visible,showchildren], ?helpwin ).

helpwin is window (), 10,10,50,8, [thinframe,popup, child, showchildren,visible],?main, ,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,8, [child,siblings,vertscroll, visible,showchildren], ?helpwin ).

helpwin is window (), 10,10,50,6, [thinframe,popup, child, showchildren,visible],?main, ,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,6, [child,siblings,vertscroll, visible,showchildren],?helpwin ).

IDSSHED requires to close any screen which has #fgreen2 the Close Me Button#d.'

end.

helpwin is window (), 10,10,50,8, [thinframe,popup, child, showchildren,visible],?main, , lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,8, [child,siblings,vertscroll, visible,showchildren], ?helpwin ).

helpwin is window (), 10,10,50,6, [thinframe,popup, child, showchildren,visible],?main, , lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,6, [child,siblings,vertscroll, visible,showchildren], ?helpwin ).
button2('Graph',15,1.2,11,1.4,char_event).
text('#x5#y3 This button shows its #fgreen2 Graphs #d.
end.

topic press.
helpwin is window(10,10,50,6,[thinframe,popup,child,
showchildren,visible],?main,,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window(1,2.3,50,6,[child,siblings,visible,
showchildren],?helpwin).
button2('Press Here To Continue',10,1.2,23,1.4,
char_event).
text('#x5#y3 This button shows #fgreen2 the Next Main
Screen#d. This button has same meaning as #fgreen2
Continue Button#d.
end.

topic table.
helpwin is window(10,10,50,6,[thinframe,popup,
child,showchildren,visible],?main,,lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window(1,2.3,50,6,[child,siblings,visible,
showchildren],?helpwin).
button2('Table',10,1.2,11,1.4,char_event).
text('#x5#y3 This button shows #fgreen2 its Table related
#d.
end.
end.(*terminology*)

topic 'References'.
helpwin is window(35,2.5,55,27,'IDSSHED Help',
[thinframe,showchildren,visible],?main,,
lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
subwin is window(1,2.3,55,24.5,[vertscroll,child,
siblings,visible,showchildren],?helpwin).

use_font(?boldfont).
text('e',?ReferenceList).
use_font( ).

topic mark(item).
helpwin is window(20,10,50,18,'IDSSHED Help',
[thinframe,showchildren,visible],?main,,
lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window(1,2.3,50,16,[child,siblings,visible,
showchildren],?helpwin).
set_file_pos('refern.txt',0,beginning).
referntxt is read('refern.txt',concat('//',?item),'//').
text('e',?referntxt).
end.
end.

    topic close.
    close_window( ).
    close_window( ).
    show_window( ).
end.
end. (*menutopic*)

end. (*mainsetup*)
Data Base for Soils
sfound.kb

new_file('soilprot.out').
@kpdblib.tpx
use_font(?boldfont).
text(:'#e #x3#y1#fbrown Data Base For Soils #d').
use_font(  ).
bitmap(?boxl0,45,2).

Text(:'#x46#y3 o. Soil Types by #mODOT Method#m
  #x47 A-1-a, A-1-b
  #x47 A-3, A-3a
  #x47 A-2-4, A-2-5, A-2-6, A-2-7
  #x47 A-4a, A-4b, A-5
  #x47 A-6a, A-6b
  #x47 A-7-5, A-7-6
  #x47 OL, OH, Pt
  #x49 #mUnified Method#m').

bitmap(?box11,6,15).
text(:'#x3#y3 o. Please input soil types for the layers
  #y4 (choose the type from the Table)
  #x7#y5.5 -. Soil Type for Embankment
  #x7#y9.5 -. Soil Type for Foundation
  #x7#yl6 Note:
  #x7 o. If more than two layers in Fill,
  #x7 type a dominant soil name in the box.
  #x7 o. If more than 2 layers in foundation,
  #x7 type a dominant soil name in the box.').

emsoiltp is set_focus(edit_line('A-3a',,10,7,10)).

ewin is button2('Continue =>>',continue,25,7,15,1.5).
wait(  ).
close_window(?ewin).

emsoil is get_text(?emsoiltp).
  if ?emsoil is 'A-1-a' or ?emsoil is 'A-1-b' or
    ?emsoil is 'A-3' or ?emsoil is 'A-3a' or
    ?emsoil is 'A-2-4' or ?emsoil is 'A-2-5' or
    ?emsoil is 'A-2-6' or ?emsoil is 'A-2-7' or
    ?emsoil is 'A-4a' or ?emsoil is 'A-4b' or
    ?emsoil is 'A-5' or ?emsoil is 'A-6a' or
    ?emsoil is 'A-6b' or ?emsoil is 'A-7-5' or
    ?emsoil is 'A-7-6' or ?emsoil is OL or
    ?emsoil is OH or ?emsoil is Pt then scode is ?emsoil
  else soil_errorl(  ).

filter is concat(scode,' = "',?scode,'"').
emsoilList is [stuw,ssuw,ucu,dcu,ufa,dfa].
db_use_file ('soildb.dbf').
db_set_filter (?filter).
rec is db_top_record (?emsoilList).
esoil gets(?rec).

estuw is element(?esoil,1).
estuw is element(?esoil,2).
estuw is element(?esoil,3).
estuw is element(?esoil,4).
estuw is element(?esoil,5).
estuw is element(?esoil,6).
db_remove_filter(?filter).

fdsoiltp is set_focus(edit_line ('A-5',5,10,11,10,10)).

button2('Press Here To Continue =>>',continue,55,26,30,1.5).

wait( ).

fdsoil is get_text(?fdsoiltp).

if ?fdsoil is 'A-1-a' or ?fdsoil is 'A-1-b' or
?fdsoil is 'A-3' or ?fdsoil is 'A-3a' or
?fdsoil is 'A-2-4' or ?fdsoil is 'A-2-5' or
?fdsoil is 'A-2-6' or ?fdsoil is 'A-2-7' or
?fdsoil is 'A-4a' or ?fdsoil is 'A-4b' or
?fdsoil is 'A-5' or ?fdsoil is 'A-6a' or
?fdsoil is 'A-6b' or ?fdsoil is 'A-7-5' or
?fdsoil is 'A-7-6' or ?fdsoil is OL or
?fdsoil is OH or ?fdsoil is Pt then fcode is ?fdsoil
else soil_error2( ).

filter is concat(scode,' = "',?fcode,'"').

fdsoilList is [stuw,ssuw,ucu,dcu,ufa,dfa,cv,cc,
    cr,ca,void].

db_use_file ('soildb.dbf').
db_set_filter (?filter).
rec is db_top_record (?fdsoilList).
soil gets(?rec).

fstuw is element(?soil,1).
fssuw is element(?soil,2).
fscu is element(?soil,3).
fsdcu is element(?soil,4).
fsufa is element(?soil,5).
fsdfa is element(?soil,6).
fscv is element(?soil,7).
fscr is element(?soil,8).
sorca is element(?soil,9).
sca is element(?soil,10).
svoid is element(?soil,11).
db_remove_filter(?filter).

topic soil_error1.

soilwin is window(, 40,12,50,9,, [dialogwindow,visible],
    ?main, ,). Text(' @ Make sure you type correct name of Soil Type.
See Table shown above again and retype it.'),
emsoiltp is set_focus(edit_line('A-4b',10,4,10)),
button2('Close Me',Continue,35,6.5,10,1.5),
wait( ).
scode is get_text(?emsoiltp).
close_window(?soilwin).
show_window( ).
end.

topic soil_error2.
soilwin is window(40,12,50,9,[dialogwindow,visible],
?main,,). Text(
@ Make sure you type correct name of Soil Type.
See Table shown above again and retype it.').
fdsoiltp is set_focus(edit_line('A-4b',10,4,10)),
button2('Close Me',Continue,35,6.5,10,1.5),
wait( ).
scode is get_text(?fdsoiltp).
close_window(?soilwin).
show_window( ).
end.

use_font(?boldFont).
text('#e fbrown Slope Stability Analysis #d'). use_font( ).
tesa is load_bitmap('box12.bmp').
bitmap(?tesa,5,10).
text ('#x5#y3.5 o. Please select A Type of Slope Stability Analysis
#x7#y11.2 Total Stress Analyses: is used to estimate the end-of-construction state with the undrained conditions. (c = Su and phi = 0)
Effective Stress Analyses: is used to estimate the long-term state with the drained conditions. (c = c and phi = phi)').
tieback is load_bitmap('tieback.bmp').
reinfor is load_bitmap('reinfor.bmp').
noneload is load_bitmap('noneload.bmp').
svalue is radio_button([['Total Stress Analyses',8,5,t],
 ['Effective Stress Analyses',8,6.5]]).
button2('Press Here To Continue =>>',continue,55,
26,30,1.5,move_up_event).
wait( ).
styp is stype(?svalue).
topic stype(handle).
typel is element(?handle,where(get_radio_button (?svalue),t)).
stype is get_text(?typel).
end.
if ?stype is 'Total Stress Analyses'
then esscu is ?esucu and essfa is ?esufa and
fsscu is ?fsucu and fssfa is ?fsufa.
if ?stype is 'Effective Stress Analyses'
then esscu is ?esdcu and essfa is ?esdfa and
fsscu is ?fsdcu and fssfa is ?fsdfa.

estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.

essuw = ?essuw*10.
essuw = ?essuw div 1.
essuw = ?essuw/10.

esscu = ?esscu*10.
esscu = ?esscu div 1.
esscu = ?esscu/10.

essfa = ?essfa*10.
essfa = ?essfa div 1.
essfa = ?essfa/10.

pestuw = ?estuw*10 mod 10.
peessuw = ?essuw*10 mod 10.
peesscu = ?esscu*10 mod 10.
peessfa = ?essfa*10 mod 10.

if ?pestuw is 0 then estuw is concat(?estuw,'.')
else estuw = ?estuw.

if ?pessuw is 0 then essuw is concat(?essuw,'.')
else essuw = ?essuw.

peesscu = ?esscu*10 mod 10.
if ?peesscu is 0 then esscu is concat(?esscu,'.')
else esscu = ?esscu.

peessfa = ?essfa*10 mod 10.
if ?peessfa is 0 then essfa is concat(?essfa,'.')
else essfa = ?essfa.

(* Initial Input for Foundation Soils *)
fstuw = ?fstuw*10.
fstuw = ?fstuw div 1.
fstuw = ?fstuw/10.

fssuw = ?fssuw*10.
fssuw = ?fssuw div 1.
fssuw = ?fssuw/10.

fsscu = ?fsscu*10.
fsscu = ?fsscu div 1.
fsscu = ?fsscu/10.

fssfa = ?fssfa div 1.
fssfa = ?fssfa/10.

pfstuw = ?fstuw*10 mod 10.
pfssuw = ?fssuw*10 mod 10.
pfsscu = ?fsscu*10 mod 10.

if ?pfstuw is 0 then fstuw is concat(?fstuw,'.')
else fstuw =?fstuw.

if ?pfssuw is 0 then fssuw is concat(?fssuw,'.')
else fssuw =?fssuw.

pfsscu = ?fsscu*10 mod 10.
if ?pfsscu is 0 then fsscu is concat(?fsscu,'.')
else fsscu =?fsscu.

if ?pfssfa is 0 then fssfa is concat(?fssfa,'.')
else fssfa =?fssfa.

write('soilprot.out', '#n Soil Properties for Embankment',
' #n -----------------------------------------------',
' #n Total Unit Weight (pcf): #s', ?estuw,
' #n Saturated Unit Weight (pcf): #s', ?essuw,
' #n Cohesion at Undrained Condition (psf): #s',
' #n Friction Angle at Undrained Condition (deg): #s',
' #n Soil Properties for Foundation',
' #n Total Unit Weight (pcf): #s', ?fstuw,
' #n Saturated Unit Weight (pcf): #s', ?fssuw,
' #n Cohesion at Undrained Condition (psf): #s',
' #n Friction Angle at Undrained Condition (deg): #s',
' #n -----------------------------------------------').

(*=======Consider loading conditions ========*)
use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis #d,
    #mPCSTABL6#m').
use_font( ).
text('#x3#y2.5 Which Loading Conditions are you considering in the calculation?').
bitmap(?tieback,5,4).
bitmap(?reinfor,45,4).
bitmap(?noneload,5,15).
Lvalue is radio_button([['Tieback',5,13],
                   ['Reinforcement',45,13], ['None of Them',5,24,t] ]
                   ,sload).

topic sload(Lvalue,smethod).
sload is ?Lvalue.
if ?sload is 'Tieback' then ssload is [ ] and Spencert( ).
if ?sload is 'Reinforcement' then ssload is 'Reinforcement' and Bishopr( ).
if ?sload is 'None of Them' then ssload is [ ] and anameth( ).
end.

topic anameth.
use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis#d').
use_font( )
text (' #x5#y3 o. Please select a Analysis Method of
Slope Stability').
mvalue is radio_button([['Simplified Bishop Method',8,5,t ],
                      ['Simplified Janbu Method',8,7], ['Spencer Method',
                       8,9] ], smethod).
bitmap(?tesa,40,4).
text('#x50#y4.5 Limit Equilibrium Failure Shapes
#x46 Moment Vertical Horizontal Available
#x41 SBM o o x Cir,Spc
#x41 SJM o o o Cir, Irr, Blk, Spc
#x41 SPM o o o Cir, Irr, Blk, Spc
#x41 -----------------------------------------------
#x41 o: satisfied, x: not satisfied
#x41 . Cir: Circular Failure Surface,
#x41 . Irr: Irregular Failure Surface,
#x41 . Blk: Block Failure Surface,
#x41 . Spc: A Specific Failure Surface').

topic smethod(mvalue).
smethod is ?mvalue.
if ?smethod is 'Simplified Bishop Method' then nmethod
is ?smethod and Bishop( ).
if ?smethod is 'Simplified Janbu Method' then nmethis
is ?smethod and Janbu( ).
if ?smethod is 'Spencer Method' then nmethis is ?smethod
and Janbu( ).

button2('Press Here Again =>>', sstable,55,26,30,1.5).
end.

topic Bishop.
text(' #x5#y16 o. Select a Failure Shape In #s',
     ?smethod).
scom is combo_box([Circular,Specific],10,17.5,27,4,
topic Janbu.
   text('  #x5#yl6 o. Select a Failure Shape In #s', ?method).
   scom is combo_box([Circular, Irregular, Specific], , 10,17.5,27,6,[simple,vertscroll],double_click_event).
   jbt is button2(Continue, Continue,5,26,15,1.5).
   wait ( ).
   close_window(?jbt).
   svalue is get_combo_box(?scom).
   if ?svalue is Circular then ssmeth is CIRCL2.
   if ?svalue is Specific then ssmeth is SURFAC.
end.

topic Bishopr.
   usefont(?boldFont).
   text('#e #fbrown Slope Stability Analysis #d').
   use_font( ).
   text('#x3#y3 Bishop Method is available for The Reinforcement
   #x5#y5 o. Select a Failure Shape In Bishop Method').
   scom is combo_box([Circular],,10,7,27,4,[simple, vertscroll],double_click_event).
   bsbt is button2(Continue, Continue,5,26,25,1.5).
   wait ( ).
   close_window(?bsbt).
   nmethod is 'Simplified Bishop Method'.
   svalue is get_combo_box(?scom).
   if ?svalue is Circular then ssmeth is CIRCL2.
   button2('Press Here Again =>>',sstable,55,26,30,1.5).
end.

topic Spencert.
   use_font(?boldFont).
   text('#e #fbrown Slope Stability Analysis #d').
   use_font( ).
   text('#x3#y3 Spencer Method is available for Tieback Load
   #x3#y5 o. Select a Failure Shape In Spencer Method').
scom is combo_box([Circular, Irregular],10,7,27,6,
simple,vertscroll,double_click_event).
spbt is button2(Continue, Continue,5,26,25,1.5).
wait( ).
close_window(?spbt).
mmethod is 'Spencer Method'.
svalue is get_combo_box(?scom).
if ?svalue is Circular then ssmeth is CIRCLE.
if ?svalue is Irregular then ssmeth is RANDOM.
button2('Press Here To Continue =>>',sstable,55,26,30,1.5).
end.
topic mark(item).
irow is 45 and icol is 3 and ewid is 43 and eht is 27.
helpwin is window(,?irow,?icol,?ewid,?eht,?item,
[thinframe,showchildren,visible],?main,,close_event).
button2('Close Me',close,1,1,11,1.3).
set_file_pos('stable.txt',0,beg).
stabl is read('stable.txt',concat('//',?item),'//').
text('#x2#y2.5',?stabl).
if ?item is PCSTABL6 then
bitmap(load_bitmap('stable.bmp'),2,14).
end.
topic close.
close_window( ).
show_window( ).
end.

(* = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
SOIL CLASSIFICATION By ODOT SYSTEM
=====================================================)
topic 'ODOT Method'.
soilwin is window(,42,6,48,13,'ODOT Soil Classification
System',[thinframe,showchildren,visible],?main,
lighgray,close_event).
button2('Restart',restart,l,1,11,1.3).
button2('Close Me',close,l2,l,11,1.3).
window (,1,2.3,48,10.5,,child,siblings,visible,
showchildren],?soilwin ).
soiltt is 'Group Classification of Your Material'.
text('#e  This system uses ODOT Soil Classification
based on AASHTO M145. In this method, soil is
classified into seven major groups, A-1 through
A-7. Soils classified under groups A-1,A-2, and
A-3 are granular materials. Groups A-4, A-5, A-6,
and A-7 are mostly silt and clay-type materials.
#fgreen2 Press the Soil-Class button below to start the
classification process.#d ').
button2('Soil Class',soil,2,9.8,15,1.4).
topic soil.

ask('Please choose a category of your material
Percent Passing in Sieve No. 10', pass10,
['Greater Than 50', '50 Max']).

do (?pass10).

topic '50 Max'.

text('Gravel and/or Stone Fragments
Group Index : 0').

doi.

topic 'Greater Than 50'.

ask('Please choose a category of your material
Percent Passing in Sieve No. 40', pass40,
['50 Max', '51 Max', 'Greater Than 51']).

do (?pass40).

topic '50 Max'.

text('Gravel and/or Stone Fragments with Sand
Group Index : 0').

doi.

topic '51 Max'.

text('Fine Sand
Group Index : 0').

doi.

topic 'Greater Than 51'.

ask('Please choose a category of your material
Percent Passing in Sieve No. 200', pass200,
['35 Max', '36 Min', '50 Min']).

do (?pass200).

topic '35 Max'.

ask('Please choose a category of your material.
Liquid Limit', LL, ['40 Max', '41 Min']).

do (LL).

topic '40 Max'.

ask('Please choose a category of your material.
Plasticity Index : PI', PI,
['6 Max', '10 Max', '11 Min']).

do (PI).

topic '6 Max'.

topic '10 Max'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-3a #x5#y4.5 Coarse and Fine Sand
#x15#y6 Group Index :  0').
end.
topic '11 Max'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-4 #x5#y4.5 Gravel and/or Stone
Fragments with Sand & Silt
#x15#y6 Group Index :  0').
end.
topic '12 Min'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-6 #x5#y4.5 Gravel and/or Stone
Fragments with Sand, Silt & Clay
#x15#y6 Group Index :  4').
end.
end.(*40 Max*)
topic '11 Min'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-5 #x5#y4.5 Gravel and/or Stone
Fragments with Sand and Silt
#x15#y6 Group Index :  0').
end.
topic '12 Min'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-7 #x5#y4.5 Gravel and/or Stone
Fragments with Sand, Silt, & Clay
#x15#y6 Group Index :  4').
end.
end.(*41 Min*)
topic '40 Max'.
ask('#e Please choose a category of your material.
(Plasticity Index : PI )', PI,
['10 Max', '11 Min']).
doi(?PI).
topic '10 Max'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-4 #x5#y4.5 Gravel and/or Stone
Fragments with Sand & Silt
#x15#y6 Group Index :  0').
end.
topic '11 Max'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-5 #x5#y4.5 Gravel and/or Stone
Fragments with Sand and Silt
#x15#y6 Group Index :  0').
end.
topic '12 Min'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-2-7 #x5#y4.5 Gravel and/or Stone
Fragments with Sand, Silt, & Clay
#x15#y6 Group Index :  4').
end.
end.(*41 Min*)
topic '36 Min'.
ask('#e Please choose a category of your material.
(Liquid Limit)', LL,
['40 Max', '41 Min']).
doi(?LL).
topic '40 Max'.
ask('#e Please choose a category of your material.
(Plasticity Index : PI )', PI,
['10 Max', '11 ~ 15','16 Min']).
doi(?PI).
topic '10 Max'.
text('#e #x5#y1.5 #s', ?soiltt,
'#x20#y3 A-4a #x5#y4.5 Sandy Silt
end.
topic '11 ~ 15'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-6a #x5#y4.5 Silt and Clay #x15#y6 Group Index : 10').
end.
topic '16 Min'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-6b #x5#y4.5 Silty Clay #x15#y6 Group Index : 16').
end.
end.(*40 Max*)
topic '41 Min'.
ask('#e Please choose a category of your material. (Plasticity Index : PI )', PI, ['10 Max', '<=LL - 30', '>LL - 30']).
do(?PI).
topic '10 Max'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-5 #x2#y4.5 Elastic Silt & Clay with or without Organic Material #x15#y6 Group Index : 12').
end.
topic '<=LL - 30'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-7 #x5#y4.5 Elastic Clay #x15#y6 Group Index : 20').
end.
topic '>LL - 30'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-7 #x5#y4.5 Clay #x15#y6 Group Index : 20').
end.
end.(*41 Min*)
end.(*36 Min*)

(*======== 50 Min ========*)
topic '50 Min'.
text('#e #x5#y1.5 #s', ?soiltt, '#x20#y3 A-4b #x5#y4.5 Silt #x15#y6 Group Index : 8').
end.
end.(*Greater Than 51*)
end.(*Greater Than 50*)
end.(*soil*)
topic restart.
  do(:soil).
end.

topic close.
  close_window( ).
  close_window( ).
  show_window( ).
end.

end.(*ODOT Method*)

topic 'Unified Method'.
  unifwin is window(,46,16,42,13,'Unified Soil Classification System',thinframe,showchildren,visible],?main,,lightgray,Close_event).
  button2('Close Me',close,1,1,11,1.3).
  window ,(1,2.3,42,10.5,,[child,siblings,visible,vertscroll,showchildren],?unifwin ).

  text('Soil Group in Soil Group in
       AASHTO          Unified

       -------------------------------
          A-1-a        GW, GP
          A-1-b        SW, SP, GM, SM
            A-3          SP
          A-2-4        GM, SM
          A-2-5        GM, SM
          A-2-6        GS, SC
          A-2-7        GM, GC, SM, SC
            A-4          ML, OL
          A-5          OH, MH, ML,OL
            A-6          CL
          A-7-5        OH, MH
          A-7-6        CH, CL').

  topic close.
  close_window( ).
  close_window( ).
  show_window( ).
end.

end.
topic sstable.
  new_file('slopel.in').
  new_file('error.out').
  new_file('slopel.out').
  new_file('slope2.out').
  new_file('slope3.out').
  new_file('slope4.out').
  new_file('sdesign.in').
  new_file('fort.out').
  zero_int is concat(0,0). use_font(?boldFont).
  text('#e #fbrown Slope Stability Analysis #d'). use_font().
  text('#x5#y2.5 ( R: Real Number, I : Integer )
  #x2#y4 @ Embankment Boundary
  #n #x3#y5 Total Number of Boundaries ( I )
  #n #x3#y6.5 Number of Surface Boundaries ( I )').

  ibnum is 4.
  isufnum is 3. if ?bdll <> [] then ibnum is ?numbdo and isufnum is
    ?numsufo.
  numbdi is set_focus(edit_line(?ibnum,,36,4.8,4)).
  numsufi is edit_line(?isufnum,,36,6.3,4).
  bitmap(load_bitmap('shapl.bmp'),2,7.5).
  smalll is button2(Continue,Continue,42,5,15,1.5).
  wait( ).
  close_window(?smalll).
  numbdo is get_text(?numbdi).
  numsufo is get_text(?numsufi).
  if ?numbdo >16 then bderror( ).
  if ?numbdo >16 then numbdo = 16.

topic bderror.
  bdwin is window(, 40,12,50,6, [dialogwindow,visible] ,
    ?main, ,,).
  Text(' @ Maximum Allowable Number of Total
    Boundaries are 16.').
  button2('Close Me',Continue,35,4.5,10,1.5).
  wait( ).
  close_window(?bdwin).
  show_window( ).
end.
comcardo = PROFIL.
titleo is ?project.
s=1.
while ?s <= 16
then
  concat(bint1,?s) is concat(0,,0) and
  concat(bint2,?s) is concat(0,,0) and
  concat(bint3,?s) is concat(0,,0) and
  concat(bint4,?s) is concat(0,,0) and
  concat(bint5,?s) is 1 and
  s =?s+1.

if ?bd11 <> [ ] then initial_input4( ).

topic initial_input4.
if ?numbdo >= ?exnum then minnum is ?exnum and maxnum is
?numbdo.
if ?numbdo < ?exnum then minnum is ?numbdo and maxnum is
?exnum.

sss=1.
while ?sss <= ?minnum
then
  concat(bint1,?sss) is concat(bint2,?sss) is
  concat(bint3,?sss) is concat(bint4,?sss) is
  concat(bint5,?sss) is
  sss =?sss+1.
end.(*)

exnum is ?numbdo.
m=8.
yaxis=?m - 2.
par=1.
if ?numbdo > 10 then m=3.5 and yaxis = ?m - 2.
text(concat('#x55#y',?yaxis),'Coordinates of Boundary
(unit: ft)
#x55 LX(R) LY(R) RX(R) RY(R) MT(I) ').

par=1.
while ?par <= ?numbdo
then
text('##n#x52',?par) and
  concat(valb1,?par) is set_focus(edit_line
  (?concat(bint1,?par),,55,?m,6)) and
  set_focus(?valb1) and
  concat(valb2,?par) is edit_line
  (?concat(bint2,?par),,61,?m,6) and
  concat(valb3,?par) is edit_line
  (?concat(bint3,?par),,67,?m,6) and
  concat(valb4,?par) is edit_line
  (?concat(bint4,?par),,73,?m,6) and
  concat(valb5,?par) is edit_line
  (?concat(bint5,?par),,79,?m,6) and
  m =?m+1.5 and
  par =?par +1.
ww is button2(Continue,continue,40,26,12,1.5).
wait( ).
close_window(?ww).
write('slopel.in',?comcardo,?titleo,?numbdo,'#s ',
?numsufo).

kopp = 1.
while ?kopp <= ?numbdo
then concat(bdl,?kopp) is get_text(?concat(valb1,?kopp))
   and concat(bd2,?kopp) is get_text(?concat(valb2,?kopp))
   and concat(bd3,?kopp) is get_text(?concat(valb3,?kopp))
   and concat(bd4,?kopp) is get_text(?concat(valb4,?kopp))
   and concat(bd5,?kopp) is get_text(?concat(valb5,?kopp))
   and kopp =?kopp +1.

write('sdesign.in',?numbdo).

if ?bunmo is 0 then bunmo is 1.
ratio is (?bd32 - ?bd12)/?bunmo.
ratio is format_number('##.#',?ratio).
seratio =?ratio.

knum = 1.
while ?knum <= ?numbdo
then write('slopel.in','#n',?concat(bdl,?knum),'#s ',
            ?concat(bd2,?knum),?concat(bd3,?knum),
            ?concat(bd4,?knum),  ?concat(bd5,?knum))  and
            write('sdesign.in','#n',?concat(bdl,?knum),'#s ',
            ?concat(bd2,?knum),?concat(bd3,?knum),
            ?concat(bd4,?knum))  and
            knum=?knum+1.

(* ========= Input for Soil Properties==================)

comsoil =SOIL.
emrow = 10.
fdrow = 18.

use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis #d').
use_font(   ) .
text('#x4#y3 @ Material Properties ( R: Real Number,  I :
    Integer )
    #x5#y4.5 Number of Materials in Embankment (I):
    #x5#y6 Number of Materials in Foundation (I):').
numemsoil is set_focus(edit_line(l,,45,4.3,4)).
numfdsoil is edit_line(l,,45,5.8,4).

small3 is button2(Continue,Continue,40,7.5,15,1.5).
wait(   ) .
close_window( small3).

numemsoil is get_text(?numemsoil).
numfdsoil is get_text(?numfdsoil).

if ?numemsoil >3 or ?numfdsoil >4 then soil_error1(   ) .
if ?numemsoil > 3 then numemsoil = 3.
if ?numfdsoil > 4 then numfdsoil = 4.

**topic soil_error1.**

soilwin is window(, 40, 12, 50, 8, [dialogwindow, visible],
?main, , ,).

Text(''
@ Maximum Allowable Number of Material Types
in Embankment are 3.

@ Maximum Allowable Number of Material Types
in Foundation are 4'').

button2('Close Me', Continue, 35, 6.5, 10, 1.5).

wait().
close_window(?soilwin).
show_window( ).
end.

efs = 1.
while ?efs <= 4 then
    concat(emint1, ?efs) is ?estuw and
    concat(emint2, ?efs) is ?essuw and
    concat(emint3, ?efs) is ?esscu and
    concat(emint4, ?efs) is ?essfa and
    concat(fdint1, ?efs) is ?fstuw and
    concat(fdint2, ?efs) is ?fssuw and
    concat(fdint3, ?efs) is ?fsscu and
    concat(fdint4, ?efs) is ?fssfa and
    efs = ?efs + 1.

if ?fdsoll <> [ ] then soil_initial( ).

**topic soil_initial.**

emx = 1.
while ?emx <= ?numemsoil then
    concat(emint1, ?emx) is ?concat(emsoil1, ?emx) and
    concat(emint2, ?emx) is ?concat(emsoil2, ?emx) and
    concat(emint3, ?emx) is ?concat(emsoil3, ?emx) and
    concat(emint4, ?emx) is ?concat(emsoil4, ?emx) and
    emx = ?emx + 1.

fdx = 1.
while ?fdx <= ?numfdsoil then
    concat(fdint1, ?fdx) is ?concat(fdsol1, ?fdx) and
    concat(fdint2, ?fdx) is ?concat(fdsol2, ?fdx) and
    concat(fdint3, ?fdx) is ?concat(fdsol3, ?fdx) and
    concat(fdint4, ?fdx) is ?concat(fdsol4, ?fdx) and
    fdx = ?fdx + 1.
end.(* *)

text('#x5#y8 Embankment Materials (Real Number)
#x10#y9#mMJW#m(pcf) #mSUW#m(pcf) #mCoh#m(psf)
#mPric#m(deg) #mPPP#m(I) #mPPC#m(I) #mPSN#m(I)').
text('#x5#y16 Foundation Materials')
numsoil is ?numemsoil + ?numfdsoil.

em=1.
while ?em <= ?numemsoil
then text(concat('"x3y',?emrow),'MT #s',?em) and
concat(emval1,?em) is edit_line
(?concat(emint1,?em),,10,?emrow,10) and
set_focus(?emval1l) and
concat(emval2,?em) is edit_line
(?concat(emint2,?em),,20,?emrow,10) and
concat(emval3,?em) is edit_line
(?concat(emint3,?em),,30,?emrow,12) and
concat(emval4,?em) is edit_line
(?concat(emint4,?em),,42,?emrow,8) and
concat(emval5,?em) is edit_line
(?zero_int,,50,?emrow,8) and
concat(emval6,?em) is edit_line
(?zero_int,,58,?emrow,8) and
concat(emval7,?em) is edit_line (1,,66,?emrow,8)
and emrow =?emrow+2
and em =?em +1.

fd=1.
while ?fd <= ?numfdsoil
then text(concat('"x3y',?fdrow),'MT #s',?numemsoil+?fd) and
concat(fdval1,?fd) is edit_line
(?concat(fdint1,?fd),,10,?fdrow,10) and
concat(fdval2,?fd) is edit_line
(?concat(fdint2,?fd),,20,?fdrow,10) and
concat(fdval3,?fd) is edit_line
(?concat(fdint3,?fd),,30,?fdrow,12) and
concat(fdval4,?fd) is edit_line
(?concat(fdint4,?fd),,42,?fdrow,8) and
concat(fdval5,?fd) is edit_line
(?zero_int,,50,?fdrow,8) and
concat(fdval6,?fd) is edit_line
(?zero_int,,58,?fdrow,8) and
concat(fdval7,?fd) is edit_line (1,,66,?fdrow,8)
and fdrow =?fdrow+2
and fd =?fd +1.

button2('Press Here To Continue =>>',Continue,55,26,30,1.5).
wait( )
write('slopel.in','#n',?comsoil,'#n',?numsoil).
emtot is 0.
emcoh is 0.
ep = 1.
while ?ep <= ?numemsoil
then concat(emsoil,?ep) is get_text(?concat(emval1,?ep))
   and concat(emsoil2,?ep) is get_text(?concat(emval2,?ep))
   and concat(emsoil3,?ep) is get_text(?concat(emval3,?ep))
   and concat(emsoil4,?ep) is get_text(?concat(emval4,?ep))
   and concat(emsoil5,?ep) is get_text(?concat(emval5,?ep))
   and concat(emsoil6,?ep) is get_text(?concat(emval6,?ep))
   and concat(emsoil7,?ep) is get_text(?concat(emval7,?ep))
   and emtot = ?concat(emsoil,?ep) + ?emtot
   and emcoh = ?concat(emsoil3,?ep) + ?emcoh
   and ep = ?ep + 1.
   pemtot = ?emtot/?numemsoil.
   pemcoh = ?emcoh/?numemsoil.
  epp = 1.
while ?epp <= ?numemsoil
then write('slopel.in', '#n', ?concat(emsoil,?epp),#s ',
   ?concat(emsoil2,?epp), ?concat(emsoil3,?epp),
   ?concat(emsoil4,?epp), ?concat(emsoil5,?epp),
   ?concat(emsoil6,?epp), ?concat(emsoil7,?epp))
and epp = ?epp+1.
fp=1.
while ?fp <= ?numfdsoil
then concat(fdsoil,?fp) is get_text(?concat(fdval1,?fp))
   and concat(fdsoil2,?fp) is get_text(?concat(fdval2,?fp))
   and concat(fdsoil3,?fp) is get_text(?concat(fdval3,?fp))
   and concat(fdsoil4,?fp) is get_text(?concat(fdval4,?fp))
   and concat(fdsoil5,?fp) is get_text(?concat(fdval5,?fp))
   and concat(fdsoil6,?fp) is get_text(?concat(fdval6,?fp))
   and concat(fdsoil7,?fp) is get_text(?concat(fdval7,?fp))
and fp = ?fp + 1.
fpp = 1.
while ?fpp <= ?numfdsoil
then write('slopel.in', '#n', ?concat(fdsoil,?fpp),#s ',
   ?concat(fdsoil2,?fpp), ?concat(fdsoil3,?fpp),
   ?concat(fdsoil4,?fpp), ?concat(fdsoil5,?fpp),
   ?concat(fdsoil6,?fpp), ?concat(fdsoil7,?fpp))
and fpp = ?fpp+1.
(*========= Limits and Water =========================*)
use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis #d #s',
   ?nmethod).
use_font( ).
text('#x3#y2.5 ( R: Real Number )
#x3#y3.5 #mWater Table#m (R) #mLimits#m (R) (Bed Rock)
#x3#y4.5 (Choose 7 points) (Choose 7 points)
#x4#y5.5 X Co Y Co X Co Y Co').
while \( ?wt \leq 7 \)
then  concat(\( iwtx,?wt \)) is \( ?\text{zero\_int} \) and
    concat(\( iwty,?wt \)) is \( ?\text{zero\_int} \) and
\( wt = ?wt +1. \)

if \( ?wtxxl <> [ ] \) then \( wt\_\text{input}() \).

\addtocounter{equation}{1}

**wt\_input.**

\( wt=1. \)
while \( ?wtt \leq 7 \)
then  concat(\( iwtx,?wtt \)) is \( ?\text{concat}(wtxx,?wtt) \) and
    concat(\( iwty,?wtt \)) is \( ?\text{concat}(wtyy,?wtt) \) and
\( wtt = ?wtt +1. \)
end.(\* \*)

\( wtx1 \) is set\_focus(\( \text{edit\_line}(?iwtx1,,4,7,7)) \).
\( wty1 \) is edit\_line(\( ?iwty1,,12,7,7) \).
\( wtx2 \) is edit\_line(\( ?iwtx2,,4,8.5,7) \).
\( wty2 \) is edit\_line(\( ?iwty2,,12,8.5,7) \).
\( wtx3 \) is edit\_line(\( ?iwtx3,,4,10,7) \).
\( wty3 \) is edit\_line(\( ?iwty3,,12,10.7) \).
\( wtx4 \) is edit\_line(\( ?iwtx4,,4,11.5,7) \).
\( wty4 \) is edit\_line(\( ?iwty4,,12,11.5,7) \).
\( wtx5 \) is edit\_line(\( ?iwtx5,,4,13,7) \).
\( wty5 \) is edit\_line(\( ?iwty5,,12,13,7) \).
\( wtx6 \) is edit\_line(\( ?iwtx6,,4,14.5,7) \).
\( wty6 \) is edit\_line(\( ?iwty6,,12,14.5,7) \).
\( wtx7 \) is edit\_line(\( ?iwtx7,,4,16,7) \).
\( wty7 \) is edit\_line(\( ?iwty7,,12,16,7) \).

\( Lmt=1. \)
while \( ?Lmt \leq 7 \)
then  concat(\( iLmx,?Lmt \)) is \( ?\text{zero\_int} \) and
    concat(\( iLmy,?Lmt \)) is \( ?\text{zero\_int} \) and
\( Lmt = ?Lmt +1. \)

if \( ?Lmxxl <> [ ] \) then \( Lm\_\text{input}() \).

\addtocounter{equation}{1}

**Lm\_input.**

\( Lmm=1. \)
while \( ?Lmm \leq 7 \)
then  concat(\( iLmx,?Lmm \)) is \( ?\text{concat}(Lmxx,?Lmm) \) and
    concat(\( iLmy,?Lmm \)) is \( ?\text{concat}(Lmyy,?Lmm) \) and
\( Lmm = ?Lmm +1. \)
end.(\* \*)

\( Lmx1 \) is edit\_line(\( ?iLmx1,,27,7,7) \).
\( Lmy1 \) is edit\_line(\( ?iLmy1,,35,7,7) \).
\( Lmx2 \) is edit\_line(\( ?iLmx2,,27,8.5,7) \).
\( Lmy2 \) is edit\_line(\( ?iLmy2,,35,8.5,7) \).
\( Lmx3 \) is edit\_line(\( ?iLmx3,,27,10.7) \).
\( Lmy3 \) is edit\_line(\( ?iLmy3,,35,10.7) \).
\( Lmx4 \) is edit\_line(\( ?iLmx4,,27,11.5,7) \).
\( Lmy4 \) is edit\_line(\( ?iLmy4,,35,11.5,7) \).
183

Lmx5 is edit_line (?ILmx5,,27,13,7).
Lmy5 is edit_line (?ILmy5,,35,13,7).
Lmx6 is edit_line (?ILmx6,,27,14.5,7).
Lmy6 is edit_line (?ILmy6,,35,14.5,7).
Lmx7 is edit_line (?ILmx7,,27,16,7).
Lmy7 is edit_line (?ILmy7,,35,16,7).

if ?ssmeth is CIRCLE then scircle( ).
if ?ssmeth is CIRCL2 then scircl2( ).
if ?ssmeth is RANDOM then srandom( ).
if ?ssmeth is SURFAC or ?ssmeth is SURBIS then sspecific( ).

topic scircle.
  text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,'#m',
     '#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4
     #x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').

  spxll is edit_line (?zero_int,,50,6,7).
  spxrl is edit_line (40.1,,58,6,7).
  spxl2 is edit_line (50.1,,66,6,7).
  spxr2 is edit_line (179.9,,74,6,7).
  me is edit_line (?zero_int,,50,9,7).
  ls is edit_line (5.5,,58,9,7).
  ccsi is edit_line (?zero_int,,66,9,7).
  csi is edit_line (?zero_int,,74,9,7).
end.

topic scircl2.
  text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,'#m',
     '#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4
     #x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').

  spxll is edit_line (?zero_int,,50,6,7).
  spxrl is edit_line (40.1,,58,6,7).
  spxl2 is edit_line (130.1,,66,6,7).
  spxr2 is edit_line (179.9,,74,6,7).
  me is edit_line (?zero_int,,50,9,7).
  ls is edit_line (4.5,,58,9,7).
  ccsi is edit_line (?zero_int,,66,9,7).
  csi is edit_line (?zero_int,,74,9,7).
end.

topic srandom.
  text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,'#m',
     '#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4
     #x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').

  spxll is edit_line (10.1,,50,6,7).
  spxrl is edit_line (40.1,,58,6,7).
  spxl2 is edit_line (130.1,,66,6,7).
  spxr2 is edit_line (179.9,,74,6,7).
  me is edit_line (?zero_int,,50,9,7).

.
ls is edit_line (15.1,,58,9,7).
cs1 is edit_line (-15.1,,66,9,7).
csi is edit_line (-44.5,,74,9,7).
end.

topic sspecific.
text('"xy2.5 Failure Surface (R)"s#m',?svalue,'"xy3.5 (select 8 points for a specific failure
surface)"
#x50#y5 X Co Y Co X Co Y Co').
sx1 is edit_line (1.2,,50,6,7).
sy1 is edit_line (5.1,,58,6,7).
sx2 is edit_line (40.1,,68,6,7).
sy2 is edit_line (8.5 ,,76,6,7).
sx3 is edit_line (55.5 ,,50,7.5,7).
sy3 is edit_line (8.1 ,,58,7.5,7).
sx4 is edit_line (110.1 ,,68,7.5,7).
sy4 is edit_line (7.2 ,,76,7.5,7).
sx5 is edit_line (125.1 ,,50,9,7).
sy5 is edit_line (17.1 ,,58,9,7).
sx6 is edit_line (140.5 ,,68,9,7).
sy6 is edit_line (20.1 ,,76,9,7).
sx7 is edit_line (179.1 ,,50,10.5,7).
sy7 is edit_line (7.5 ,,58,10.5,7).
sx8 is edit_line (179.1 ,,68,10.5,7).
sy8 is edit_line (7.5 ,,76,10.5,7).
end.

if ?ssload is 'Reinforcement' then reinf( ).
if ?ssload is Tieback then tieback( ).

topic reinf.
small5 is button2(Continue,Continue,55,12,15,1.5).
wait( ).
close_window(?small5).

text('"x50#y13 mReinforcement#m (R)
x50#y14.5 Number of Reinforcement (I):
x50#y16 Number of Points (I):').
numreinf is set_focus(edit_line(1,,80,14.3,4)).
numpont is edit_line(3,,80,15.8,4).

rein is button2(Continue,Continue,75,17.5,10,1.5).
wait( ).
close_window(?rein).

nreinf is get_text(?numreinf).
npoint is get_text(?numpont).
if ?npoint >=5 then npoint = 4 and npnum( ).

topic npnum.
reinfwin is window(/,45,15,41,4, [dialogwindow,
visible], ?main, , ).
Text('Maximum Allowable Number of Points are 4').
button2('Close Me',Continue,5,3.5,15,1).
wait( ).
close_window(?reinfwin).
show_window( ).
end.
text('#x50#yl7.5 X Co(R) Y Co(R) Force(Lb/ft) I-Fac (R)').
iwx = concat(40,.,0).
iwy = concat(35,.,0).
rp = 1.
rr = 1.
m = 18.5.
iforcel = concat(0,.,0).
iforcel2 = concat(503,.,0).
iforcel3 = concat(503,.,0).
while ?rr <= ?npoint
then concat(rvalbl,?rr) is edit_line (?iwx,,50,?m,6) and
    set_focus(?rvalbl) and
    concat(rvalb2,?rr) is edit_line (?iwy,,58,?m,6) and
    concat(rvalb3,?rr) is edit_line (?concat(iforcel,?rr),,66,?m,10) and
    concat(rvalb4,?rr) is edit_line (?zero_int,,78,?m,6) and
    m =?m+2 and rr =?rr +1.
end. (*reinf*)
topic tieback.
small6 is button2(Continue,Continue,55,12,15,1.5).
wait( ).
close_window(?small6).
text('#x47#yl3 #mTieback Load#m (R) #x47#yl4.5 Number of Tieback Loads (I):').
numtie is set_focus(edit_line(l,,77,14.3,4)).
tie is button2(Continue,Continue,68,16,10,1.5).
wait( ).
close_window(?tie).
ntie is get_text(?numtie).
if ?ntie >=6 then ntie = 6 and ntienum( ).
topic ntienum.
tiewin is window(, 45,15,41,4, [dialogwindow,visible] , ?main, , ).
Text('Maximum Allowable Number of Tieback Loads are 5').
button2('Close Me',Continue,5,3.5,15,1).
wait( ).
close_window(?tiewin).
show_window( ).
end. (*ntienum*)
text(#x47#y16 N(I) Y (R) Load(R) Sp(R) In(R) Lth(R')).

tb=1.
m=17.
while ?tb <= ?ntie then concat(tvalblb1,tb) is edit_line (1,,47,?m,4) and
  set_focus(?tvalblb1) and
  concat(tvalblb2,tb) is edit_line (,,52,?m,6) and
  concat(tvalblb3,tb) is edit_line (,,59,?m,8) and
  concat(tvalblb4,tb) is edit_line (,,66,?m,6) and
  concat(tvalblb5,tb) is edit_line (,,73,?m,6) and
  concat(tvalblb6,tb) is edit_line (,,80,?m,6) and
  m =?m+1.5 and
end.*(tieback*)
button2('Press Here To Continue =>>',Continue,55,26,30,1.5).
wait( ) .

wtg=1.
while ?wtg <= 7 then concat(wtxx,wtg) is get_text (?concat(wtx,?wtg))
  and concat(wtyy,wtg) is get_text (?concat(wty,?wtg))
  and wtg = ?wtg +1.
write('slopel.in', '#n', WATER, '#n', 1,'#s',62.4,'#n', 7,'#n',
  ?wtxxl,'#s',?wtyyl,'#n',
  ?wtxx2,'#s',?wtyy2,'#n',
  ?wtxx3,'#s',?wtyy3,'#n',
  ?wtxx4,'#s',?wtyy4,'#n',
  ?wtxx5,'#s',?wtyy5,'#n',
  ?wtxx6,'#s',?wtyy6,'#n',
  ?wtxx7,'#s',?wtyy7).
write('sdesign.in', '#n', 7,'#n',
  ?wtxxl,'#s ',?wtyyl,'#n',
  ?wtxx2,'#s',?wtyy2,'#n',
  ?wtxx3,'#s',?wtyy3,'#n',
  ?wtxx4,'#s',?wtyy4,'#n',
  ?wtxx5,'#s',?wtyy5,'#n',
  ?wtxx6,'#s',?wtyy6,'#n',
  ?wtxx7,'#s',?wtyy7).

Lm=1.
while ?Lm <= 7 then concat(Lmxx,?Lm) is get_text (?concat(Lmx,?Lm))
  and concat(Lmyy,?Lm) is get_text (?concat(Lmy,?Lm)) and
  Lm = ?Lm +1.
write('slopel.in', '#n', LIMITS, '#n',6,'#s ',6,'#n',
  ?Lmxx1,'#s',?Lmyy1,?Lmxx2,?Lmyy2,'#n',
  ?Lmxx2,'#s',?Lmyy2,?Lmxx3,?Lmyy3,'#n',
  ?Lmxx3,'#s',?Lmyy3,?Lmxx4,?Lmyy4,'#n',
  ?Lmxx4,'#s',?Lmyy4,?Lmxx5,?Lmyy5,'#n',
  ?Lmxx6,'#s',?Lmyy5,?Lmxx7,?Lmyy7).
write('sdesign.in', '#n', 7, '#n',
'Lmxx1', '#s', 'Lmyy1', '#n',
'Lmxx2', '#s', 'Lmyy2', '#n',
'Lmxx3', '#s', 'Lmyy3', '#n',
'Lmxx4', '#s', 'Lmyy4', '#n',
'Lmxx5', '#s', 'Lmyy5', '#n',
'Lmxx6', '#s', 'Lmyy6', '#n',
'Lmxx7', '#s', 'Lmyy7').

if ?ssload is Reinforcement then reinftw( ).
if ?ssload is Tieback then tiewt( ).

topic reinftw.
write('slopet.in', '#n', REINF, '#n', ?nreinf, '#n', ?npoint).
while ?rp <= ?npoint
    then write('slopet.in', '#n',
        [get_text(?concat(rvalbl, ?rp)),'#s ',
        get_text(?concat(rvalb2, ?rp)),
        get_text(?concat(rvalb3, ?rp)),
        get_text(?concat(rvalb4, ?rp)) ] ) and
        rp += ?rp +1.
end.

topic tiewt.
trp = 1.
write('slopet.in', '#n', TIES, '#n', ?ntie).
while ?trp <= ?ntie
    then write('slopet.in', '#n',
        [get_text(?concat(tvalbl, ?trp)),'#s ',
        concat(0, ., 0),
        get_text(?concat(tvalb2, ?trp)),
        get_text(?concat(tvalb3, ?trp)),
        get_text(?concat(tvalb4, ?trp)),
        get_text(?concat(tvalb5, ?trp)),
        get_text(?concat(tvalb6, ?trp)) ] ) and
        trp += ?trp +1.
end.

if ?ssmeth is CIRCLE or ?ssmeth is CIRCL2 or ?ssmeth is
    RANDOM then wcircle( ).
if ?ssmeth is SURFAC or ?ssmeth is SURBIS then wspecific( ).

topic wcircle.
write('slopet.in', '#n', ?ssmeth, '#n', 10, '#s ', 10, '#n',
    [get_text (?spxl1), '#s', get_text (?spxl1),
    get_text (?spxl2), get_text (?spxl2), '#n',
    get_text (?me), get_text (?ls), '#s', get_text (?ccsi),
    get_text (?ccsi)], '#n').
end.

topic wspecific.
write('slopet.in', '#n', ?ssmeth, '#n8',

get_text (?sx1), '#s ', get_text (?syl), '#n',
get_text (?sx2), '#s ', get_text (?sy2), '#n',
get_text (?sx3), '#s ', get_text (?sy3), '#n',
get_text (?sx4), '#s ', get_text (?sy4), '#n',
get_text (?sx5), '#s ', get_text (?sy5), '#n',
get_text (?sx6), '#s ', get_text (?sy6), '#n',
get_text (?sx7), '#s ', get_text (?sy7), '#n',
get_text (?sx8), '#s ', get_text (?sy8), '#n', EXECUT).

write('sdesign.in', '#n8',
get_text (?sx1), '#s ', get_text (?syl), '#n',
get_text (?sx2), '#s ', get_text (?sy2), '#n',
get_text (?sx3), '#s ', get_text (?sy3), '#n',
get_text (?sx4), '#s ', get_text (?sy4), '#n',
get_text (?sx5), '#s ', get_text (?sy5), '#n',
get_text (?sx6), '#s ', get_text (?sy6), '#n',
get_text (?sx7), '#s ', get_text (?sy7), '#n',
get_text (?sx8), '#s ', get_text (?sy8)).

end. (*wspecific*)

key_sstable is 999.
write('error.out', 1).
set_file_pos('slopel.in', 0, beginning).
run('stable.exe').

if ?ssload is Reinforcement then wreinf( ) else woutren( ).
topic wreinf.
rrpp=1.
write('slope4.out', ?npoint).
while ?rrpp <= ?npoint
  then write('slope4.out', '#n',
    [get_text(?concat(rvalbl, ?rrpp)),'#s ',
    get_text(?concat(rvalb2, ?rrpp))]) and
    rrpp = ?rrpp + 1.
end.
topic woutren.
write('slope4.out', 0).
end.

set_file_pos('slopel.out', 0, beginning).
set_file_pos('slope2.out', 0, beginning).
set_file_pos('slope3.out', 0, beginning).
set_file_pos('slope4.out', 0, beginning).
use_font( ?boldfont).
text('#e #fbrown Failure Surfaces for Soil Embankment #d').
use_font( ).
run('vbslope.exe').
set_file_pos('slopel.out', -10, end).
OFOS is read_line('slopel.out').
FFOS is ?OFOS.
set_file_pos('error.out', 0, beginning).
error_read is read_line('error.out').

bitmap(load_bitmap('box19.bmp'),15,7).
text('#x18#y9 Please select the following option.').
check_box('Recalculate Stability Using Soils',recalsstabl,20,11).
check_box('Continue to Settlement Analysis',ssettle,20,13).
if ?error_read <> 1 then text('#x20#y15 Error in the Program is #s#m',?error_read,'#m') else text( ).

topic recalsstabl.
do(!main:soil_all:sstable).
end.

topic mark(item).
irow is 10 and icol is 17 and ewid is 60 and eht is 10.
subtitle is concat('Error Code ',?item).
if ?item is 'Embankment Boundary' then subtitle is ?item and irow is 42 and icol is 4 and ewid is 45 and eht is 22.
if ?item is MUW or ?item is SUW or ?item is Coh or ?item is Fric or ?item is PPP or ?item is PPC or ?item is PSN then subtitle is 'Material Properties' and irow is 2 and icol is 5 and ewid is 47 and eht is 23.
if ?item is Circular or ?item is Irregular then irow is 2 and icol is 5 and ewid is 42 and eht is 20 and subtitle is 'Range of Failure Surface Generation'.
if ?item is 'Reinforcement' then subtitle is ?item and irow is 10 and icol is 5 and ewid is 37 and eht is 20.
if ?item is 'Water Table' or ?item is Limits or ?item is 'Tieback Load' or ?item is Specific then subtitle is ?item and irow is 2.5 and icol is 19 and ewid is 45 and eht is 11.5.
if ?item is 'Correction Factor' then subtitle is concat(?item,' for Su') and irow is 2 and icol is 12 and ewid is 55 and eht is 15.5.

helpwin is window(?,irow,icol,ewid,eht,subtitle, [thinline,showchildren,visible],?main,, lightgray,close_event).
button2('Close Me',close,1,1,1,1.3).
window(,1,2.3,?ewid,?eht-2.5, [child,siblings, visible,vertscroll,showchildren],?helpwin ).
set_file_pos('stable.txt',0,beginning).
stabl is read('stable.txt',concat('//',?item),'//').
text('#e',?stabl).
if ?item is 'Embankment Boundary' then bitmap(load_bitmap('shape2.bmp'),1,1).
if ?item is Reinforcement then bitmap(load_bitmap('reinf.bmp'),1,1).
if ?item is 'Water Table' then bitmap(load_bitmap('waterhlp.bmp'),2,2).
if ?item is Limits
then bitmap(load_bitmap('limithlp.bmp'),2,2).
if ?item is 'Tieback Load'
then bitmap(load_bitmap('tiehlp.bmp'),2,2).
if ?item is Specific
then bitmap(load_bitmap('spechlp.bmp'),2,2).
if ?item is Circular or ?item is Irregular
then bitmap(load_bitmap('circle.bmp'),2,2)
and recir is button2('Press Here For
More',cir_more,7,16.5,25,1.5).
if ?item is 'Correction Factor' then
bitmap(load_bitmap('sufactor.bmp'),1,1).
end.
topic cir_more.
  close_window(?recir).
  window(2,6.2,38,22.5, 'Generation of Trial Failure
  Surface', [thinframe,showchildren,visible],?main, ,
  lightgray,close_event).
  bitmap(load_bitmap('circle2.bmp'),2,2).
  button2('Close Me',closes,1,1,11,1.3).
topic closes.
  close_window( ) .
end.
topic close.
  close_window( ).
  close_window( ).
  show_window( ).
end.
end.(*numeric*)
topic ssettle.
new_file('settle.in').
new_file('settle1.out').
new_file('settle2.out').
new_file('pcset.in').
use_font(?boldfont).
text('#e#x3#y1.5 #fbrown Settlement Analysis #d').
use_font().
text('#x3#y3 What type of drainage conditions do you have in the field ?').
bitmap(load_bitmap('boxl3.bmp'),4,5).
bitmap(load_bitmap('drain.bmp'),6,7).

```
drainList is [  
  'Type I - A',  
  'Type I - B',  
  'Type I - C',  
  'Type I - D',  
  'Type II - A',  
  'Type III - A' ].
```

draintp is combo_box (?drainList, ,45,6,30,12,'Type I - A', [simple,vertscroll], double_click_event).
button2('Press Here To Continue =>>',continue,55,26,30,1.5).
wait().

```
drain is get_combo_box(?draintp).

if ?drain is 'Type I - A' or ?drain is 'Type I - B' or  
  ?drain is 'Type I - C' then type = 1 and path = 1.
if ?drain is 'Type I - D' then type = 1 and path = 2.
if ?drain is 'Type II - A' then type = 2 and path = 2.
if ?drain is 'Type III - A' then type = 3 and path = 2.

  ise is 1.
  isf is 1.
  if ?key_ssettle > 0 then ise is ?numse and isf is ?numsf.
```

use_font(?boldfont).
text('#e#x3#y1.5 #fbrown Settlement Analysis #d').
use_font().
text('#x3#y4 @ Define number of layers
  #x3#y6 Number of Layers in Embankment
  #x3#y7.5 Number of Layers in Foundation
  #x47#y12 Number of Layers in Fill : #fgreen 2 #d
  #x47 Number of Layers in Foundation : #fgreen 3 #d
  #x47#y14.5 o. Note:'
Split many layers of foundation to get accurate settlement calculation even if the foundation includes one uniform soil property.

numsei is set_focus(edit_line(?ise,,36,5.8,4)).
numsf is edit_line(?isf,,36,7.3,4).
bitmap(load_bitmap('box13.bmp'),45,4).
bitmap(load_bitmap('laynum.bmp'),47,5).
button2('Press Here To Continue =>',continue,55,26,30,1.4).
wait( ).

numse is get_text(?numsei).
numsf is get_text(?numsf).
if ?numse >=4 or ?numsf >=4 then sethlp( ).
if ?numse >=4 then numse = 3.
if ?numsf >=4 then numsf = 3.

topic sethlp.
setwin is window(, 40,12,50,8,, [dialogwindow, visible],?main, , ).
text('@ Maximum Allowable Number of Layers in Embankment are 3.

@ Maximum Allowable Number of Layers in Foundation are 3').
button2('Close Me',Continue,35,6.5,10,1.5).
wait( ).
close_window(?setwin).
show_window( ).
end.
use_font(?boldfont).
text('#x3y1.5 #fbrown Settlement Analysis #d').
use_font( ).
text('#x3y4 @ Determine OCR value at the middle of each depth.

Average Stress History Profile
OCR = svm / Svo

oint =1.
while ?oint <=3
then concat(ocrint,?oint) is concat(1,..,0) and oint=?oint+1.

if ?key_ssettle > 0 then ocr_init( ).

topic ocr_init.
if ?numse >= ?senum then eminnum is ?senum and emaxnum is ?numse.
if ?numse < ?senum then eminnum is ?numse and emaxnum is ?senum.
if ?numsf >= ?sfnum then fminnum is ?sfnum and fmaxnum is ?numsf.
if ?numsf < ?sfnum then fminnum is ?numsf and fmaxnum is ?sfnum.

ocrit=1.
while ?ocrit <= ?fminnum
then concat(ocrint,?ocrit) is ?concat(ocr,?ocrit) and
ocrit =?ocrit +1.
end.

ocrmm = 10.
ocrm=1.
while ?ocrm <= ?numsf
then concat(ocred,?ocrm) is edit_line
   (?concat(ocrint,?ocrm),?,ocrmm,7.4,7) and
   set_focus(?ocred1) and

bitmap(load_bitmap('box1.bmp'),45,4).
bitmap(load_bitmap('ocr.bmp'),47,8).
button2('Press Here To Continue =>>',continue,55,26,
       30,1.4).
wait( ).

ocrmg=1.
while ?ocrmg <= ?numsf
then concat(ocr,?ocrmg) is
get_text(?concat(ocred,?ocrmg)) and
ocrmg = ?ocrmg +1.

text('#e #fbrown Settlement Calculation #d, #mSETFGD#m').
use_font ( ).
text ('#e #y2.5 @ Shape of Embankment
   x3#y4 Slope Ratio(SR) #x3#y5.5 Height of Embankment
   (ft) (HE) #x3#y7 Depth of Foundation (ft) (DF) #x3#y8.5
   Water Table (ft)(WT)#x3#y10 Top half width (ft)(TW) ').
text ('x2#y2.5 @ Material Properties of Embankment.
   x2#y14.5 mDepth of Each Layer(m (EDEL)
   x2#y16 mTotal Unit Weight(pcf)#m').
text ('x45#y2.5 @ Material Properties of Foundation
   Soil').
text('x45#y4 mDepth of Each Layer(m(FDEL)
   x45#y5.5 mTotal Unit Weight(pcf)#m
   x45#y7 mSaturated Unit Weight(pcf)#m
   x45#y8.5 mCompression Index#m
   x45#y10 mRecompression Index#m
   x45#y11.5 mVoid Ratio#m').
text('x45#y13.5 mConsolidation Coef.(ft2/day)#m
   x50#y15 m2nd Compression Index#m
   x45#y18 @ mConstruction Period (days)#m
   x45#y19.5 @ mConsolidation Degree (%)#m').
text('The default values were obtained from Data Base or from Previous Input').

bitmap(load_bitmap('setshp.bmp'),5,18).

rat is ?ratio.
hi is ?height.
dep is ?depth.
wat is concat(10,..,0).
wid is ?width.
dri is concat(20,..,0).

if ?key_ssettle > 0 then initial_shape( ).

end.

rat is set_focus(edit_line(?rat,34,4,7)).
hi is edit_line(?hi,34,5.5,7).
dep is edit_line(?dep,34,7,7).
wat is edit_line(?wat,34,8.5,7).
topw is edit_line(?topw,34,10,7).

sem =1.
while ?sem <= 3
then concat(seint,?sem) is concat(10,..,0) and
concat(seint2,?sem) is ?emsol11 and
sem=sem+1.

s=1.
while ?s <= 3
then concat(sfint,?s) is concat(10,..,0) and
concat(sfint2,?s) is ?fdsol11 and
concat(sfint3,?s) is ?fdsol21 and
concat(sfint4,?s) is ?fsc and
concat(sfint5,?s) is ?scr and
concat(sfint6,?s) is ?fsvoid and
concat(sfint7,?s) is ?fscv and
s =?s+1.

if ?key_ssettle > 0 then initial_sf2( ).

end.
concat(sfint5,?sf) is ?concat(sfgval5,?sf) and
concat(sfint6,?sf) is ?concat(sfgval6,?sf) and
concat(sfint7,?sf) is ?concat(sfgval7,?sf) and
sf =?sf+1.
end.(*  *)
senum is ?numse.
sfnum is ?numsf.
semm = 28.
set=1.
while ?set <= ?numse then
  concat(sevall,Pset) is edit_line
    (?concat(seint1,Pset),,semm,14.5,7) and
  concat(seval2,Pset) is edit_line
    (?concat(seint2,Pset),,semm,16,7) and
sfm = 69.
sft=1.
while ?sft <= ?numsf then
  concat(sfval1,Psft) is edit_line
    (?concat(sfint1,Psft),,sfm,4,7) and
  concat(sfval2,Psft) is edit_line
    (?concat(sfint2,Psft),,sfm,5.5,7) and
  concat(sfval3,Psft) is edit_line
    (?concat(sfint3,Psft),,sfm,7,7) and
  concat(sfval4,Psft) is edit_line
    (?concat(sfint4,Psft),,sfm,8.5,7) and
  concat(sfval5,Psft) is edit_line
    (?concat(sfint5,Psft),,sfm,10,7) and
  concat(sfval6,Psft) is edit_line
    (?concat(sfint6,Psft),,sfm,11.5,7) and
  concat(sfval7,Psft) is edit_line
    (?concat(sfint7,Psft),,sfm,13,7) and
  sfm =?sfm+7 and sft =Psft +1.
sendc is edit_line(?fsca,,75,15,7).
timel is edit_line(200,,71,18,7).
condeg is edit_line(90,,71,19.5,7).
button2('Press Here To Continue =>>',continue,55,26,30,1.5).
wait(  ).
  ratio is get_text(?rat).
  height is get_text(?hig).
  depth is get_text(?dep).
  water is get_text(?wat).
  width is get_text(?topwid).
emhigh=0.
setg=1.
while ?setg <= ?numse then
  concat(segvall,Psetg) is get_text
(?concat(seval1,?setg)) and
concat(segval2,?setg) is get_text
(?concat(seval2,?setg)) and
emhigh = ?concat(segval1,?setg) + ?emhigh and
setg= ?setg+1.

fdhigh = 0.
sftg=1.
while ?sftg <= ?numsf
then concat(sfgval1,?sftg) is get_text
(?concat(sfval1,?sftg)) and
concat(sfgval2,?sftg) is get_text
(?concat(sfval2,?sftg)) and
concat(sfgval3,?sftg) is get_text
(?concat(sfval3,?sftg)) and
concat(sfgval4,?sftg) is get_text
(?concat(sfval4,?sftg)) and
concat(sfgval5,?sftg) is get_text
(?concat(sfval5,?sftg)) and
concat(sfgval6,?sftg) is get_text
(?concat(sfval6,?sftg)) and
concat(sfgval7,?sftg) is get_text
(?concat(sfval7,?sftg)) and
fdhigh = ?concat(sfgval1,?sftg) + ?fdhigh and
sftg = ?sftg +1.

sendcom is get_text(?sendc).
cond is get_text(?condeg).
time is get_text(?timei).
if ?height <> ?emhigh then emhigh_hlp( ).
if ?depth <> ?fdhigh then fdhigh_hlp( ).
if ?water < 0 or ?water >= ?fdhigh then water_hlp( ).

topic emhigh_hlp.
emhwin is window(, 40,12,50,9,, [dialogwindow,visible]
,?main, ,,).
text('@ Total Height of Embankment should be equal
to sum of each layer in Embankment.
HE = Sum of EDEL' ).

semh = 5.
seh=1.
while ?seh <= ?numse
then concat(seval1,?seh) is edit_line
(concat(0,,0),,?semh,6.5,7) and
set_focus(?sevall1) and
seh= ?seh+1 and
semh= ?semh+7.

button2('Close Me',Continue,35,6.5,10,1.5).

wait( ).
seth=1.
while ?seth <= ?numse
then concat(segval1,?seth) is get_text
(?concat(seval1,?seth)) and
seth=seth+1.
close_window(?emhwin).
show_window(?).
end.

**topic fdhigh_hlp.**

fdhwin is window(,40,12,50,9,,[dialogwindow,visible]
,?main,,).
text('@ Total Depth of Foundation should be equal
to sum of each layer in Foundation.
\[DF = \text{Sum of FDEL}\].
sfmh=5.
sfth=1.
while ?sfth <= ?numsf
then concat(sfvall,?sfth) is edit_line(concat(0,,0),

   ?sfmh,6,5,7) and
   set_focus(?sfval11) and
   sfth = ?sfth +1 and
   sfmh = ?sfmh +7.

button2('Close Me',Continue,35,6,5,10,1.5).
wait( ).
sftgh=1.
while ?sftgh <= ?numsf
then concat(sfgvall,?sftgh) is get_text
   (?concat(sfvall,?sftgh)) and
   sftgh = ?sftgh +1.
close_window(?fdhwin).
show_window( ).
end.

**topic water_hlp.**

wtwin is window(,40,12,50,9,,[dialogwindow,visible]
,?main,,).
text('@ Depth of Water Table is assumed within
the depth of foundation.
\[0 \leq \text{water table} \leq \text{depth of foundation}\].
wat is set_focus(edit_line(0,7,6,5,7)).

button2('Close Me',Continue,35,6,5,10,1.5).
wait( ).
water is get_text(?wat).
close_window(?wtwin).
show_window( ).
end.

(*============ write in files============*)

segment=1.
write('settle.in',?ratio,'@s ',?height,?depth,

   ?water,?width,?sendcom,'@n',?numse).
while ?segment <= ?numse
then write('settle.in','@n',?concat(segval1,?segment),'@s '

   ,?concat(segval2,?segment)) and
segm = segm + 1.

write('settle.in', '#n', 'type', '#s ', 'path', 'cond', 'time,
' '#n', 'numsf').
ungw = 1.
while unsgw <= unnumsf
then write('settle.in', '#n', 'concat(ocr, unsgw), '#s
' , 'concat(sfgvall, unsgw), 'concat(sfgval2, unsgw),
' 'concat(sfgval3, unsgw), 'concat(sfgval4, unsgw),
' 'concat(sfgval5, unsgw), 'concat(sfgval6, unsgw),
' 'concat(sfgval7, unsgw)) and
unsgw = unsgw + 1.

(*===== Write for Report ========*)
write('pcset.in', '#n Input Data for SETFGD',
' #n -------------------------------',
' #n Slope Ratio: #s', 'ratio,
' #n Height of Embankment(ft): #s', 'height,
' #n Thickness of Foundation(ft): #s', 'depth,
' #n Water Table from the bottom of fill(ft): #s', 'water,
' #n Width of Top half(ft): #s', 'width,
' #n Number of Layers of Embankment: #s', 'numse,
' #n Thickness Each Layer Total Unit Weight #n').
unsgm = 1.
while unsgm <= unnumse
then write('pcset.in', '#s ', 'concat(segm, unsgm),
' '#s ', 'concat(segvall, unsgm), '#n') and
unsgm = unsgm + 1.

write('pcset.in', '#s Drainage Type: #s', 'type,
' '#n Consolidation Degree(%); #s', 'cond,
' '#n Construction Period(days): #s', 'time,
' '#n Number of Layers of Foundation: #s', 'numsf,
' '#n OCR TL TUW SUW Cc Cr Void Cv #n').
sftgw = 1.
while unsgw <= unnumsf
then write('pcset.in', '#s', 'concat(ocr, unsgw), '#s
' , 'concat(sfgvall, unsgw), 'concat(sfgval2, unsgw),
' 'concat(sfgval3, unsgw), 'concat(sfgval4, unsgw),
' 'concat(sfgval5, unsgw), 'concat(sfgval6, unsgw),
' 'concat(sfgval7, unsgw), '#n') and
unsgw = unsgw + 1.

write('pcset.in', '#n -------------------------------').

key_settle is 999.
sf_file_pos('settle.in', 0, beginning).
run('settle.exe').

sf_file_pos('settle2.out', 0, beginning).
[STOT, TP, conset, centime, consol, remset] is_c
read('settle2.out', '//SUNG', '//HWAN').
if ?ratio >= 4 then SFOS is ?OFOS else SFOS is 0.

use_font (?boldfont).
text('#e #x3#y2.5 #fbrown The Relationships of Elapsed
   Time and Settlement #d').
use_font( ).
set_file_pos('settle1.out', 0, beginning).
set_file_pos('settle2.out', 0, beginning).
run('setplt.exe').

bitmap(load_bitmap('boxl9.bmp'), 15, 7).
text('#xl8#y9 Please select the following option').
check_box('Recalculate Settlement Using Soil',
    recal_ssettle, 20, 11).
check_box('See Conclusion', soilcon, 20, 13).

topic recal_ssettle.
do(!main; soil_all: ssettle).
end.
topic mark(item).
helpwin is window(, 40, 8, 50, 18, ?item,
    [thinframe, showchildren, visible], ?main, ,
    lightgray, close_event).
button2('Close Me', close, 1, 1, 11, 1.3).
window (, 1, 2.3, 50, 15.5, [child, siblings, visible,
    vertscroll, showchildren], ?helpwin ).
set_file_pos('settle.txt', 0, beginning).
settle is read('settle.txt', concat('//', ?item), '/').
text('#e', ?settle).
end.
topic close.
close_window( ).
close_window( ).
show_window( ).
end.
end.(*ssettle*)
Conclusion with Soil

==----------------------------------------------------------------------------------
soilcon.kb
==----------------------------------------------------------------------------------*

topic soilcon.
new_file('ssumm.out').
material_evaluation is normal.
if ?material_evaluation is well then CFOS is 1.2.
if ?material_evaluation is normal then CFOS is 1.4.

con1 is 'o. No more modification'.
con2 is 'o. Steepen slope ratio'.
con3 is 'o. Increase stability'.
con4 is 'o. Reduce settlement'.
con5 is 'o. Place stiff layers using FGD By-products'.
con6 is 'o. Need foundation treatment'.
con7 is 'o. Increase Stability and Reduce Settlement'.
sugg1 is ' -.Use higher strength FGD materials'.
sugg2 is ' -.Use lighter weight FGD materials'.
sugg3 is ' -.Reduce water table'.
sugg4 is ' -.Use lower permeable FGD materials'.

FTUW= ?pemtot.
FUCU = ?pemcoh.
SSRE = ?ratio.
DF=?depth.

use_font(?boldfont).
text('#e #x7#y2.5 #fbrown Summary of the Numerical
Analysis #d').
use_font (  ).
bitmap(?box6,7,5).
text('#xl0#y6.5 o. Factor of Safety : #s',?OFOS,
'#xl0#y8 o. Maximum Slope Ratio #s',?ratio,
'#xl0#y9.5 o. Total Settlement during Life Time (ft) : #s',?stot,
'#xl0#y11 o. Time required for 90% consolidation(days) : #s',?tp,
'#xl0#y12.5 o. % Consolidation at End-Construction : #s',?consol,
'#xl0#y14 o. Remaining Settlement after End_Construction(ft) : #s',?remset).

(*========== Write for Report ================*)
write('ssumm.out','#n Summary of the Numerical Analysis',
'#n -------------------------------',
'#n o. Factor of Safety : #s',?OFOS,'#n o. Maximum Slope Ratio #s',?ratio,
'#n o. Total Settlement during Life Time (ft) : #s',?stot,'#n o. Time required for 90% consolidation(days): #s',?tp,'
Consolidation at End-Construction:

Remaining Settlement after End_Contruction(ft): 

\text{seratio} \text{ is } \text{?ratio}.

\text{sestot} \text{ is } \text{?stot}.

\text{button2('Press Here To Continue =>>',continue,55, 26,30,1.5).}

wait( ).

\text{use_font(\textbf{?boldfont}).}

\text{text('The Conclusion In Soil Fill\#d').}

\text{use_font( ).}

\text{bitmap(load_bitmap('box6. bmp'),5,1).}

\text{\textbf{\#x5\#yl5 #fbrown Factor of Safety : \#s',?OFOS, \#x7\#yl0 Total Settlement (ft) \#s', ?STOT, \#x7\#yl1 Total Unit Weight of Fill Soil (pcf): \#s',?FTUW, \#x7\#yl2 Cohesion of Fill Soil (psf): \#s',?FUCU, \#x7\#yl3 Side Slope Ratio: \#s',?ratio).}

\text{fill_type1 is "Full Thickness of SOIL".}

\text{fill_type2 is "Mixed Layers of Soil and FGD Waste".}

\text{\textbf{if } ?OFOS \geq ?CFOS \text{ and } \text{STOT} \leq 1 \text{ and } \text{FTUW} < 100 \text{ and } \text{SSRE} \leq 2 \text{ then text('\#x7\#y4',?con1) \text{ and write('sdesign.in',\#n,\#n,fill_type1) \text{ and bitmap(load_bitmap('dntype1.bmp'),45,4).})}

\text{\textbf{if } ?OFOS \geq ?CFOS \text{ and } \text{STOT} \leq 1 \text{ and } \text{FTUW} \geq 100 \text{ and } \text{SSRE} > 2 \text{ then text('\#x7\#y4',?con2,'\#x7\#y6',?sugg2) \text{ and write('sdesign.in',\#n,\#n,fill_type1) \text{ and bitmap(load bitmap('dntype2.bmp'),45,4).})}

\text{\textbf{if } ?OFOS \geq ?CFOS \text{ and } \text{STOT} \leq 1 \text{ and } \text{FUCU} < 10000 \text{ and } \text{SSRE} > 2 \text{ then text('\#x7\#y4',?con2,'\#x7\#y6',?sugg1) \text{ and write('sdesign.in',\#n,\#n,fill_type1) \text{ and bitmap(load_bitmap('dntype3.bmp'),45,4).})}

\text{\textbf{if } ?OFOS \geq ?CFOS \text{ and } \text{STOT} \leq 1 \text{ and } \text{FUCU} < 10000 \text{ and } \text{SSRE} > 2 \text{ then text('\#x7\#y4',?con2,'\#x7\#y6',?sugg1) \text{ and write('sdesign.in',\#n,\#n,fill_type1) \text{ and bitmap(load_bitmap('dntype3.bmp'),45,4).})}
and ?SSRE <= 2
then text('#x7#y4',?con1)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype1.bmp'),45,4).

if ?OFOS >= ?CFOS
   and ?STOT <= 1
   and ?FTUW >= 100
   and ?SSRE <= 2
then text('#x7#y4',?con1)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype1.bmp'),45,4).

if ?OFOS >= ?CFOS
   and ?STOT > 1
   and ?FTUW >= 100
then text('#x7#y4',?con6)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype6.bmp'),45,4).

if ?OFOS < ?CFOS
   and ?STOT <= 1
   and ?FTUW < 100
   and ?FUCU < 10000
then text('#x7#y4',?con3,'#x7#y6',?sugg1)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype5.bmp'),45,4).

if ?OFOS < ?CFOS
   and ?STOT <= 1
   and ?FTUW < 100
   and ?FUCU >= 10000
then text('#x7#y4',?con6)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype6.bmp'),45,4).

if ?OFOS < ?CFOS
   and ?STOT <= 1
   and ?FTUW < 100
   and ?FUCU < 10000
then text('#x7#y4',?con3,'#x8',?sugg1,'#x8',?sugg2)
   and write('sdesign.in','#n',?fill_type1)
   and bitmap(load_bitmap('dntype10.bmp'),45,4).

if ?OFOS < ?CFOS
   and ?STOT <= 1
and \( \text{FTUW} \geq 100 \) 
and \( \text{FUCU} \geq 10000 \) 

\[
\text{if } \text{OFOS} < \text{CFOS} \\
\text{and } \text{STOT} > 1 \\
\text{and } \text{FTUW} < 100 \\
\text{and } \text{FUCU} < 10000 \\
\text{then } \text{text}'(#x7#y4',?con3,'#x8',?sugg2) \\
\text{and write}'(\text{sdesign.in}',#'n',?fill_type1) \\
\text{and bitmap}(\text{load_bitmap}'(\text{dntype8.bmp}'),45,4). \\
\]

\[
\text{if } \text{OFOS} < \text{CFOS} \\
\text{and } \text{STOT} > 1 \\
\text{and } \text{FTUW} < 100 \\
\text{and } \text{FUCU} < 10000 \\
\text{then } \text{text}'(#x7#y4',?con6) \\
\text{and write}'(\text{sdesign.in}',#'n',?fill_type1) \\
\text{and bitmap}(\text{load_bitmap}'(\text{dntype6.bmp}'),45,4). \\
\]

\[
\text{if } \text{OFOS} < \text{CFOS} \\
\text{and } \text{STOT} > 1 \\
\text{and } \text{FTUW} > 100 \\
\text{and } \text{FUCU} > 10000 \\
\text{then } \text{text}'(#x7#y4',?con6) \\
\text{and write}'(\text{sdesign.in}',#'n',?fill_type1) \\
\text{and bitmap}(\text{load_bitmap}'(\text{dntype6.bmp}'),45,4). \\
\]

\[
\text{if } \text{OFOS} < \text{CFOS} \\
\text{and } \text{STOT} > 1 \\
\text{and } \text{FTUW} > 100 \\
\text{and } \text{FUCU} > 10000 \\
\text{then } \text{text}'(#x7#y4',?con7,'#x8',?sugg1,'#x8',?sugg2) \\
\text{and write}'(\text{sdesign.in}',#'n',?fill_type1) \\
\text{and bitmap}(\text{load_bitmap}'(\text{dntype9.bmp}'),45,4). \\
\]

\[
\text{if } \text{OFOS} < \text{CFOS} \\
\text{and } \text{STOT} > 1 \\
\text{and } \text{FTUW} > 100 \\
\text{and } \text{FUCU} > 10000 \\
\text{then } \text{text}'(#x7#y4',?con7,'#x8',?sugg2) \\
\text{and write}'(\text{sdesign.in}',#'n',?fill_type1) \\
\text{and bitmap}(\text{load_bitmap}'(\text{dntype9.bmp}'),45,4). \\
\]

\[
\text{bitmap}'(\text{box19},6,16). \\
\text{text}'(#x8#y18 o. Please select the following option.'). \\
\text{check_box}'(\text{Consider FGD Wastes as Fill Material}', \text{return_fgd},9,20). \\
\text{check_box}'(\text{See Final Design}',\text{sdesign},9,22). \\
\]

\[
\text{topic return_fgd.} \\
\text{do(!main:fgd_all:fgddb).} \\
\text{end.} \\
\text{end.(*soil_con*)}
Design for Embankment Using Soil

```
(* ===========================================================================
Design for Embankment Using Soil
sdesign.kb
 ===========================================================================*)

topic sdesign.
  use_font( ?bldfont).
  text( '
  use_font( ).
  set_file_pos( 'sdesign.in', 0, beginning).
  run( 'sdesign.exe' ).

window(, 15, 9, 48, 7, [visible, dialogwindow], ?main, , close_event).
  text( '
  button2('Press Here To Continue =>>', continue, 8, 4.5, 30, 1.5).
  wait( ).
  close_window( ).
  show_window( ).

text( ' Press the following button to continue').

button2('Press Here To Continue =>>', continue, 8, 4.5, 30, 1.5).
  wait( ).
  close_window( ).
  show_window( ).

text( '#e ').
  bitmap( ?box6, 2, 1.2).
  use_font( ?bldfont).
  text( '#x3#y2 #f brown Construction Procedures #d').
  use_font( ).
  text( '#x3#y3.7 o. Range of Slope : greater than
    #s', ?ratio,
    '#x3#y4.9 o. Total Unit Weight : #s', ?segval21, (pcf),
    '#x3#y7.3 o. Compaction Required in the field : 95 %',
    '#x3#y8.5 o. Compaction Thickness of each layer : 6 to 8
    inches',
    '#x3#y9.7 o. #m Compaction Equipment#m',
    '#x3#y10.9 o. #m Erosion Control#m',
    '#x3#y12.1 o. #m Compaction During Wet Weather#m',
    '#x3#y13.3 o. #m Construction During Freezing
    Weather#m').
  conpro is load_bitmap( 'const.bmp' ).
  bitmap( ?box7, 2, 16.3).
  bitmap( ?conpro, 5, 17).

button2( Report, report, 76, 23.5, 15, 1.5).
  button2( 'FGD Material', restart, 76, 25, 15, 1.5).
  button2( End, end, 76, 26.5, 15, 1.5).

*------------------------------------------------------------------------*

```

text('#e', ?design).

topic close.
  close_window( ).
  close_window( ).
  show_window(?main).
end.
end.(*mark*)

topic report.
  repwin is window(, 10,4,60,24, 'Report Format of the
  Consultation', [thinframe, showchildren, visible], ?main).
  button2(Material, soil_out, 1,1,12,1.3).
  button2('Stability', sstable_out, 13,1,12,1.3).
  button2('Settlement', ssettm, 25,1,12,1.3).
  button2('Summary', soil_cont, 37,1,12,1.3).
  button2('Close Me', close, 49,1,12,1.3).
  window (, 3,2,3,58,21.5,, [child, siblings, horzscroll,
  vertscroll, visible, showchildren], repwin ) .

  use_font(?boldfont).
  text('#e #x5#y5 #fred List and Print the Consultation
  Results.#d
  #x5#y7 Please select a button shown above.').
  use_font( ).

topic soil_out.
  set_file_pos('soilprot.out', 0, beginning).
  soil_summ is read('soilprot.out').
  text('#e', ?soil_summ).
  print('#x3#y2', ?soil_summ).
end.(*soil_out*)

topic sstable_out.
  text('#e #y3 Output Files of Stability Calculation').
  ssta_prot is radio_button([[Input File', 8,5,t],
    ['Summary of Stability Calculation', 8,6.5],
    ['Output File', 8,8],[Graphic of Failure
    Surface', 8,9.5]])).
  button2('Press Here Again =>>', continue, 5,18,
    30,1.5, move_up_event).
  wait( ).

  sstaout is sstablout(?ssta_prot).
  topic sstablout(handle).
    sstableout is element(?handle,
    where(get_radio_button(?ssta_prot), t)).
    ssta_out is get_text(?sstableout).
end.

  if ?ssta_out is 'Input File'
    then ssta_input( ).
if ?ssta_out is 'Summary of Stability Calculation'
then ssta_sum().

if ?ssta_out is 'Output File'
then ssta_output().

if ?ssta_out is 'Graphic of Failure Surface'
then ssta_graph().
end.(*sstable_out*)

topic ssettm.
text('#e #y3 Output Files of Settlement Calculation'),
ssett prot is radio_button([['Input File',8,5,t],
  ['Summary of Settlement Calculation',8,6.5],
  ['Output File',8,8], ['Graphic of Time - Settlement',8,9.5]]).
button2('Press Here Again =>>',continue,5,18,
  30,1.5,move_up_event).
wait( ).

ssetout is ssettout(?ssett_prot).
topic ssettout(handle).
ssettleout is element(?handle,
  where(get_radio_button(?ssett_prot),t)).
ssettle_out is get_text(?ssettleout).
end.

if ?ssettle_out is 'Input File'
then ssett_input().

if ?ssettle_out is 'Summary of Settlement Calculation'
then ssett_sum().

if ?ssettle_out is 'Output File'
then ssett_output().

if ?ssettle_out is 'Graphic of Time - Settlement'
then ssett_graph().
end.(*settle_out*)

topic ssta_input.
set_file_pos('slopel.in',0,beginning).
ssta_file is read('slopel.in').
text('#e',?ssta_file).
print(?ssta_file).
end.

topic ssta_sum.
text('#e Engineering Summary'),
print('#x3#y2 Engineering Summary').
end.
topic ssta_output.
    spcstaout is read('fort.out').
    text('\#e', spcstaout).
    (* print('\#x3\#y2', spcstaout)*)
end.

topic ssta_graph.
    text('\#e Engineering graphic').
    print('\#x3\#y2 Engineering Property').
end.

topic ssett_input.
    set_file_pos('pcset.in',0,beginning).
    ssett_file is read('pcset.in').
    text('\#e', ssett_file).
    print('\#n \#n\#n\#n\#n', ssett_file).
end.

topic ssett_sum.
    text('\#e Settlement Summary').
    (* print('\#x3\#y2 Engineering Summary').*)
end.

topic ssett_output.
    ssetfile_out is read('pcset.out').
    text('\#e', ssetfile_out).
    (* print('\#x3\#y2', ssetfile_out)*)
end.

topic ssett_graph.
    text('\#e Settlement graphic').
    (* print('\#x3\#y2 Engineering Property').*)
end.

(*====== All of Summary================*)

topic soil_cont.
    set_file_pos('ssumm.out',0,beginning).
    soil_summ is read('ssumm.out').
    text('\#e', soil_summ).
    print('\#x3\#y2', soil_summ).
end.

topic close.
    close_window( ).
    close_window( ).
    show_window(?main).
end.
end.(*report*)

topic end.
    clear( ).
end.
topic restart.
    do(!main:fgd_all:fgddb).
end.
end.(*design*)

(* The end of the Program for Path I *)
(*------------------------------------------------------------
Data Base for FGD By-product
dbase.kb
-------------------------------------------------------------*)

fgddb().
topic fgddb.
use_font(?boldfont).
Text('#e #x3#y1.5 #fbrown Embankment Material : FGD By-
Products#d').
use_font( ).
text('#x3#y3 o. Click one item from each box listed below.
').
cap1 is load_bitmap('cap1.bmp').
cap2 is load_bitmap('cap2.bmp').
cap3 is load_bitmap('cap3.bmp').
use_font( ).
SystemList is [
    Dry,
    Wet].

ProWetList is [
    Consville,
    'AEP Nol',
    'All of the plants'].

ProDryList is [
    AFBC,
    CLS,
    DUCT,
    LIMB,
    PFBC,
    SPD,
    Others,
    'All of the Processes'].

PlantLIMBLList is [
    Edgewater].

PlantSPDList is [
    McCracken,
    'EPRI HSTC',
    'Northern State Power',
    'All of the Plants'].

PlantDUCTList is [
    'Muskingum River'].

PlantCLSLList is [
    Edgewater].

PlantFBCLList is [
    'General Motors',
    'Quaker State',

'TVA Shawnee',
  Staley,
  'All of the Plants'].

PlantPFBCList is [
  Tidd].

PlantOthersList is [
  Rockport,
  Burger,
  PEDCO,
  'BW ARC'].

PlantALLList is [
  'BW ARC',
  Burger,
  'Edgewater',
  'EPRI HSTC',
  'General Motors',
  'McCracken',
  'Muskingum River',
  'Northern State Power',
  PEDCO,
  'Quaker State',
  Rockport,
  Staley,
  Tidd,
  'TVA Shawnee',
  'All of the Plants'].

(*==========================================================================*)
bitmap(?capl,3,4.5).
hyper_region(system,3,4.5,20,3.5).

Syscom is combo_box (?SystemList, , 3,7,30,7,,
[simple,vertscroll], double_click_event).
  caplbut is button2('Press Here
>>',continue,13,14,20,1.5).
  wait ( ),
  close_window(?caplbut).
  show_window( ).
Sysvalue is get_combo_box(?Syscom).
sysdisplay is ?Sysvalue.

if ?Sysvalue is Dry  then process is ?ProDryList and
dryplant( ).
if ?Sysvalue is Wet  then process is ?ProWetList and
wetplant( ).

topic wetplant.
Provalue is [ ].
Prodisplay is [ ].
bitmap(?cap3,50,4.5).
hyper_region(plant,50,4.5,22,3.5).
Plantcom is combo_box(?process,, 50,7,30,7,, [simple, vertscroll], double_click_event).
cap3wbut is button2('Press Here >>', continue,60,14,20,1.5).
wait ( ).
close_window(?cap3wbut).
Plantvalue is get_combo_box(?Plantcom).
Plantdisplay is ?Plantvalue.
if ?Plantvalue is 'All of the Plants' then Plantvalue is [ ].
dball( ).
end.(*wetplant*)
topic dryplant.
bitmap(?cap2,52,4.5).
hyper_region(process,52,4.5,22,3.5).
Procom is combo_box(?process,, 52,7,30,7,, [simple, vertscroll], double_click_event).
cap2but is button2('Press Here >>', continue,62,14,20,1.5).
wait ( ).
close_window(?cap2but).
show_window( ).
Provalue is get_combo_box(?Procom).
Prodisplay is ?Provalue.
if ?Provalue is LIMB then plant is ?PlantLIMBList.
if ?Provalue is SPD then plant is ?PlantSPDList.
if ?Provalue is 'DUCT' then plant is ?PlantDUCTList.
if ?Provalue is CLS then plant is ?PlantCLSList.
if ?Provalue is AFBC then plant is ?PlantPBCList.
if ?Provalue is PFBC then plant is ?PlantPFBCList.
if ?Provalue is Others then plant is ?PlantOthersList and
Provalue is [ ].
if ?Provalue is 'All of the Processes' then plant is
?PlantALLList and Provalue is [ ].

bitmap(?cap3,26,15).
hyper_region(plant,24.5,18,30,14).
Plantcom is combo_box(?plant,, 25,17.5,30,5,, [simple, vertscroll], double_click_event).
cap3but is button2('Press Here >>', continue,35,23,20,1.5).
wait ( ).
close_window(?cap3but).
Plantvalue is get_combo_box(?Plantcom).
Plantdisplay is ?Plantvalue.
if ?Plantvalue is 'All of the Plants' then Plantvalue is [ ].
dball( ).
end.(*dryplant*)
(*===================================================================
Operation Conditions
====================================================================*)
topic dball.
gsystem = ?Sysvalue.
gprocess = ?Provalue.
gplant = ?Plantvalue.

@kpdblib.tpx
(* =========== Operating Conditions =========*)
use_font(?boldfont).
text(#e  #fbrown Embankment Material: FGD By-Products #d').
use_font( ).
wLeads1 is window (.14,6.58,22, 'Operating Conditions',
[popup,visible,siblings,showChildren,titlebar,thinframe],?main ).
button2 ('>>',next,1,1,19,1.4 ).
button2 ('<<',back,20,1,19,1.4).
button2 (Continue,lea,39,1,20,1.4).

text('

Material Code:
Plant Name:
Location:
State:
Capacity(MW)
FGD System:
FGD Process:
Coal Type:
Sorbent Used:
Sampling Point:
Sampling Date:

wLeads3 is window (.20,3,40,17,,'[child,siblings,
showchildren,visible],?wLeads1).
list ( ).
topic List.
filter is concat(fsyst,' = "',?gsystem,'"').
if ?gprocess <> [ ] then
filter is concat (?filter, ' .and. fproc = "',
?gprocess,'"').
if ?gplant <> [ ] then
filter is concat (?filter, ' .and. pname = "',
?gplant,'"').

fieldList is [mcode,pname,ploct,psstat,capat,fsyst, fproc,coalt,sorbt,sampt,samdt].
EOF is number_to_char (26).

db_use_file ('mcode').
db_set_filter (?filter).
Records is [ ].
rec is db_top_record (?fieldlist).
while ?rec <> ?EOF then
    Records gets [ ?rec ] and
    rec is db_skip_record (, ?fieldlist).

index is 1.
set_text (?wLeads3, element (?Records, ?index)).
show_window (?wLeads1).
end. (*List*)

topic Next.
    index is ?index + 1.
    updateLeads ( ).
end.

topic Back.
    index is ?index - 1.
    updateLeads ( ).
end.

topic updateLeads.
    if ?index < 1 then
        index is 1.
    if ?index > list_length (?Records) then
        index is list_length (?Records).
    set_text (?wLeads3, element (?Records, ?index)).
end.

topic close.
    close_window( ).
    close_window( ).
    close_window( ).
end.
end. (*dball*)

topic system.
    dbwin is window(, 38,5,50,23, 'FGD System',
    [thinline,showchildren,visible],?main,
    ,lightgray,close_event).
    button2('Close Me',close1,1,1,11,1.3).
    syswin is window (,1,2.3,50,20.5,,
    [vertscroll,child,siblings,visible,showchildren],?dbwin).

    text(''
        #fblue What Is an FGD System ? #d
    
        The burning of coal in electric power plants
        results in the production of sulfur dioxide gas
        (SO2), which is considered to be a primary
        cause of acid rain. To reduce the emission
        of SO2 gas, sorbents such as lime or limestone
        are injected in the flue gas control systems.
    )
As a consequence of this SO2 emission control process, a solid waste material is generated. This waste is called a Flue Gas Desulfurization (FGD) by-product. There are two major classes of SO2 control systems available today: mWet Scrubbing\#m and mDry Scrubbing\#m.

There are several types of dry FGD processes classified by types of sorbent used and by the location where the sorbent is injected.

- **LIMB (Lime Injection Multistage Burner)**: Edgewater Plant
- **SPD (Spray Dryer)**: McCracken, EPRI HSTC, Northern State Power
- **DUCT (Duct Injection)**: Muskingum River Plant
- **CLS (Coolside)**: Edgewater Plant
- **AFBC (Atmospheric-Fluidized Bed Combustion)**: General Motors, Quaker State, TVA Shawnee and Staley Plants
- **PFBC (Pressurized Fluidized Bed Combustion)**: Tidd Plant
- **Others**: Rockport, Burger, PEDCO, B&W ARC Plants

All of the Processes include all plants shown above'.
[thinframe,showchildren,visible],?main, lightgray,close_event).
button2('Close Me',close1,1,1,11,1.3).
window (1.2.3,50,20.5,, vertscroll,child,siblings,visible,showchildren],?ptwin ).
text(' #fblue Plant Help will be developed soon #d').
end.
topic mark(item).
dbwin is window(25,7,50,23, 'FGD System', [thinframe,showchildren,visible],?main, lightgray,close_event).
markwin is window (1,1,2.3,50,20.5,, vertscroll,child,siblings,visible,showchildren],?dbwin ).
set_file_pos('dbase. txt',0,beginning).
dbase is read('dbase.txt',concat('//',?item),'//').
text('#e',?dbase).
window (38,1,6,2.8,,child,siblings,showchildren,visible],?markwin).
notefig is load_bitmap('note.bmp').
bitmap(?notefig).
end.(*mark*)
topic 'Figure 3'.
dbwin is window(10,10,47,15, 'FGD System', [thinframe,showchildren,visible],?main, lightgray,close_event).
markwin is window (1,1,2.3,47,12.5,, child,siblings,visible,showchildren],?dbwin ).
limb is load_bitmap('limbfig.bmp').
bitmap(?limb).
end.
topic close1.
close_window( ).
close_window( ).
close_window( ).
show_window( ).
end.
topic close2.
close_window( ).
close_window( ).
show_window( ).
end.(*fgddb*)
(* ============== Leachate Properties ================*)

FGD Properties

(* ============== Leachate Properties ================*)

FGD System : #s',?Sysdisplay,
FGD Process : #s',?Prodisplay,
Power Plant : #s', PPlantdisplay).

Number of Records ; #s', ?lnum,
unit: mg/L

Elements Average Standard Deviation

#s pH  ',?aph,'  ',?stdph,
Arsenic  ',?aars,'  ', ?stdars,
Barium  ',?abar,'  ',?stdbar,
Cadmium  ',?acad,'  ',?stdcad,
Chromium  ',?achr,'  ',?stdchr,
Lead  ',?alead,'  ',?stdlead,
Mercury  ',?amer,'  ',?stdmer,
Selenium  ',?asel,'  ',?stdsel,
Silver  ',?asil,'  ',?stdsil).

new_file('leach.in').
leach name is [Arsenic, Barium, Cadmium,
Chromium,Lead,Mercury,Selenium,Silver].
leach list is [?aars,?abar,?acad,?achr,?alead,?amer,
?asel,?asil].
write('leach.in',concat("",?gsystem,"),'\$
,concat("",?wprocess,"),'\$
,concat("",?leach_name,"),
'\$
, ?leach_list).

topic leach_graph.
  set_file_pos('leach.in').
  run('leach.exe').
end.

topic ph_graph.
  set_file_pos('ph.in').
  run('ph.exe').
end.

topic avglea.
aph=format_number('##.#',?aph).
aars=format_number('###.####',?aars).
abar=format_number('###.####',?abar).
acad=format_number('###.####',?acad).
achr=format_number('###.####',?achr).
alead=format_number('###.####',?alead).
amer=format_number('###.####',?amer).
asel=format_number('###.####',?asel).
asil=format_number('###.####',?asil).
end.

topic stand.
stdph=format_number('##.#',?stdph).
stdars=format_number('##.####',?stdars).
stdbar=format_number('##.####',?stdbar).
stdcad=format_number('##.####',?stdcad).
stdchr=format_number('##.####',?stdchr).
stdlead=format_number('##.####',?stdlead).
stdmer=format_number('##.####',?stdmer).
stdsel=format_number('##.####',?stdsel).
stdsil=format_number('##.####',?stdsil).
end.
end.(*lea*)

(* ============== Chemical Property ==============*)
topic chem.
  @kpdblib.tpx
  library is load_library('kpmath.dll').
  #include flyvar.ckb
  #include caco3var.ckb
  #include caovar.ckb
  #include caoh2var.ckb
  #include caso3var.ckb
  #include caso4var.ckb
  #include camg2var.ckb
  #include mgovar.ckb
  avgchem ( ).
  stdchem ( ).
chem_win is window(, 14, 6, 56, 22, 'Chemical Properties', [thinframe, showchildren, visible], main, lightgray, close_event).
button2('Chemical Components', chem_graph, 1, 1, 25, 1.3).
button2('Distribution', chemgrf, 26, 1, 15, 1.3).
button2('Continue', eng, 41, 1, 15, 1.3).
window(, 1, 2, 3, 56, 19.5, [child, siblings, visible, showchildren], chem_win).
new_file('chem.in').
text('Number of Records : #s', cnum).
text('Elements Average Standard Deviation').
text('# Fly Ash(%)', afly, stdfly, CaCO3(%)', acaco, stdcaco, CaO(%)', acao, stdcao, Ca(OH)2(%)', acaoh, stdcaoh, CaSO3(%)', acast, stdcast, CaSO4(%)', acasf, stdcasf, CaMg2(%)', acamg, stdcamg, MgO(%)', amgo, stdmgo).
text('if ?gprocess is [ ] then wprocess is All.
if ?afly is 0 then afly is [ ] and flyn is [ ] else afly is ?afly and flyn is 'FlyAsh'.
if ?acaco is 0 then acaco is [ ] and cacon is [ ] else acaco is ?acaco and cacon is 'CaCO3'.
if ?acao is 0 then acao is [ ] and caon is [ ] else acao is ?acao and caon is CaO.
if ?acaoh is 0 then acaoh is [ ] and caohn is [ ] else acaoh is ?acaoh and caohn is 'Ca(OH)2'.
if ?acast is 0 then acast is [ ] and castn is [ ] else acast is ?acast and castn is CaSO3.
if ?acasf is 0 then acasf is [ ] and casfn is [ ] else acasf is ?acasf and casfn is CaSO4.
if ?acamg is 0 then acamg is [ ] and camgn is [ ] else acamg is ?acamg and camgn is CaMg2.
if ?amgo is 0 then amgo is [ ] and mgon is [ ] else amgo is ?amgo and mgon is MgO.
chem_name is [flyn, cacon, caon, caohn, castn, casfn, camgn, mgon].
chem_list is [afly, acaco, acao, acaoh, acast, acasf, acamg, amgo].
xw is ttotal(?chem_list).
topic ttotal (chem_list).
count_num = 0.
apply (ad, ?chem_list).

\[ ttotal = ?count_num. \]

\[ \text{topic ad(xw).} \]

\[ \text{if } ?xw \text{ is } [ ] \text{ then } wq \text{ is } 0 \text{ else } wq \text{ is } 1. \]

\[ \text{count_num } = ?count_num + ?wq. \]

end.

write('chem.in', concat("", ?gsystem, ","), concat("", ?wprocess, "),


\[ \text{topic chem_graph.} \]

\[ \text{set_file_pos('chem.in').} \]

\[ \text{run('chem.exe').} \]

end.

\[ \text{topic chemgrf.} \]

\[ \text{cgwin is window(,4,3,85,27, 'Distribution of Chemical} \]

\[ \text{Components',[thinframe,showchildren,visible],?main, ,} \]

\[ \text{lightgray,close_event}).} \]

\[ \text{button2('Close Me',close,1,1,11,1.3).} \]

\[ \text{window (,1,2,3,85,24.5,, [child,siblings,visible,} \]

\[ \text{showchildren],?cgwin )).} \]

\[ \text{limb is load_bitmap('limb.bmp').} \]

\[ \text{duct is load_bitmap('ductl.bmp').} \]

\[ \text{pfbc is load_bitmap('pfbc.bmp').} \]

\[ \text{spd is load_bitmap('spdl.bmp').} \]

\[ \text{bitmap(?limB,3,1).} \]

\[ \text{bitmap(?spd,45,1).} \]

\[ \text{bitmap(?duct,3,14).} \]

\[ \text{bitmap(?pfbc,45,14).} \]

\[ \text{topic close.} \]

\[ \text{close_window( ).} \]

\[ \text{close_window( ).} \]

end.

end.*(chemgrf*)

describe avgchem.

\[ \text{afly=format_number('##.#',?afly).} \]

\[ \text{acaco=format_number('##.#',?acaco).} \]

\[ \text{acao=format_number('##.#',?acao).} \]

\[ \text{acaoh=format_number('##.#',?acaoh).} \]

\[ \text{acast=format_number('##.#',?acast).} \]

\[ \text{acasf=format_number('##.#',?acasf).} \]

\[ \text{acamg=format_number('##.#',?acamg).} \]

\[ \text{amgo=format_number('##.#',?amgo).} \]

end.

describe stdchem.

\[ \text{stdfly=format_number('##.#',?stdfly).} \]

\[ \text{stdcaco=format_number('##.#',?stdcaco).} \]

\[ \text{stdcao=format_number('##.#',?stdcao).} \]
stdcaoh=format_number('##.#',?stdcaoh).
stdcast=format_number('##.#',?stdcast).
stdcasf =format_number('##.#',?stdcasf).
stdcamg=format_number('##.#',?stdcamg).
stdmgo=format_number('##.#',?stdmgo).
end.
end.(*chem*)

(*===== Engineering Property ===============*)
topic eng.
@kpdblib.tpx
library is load_library('kpmath.dll').
#include permvar.ckb
#include ucsvar.ckb
#include denvar.ckb
#include swellvar.ckb
#include omcvar.ckb
#include fricvar.ckb
#include cohvar.ckb

eng_win is window (,14,6,56,22, 'Engineering Properties',
[popup,visible,siblings, showChildren,
titlebar,thinframe],?main ).
button2('OMC', omc,1,1,11,1.3).
button2('DUW', duw,12,1,11,1.3).
button2('UCS', ucs,23,1,11,1.3).
button2('Perm',perm,34,1,11,1.3).
button2('Swell', swell,45,1,12,1.3).
button2('Cohesion', cohesion,1,12,1,1.3).
button2('Friction', friction,19,2,18,1.3).
button2('Continue',dvalue,37,2,20,1.3).
window (,1,4,3,56,19.5,[child,siblings, showchildren,
visible],?eng_win).
avgeng ( ).
engstand ( ).
text('#e
FGD System :  #s', ?Sysdisplay,
' #n FGD Process :#s ', ?Prodisplay,
' #n Power Plant :  #s', ?Plantdisplay).
text(' #n Number of Records :  #s', ?enum).
text(' #n -------------------------------------------------').
text(' #n Elements Average Standard Deviation').
text(' #n -------------------------------------------------').
text(' #n Permeability #s ',?aperm,' ', ?stdperm,
' #n (cm/sec)',
' #n UCS(psf) ',?aucs,' ', ?stducms,
' #n Swell(%) ',?aswell,' ', ?stdswell,
' #n Dry Density(pcf) ',?aden,' ', ?stddden,
' #n OMC(%) ',?aomc,' ', ?stdomc,
' #n Friction Angle(deg) ',?acufric,' ', ?stdcufric,
' #n Cohesion(psf) ',?acucoh,' ', ?stdcucoh).
text(' #n -------------------------------------------------').
topic avgeng.
   aperm=format_number('#.##########',?aperm).
   aucs=format_number('#####.#',?aucs).
   aden=format_number('###.#',?aden).
   aswell=format_number('###.#',?aswell).
   aomc=format_number('###.#',?aomc).
   acufric=format_number('###.#',?acufric).
   acucoh=format_number('#####.#',?acucoh).
end. (*dbengcal*)

topic engstand.
   stdperm=format_number('#.##########',?stdperm).
   stducs=format_number('#####.#',?stducs).
   stdswell=format_number('###.#',?stdswell).
   stddden=format_number('###.#',?stddden).
   stdomc=format_number('###.#',?stdomc).
   stdcufric=format_number('###.#',?stdcufric).
   stdcucoh=format_number('#####.#',?stdcucoh).
end.

topic omc.
   #include omcvar.kb
   set_file_pos('omc.in').
   run('omc.exe').
end. (*omc*)

topic duw.
   #include denvar.kb
   set_file_pos('den.in').
   run('den.exe').
end. (*duw*)

topic ucs.
   #include ucsvar.kb
   set_file_pos('ucs.in').
   run('ucs.exe').
end. (*ucs*)

topic perm.
   #include permvar.kb
   set_file_pos('perm.in').
   run('perm.exe').
end. (*perm*)

topic swell.
   #include swellvar.kb
   set_file_pos('swell.in').
   run('swell.exe').
end. (*swell*)

topic friction.
   #include fricvar.kb
set_file_pos('fric.in').
run('fric.exe')
end.(*fric*)

topic cohesion.
#include cohvar.kb
set_file_pos('coh.in').
run('coh.exe')
end.(*cohesion*)

end.(*eng*)
Define default values

Default.kb

==========================================================================
dvalue( ).
topic dvalue(item,handle).

@kpdblib.tpx
dewin is window( , 14,6,54,22, 'Default Values From Database', [thinline,showchildren,visible],?main, ,
lgray,close_event).
button2 ('Continue',leach,1,1,11,1.3 ).
window (,1,2.3,54,19.5,, [child,siblings,visible, showchildren],?dewin ).

use_font(?boldfont).
text('#e This Screen determines the Default Values for the Following Inputs.
Please select one of the ranges#m').
use_font ( ).
value is radio_button([[Mean - Standard Deviation',6,6],
 ['Mean',6,8],[Mean + Standard Deviation',6,10],
 ['By Manual',6,12],[A Specific Material',6,14]] ,
period).

markwin is window(, 43,5,45,23, 'Default Values Help', [thinline,showchildren,visible],?main, ,
lgray,close_event).
button2 ('Close Me',close,1,1,11,1.3 ).
window (,1,2.3,45,20.5,, [child,siblings,visible, showchildren],?markwin ).
set_file_pos('default. txt',0,beginning).
defat is read('default. txt',concat('//',?item),'//').
text('#e',?defat).
end.(***mark*)
topic close.
close_window( ).
close_window( ).
show_window(?dewin).
end.

period( value).
period is ?value.
if ?period is Mean+Standard Deviation' then meanpls( ).
if ?period is Mean-Standard Deviation' then meanmus( ).
if ?period is Mean or ?period is 'By Manual' then mean( ).
if ?period is 'A Specific Material' then specmat( ).

meanpls.
aars = ?aars + ?stdars.
acad = ?acad + ?stdcad.
alead = ?alead + ?stdlead.
amer = ?amer + ?stdmer.
aph = ?aph + ?stdph.
afly = ?afly + ?stdfly.
acaco = ?acaco + ?stdcaco.
acast = ?acast + ?stdcast.
aucs = ?aucs + ?stducs.
ad = ?aden + ?stdden.
acufric = ?acufric + ?stdcufric.
avglea( ).
avgchem( ).
avgeng( ).
end.

topic meanmus.
acad = ?acad - ?stdcad.
ad = ?aden - ?stdden.
\text{acufric} = \text{?acufric} - \text{stdcufric}.
\text{avglea}( ).
\text{avgchem}( ).
\text{avgeng}( ).
\text{end}.

\text{topic mean}.
\text{aars} = \text{?aars}.
\text{abar} = \text{?abar}.
\text{acad} = \text{?acad}.
\text{achr} = \text{?achr}.
\text{alead} = \text{?alead}.
\text{amer} = \text{?amer}.
\text{asel} = \text{?asel}.
\text{asil} = \text{?asil}.

\text{aph} = \text{?aph}.
\text{afly} = \text{?afly}.
\text{acaco} = \text{?acaco}.
\text{acao} = \text{?acao}.
\text{acaoh} = \text{?acaoh}.
\text{acast} = \text{?acast}.
\text{acasf} = \text{?acasf}.
\text{acamg} = \text{?acamg}.
\text{amgo} = \text{?amgo}.

\text{aucs} = \text{?aucs}.
\text{aden} = \text{?aden}.
\text{aperm} = \text{?aperm}.
\text{aswell} = \text{?aswell}.
\text{aomc} = \text{?aomc}.
\text{acucoh} = \text{?acucoh}.
\text{acufric} = \text{?acufric}.
\text{avglea}( ).
\text{avgchem}( ).
\text{avgeng}( ).
\text{end}.

\text{topic avglea}.
\text{aars} = \text{format\_number'}(#.#####', \text{?aars}).
\text{abar} = \text{format\_number'}(#.#####', \text{?abar}).
\text{acad} = \text{format\_number'}(#.#####', \text{?acad}).
\text{achr} = \text{format\_number'}(#.#####', \text{?achr}).
\text{alead} = \text{format\_number'}(#.#####', \text{?alead}).
\text{amer} = \text{format\_number'}(#.#####', \text{?amer}).
\text{asel} = \text{format\_number'}(#.#####', \text{?asel}).
\text{asil} = \text{format\_number'}(#.#####', \text{?asil}).
\text{end}.

\text{topic avgchem}.
\text{aph} = \text{format\_number'}(##.#', \text{?aph}).
\text{afly} = \text{format\_number'}(##.#', \text{?afly}).
\text{acaco} = \text{format\_number'}(##.#', \text{?acaco}).
\text{acao} = \text{format\_number'}(##.#', \text{?acao}).
\text{acaoh} = \text{format\_number'}(##.#', \text{?acaoh}).
acast=format_number('##.#',?acast).
acasf=format_number('##.#',?acasf).
acamg=format_number('##.#',?acamg).
amgo=format_number('##.#',?amgo).
end.
topic avgeng.
aucs=format_number('#####.#',?aucs).
aden=format_number('###.#',?aden).
aperm=format_number('#####.#',?aperm).
aswell=format_number('##.#',?aswell).
aomc=format_number('###.#',?aomc).
acufric=format_number('##.#',?acufric).
acucoh=format_number('#####.#',?acucoh).
end. (*avgeng*)
topic specmat(item,handle).
close__window( ).
specwin is window(, 14,6,58,22, 'Default Values From Database', [thickness,showchildren,visible],?main, , close_event).
use_font(?!boldfont).
text ('#e Please select engineering properties listed below and Press ENTER Key.').
use_font( ).
list_box([['Dry Unit Weight','Unconfined Compressive Strength']], select,5,5,35,3 ,T).
topic select (item,event,handle).
set_error_topic(mtcode:error).
EOF is number_to_char (26).
selection is get_list_box (?handle).
if ?selection is 'Dry Unit Weight'
then select1( ).
if ?selection is 'Unconfined Compressive Strength'
then select2( ).
if ?selection is ['Dry Unit Weight','Unconfined compressive Strength']
then select4 ( ).
topic select1(item).
subwin is window(, 14,6,58,22, 'Default Values From Database', [thickness,showchildren,visible],?main, , lightgray,close_event).
button2 ('Continue',continue,1,11,1.4 ).
window (,1,2.3,58,19.5,, [child,siblings,visible, showchildren],?subwin ).
use_font(?!boldfont).
text('#e Choose a range to define the criteria').
use_font( ).
range is radio_button([[ 'Dry Unit Weight > A',6,4],
[A > Dry Unit Weight',6,6] ],ransell).
topic ransell(range).
ransell is ?range.
if ?ransell is 'Dry Unit Weight > A' then qql( ).
if ?ransell is 'A > Dry Unit Weight' then qq2( ).

topic qql.
text(' #x3#y13 Dry Unit Weight (pcf) > ').
sell is set_focus(edit_line (0,,26,12.7,10)).
wait( ).
dsell is get_text(?sell).
dfilter is concat('dryden >',?dsell).
mtcode( ). (*material code selection*)
dbqwin( ).
text('#x2#y3.5 To see the detailed material properties, Press Table Button
#x3#y15 The Material Codes shown above have the range:
#x10#y16 #fgreen2 Dry Unit Weight (pcf) > #s ',
?dsell,'#d').
scom is combo_box(?mcode,, 10,5,30,10,[simple,
vertscroll], double_click_event).
wait ( ).
svalue is get_combo_box(?scom).
dindfilter is concat(mcode,' = ''',?svalue,'''').
default( ).
end.(*qql*)

topic qq2.
text(' #x17#y13 > Dry Unit Weight (pcf)').
sel2 is set_focus(edit_line (0,,6,12.7,10)).
wait( ).
dsel2 is get_text(?sel2).
dfilter is concat('dryden <',?dsel2).
mtcode( ). (*material code selection*)
dbqwin( ).
text('#x2#y3.5 To see the detailed material properties, Press Table Button
#x3#y15 The Material Codes shown above have the range:
#x10#y16 #fgreen2 #x16#y16 > Dry Unit Weight (pcf) #d').
scom is combo_box(?mcode,, 10,5,30,10,[simple,
vertscroll], double_click_event).
wait ( ).
svalue is get_combo_box(?scom).
dindfilter is concat(mcode,' = ''',?svalue,'''').
default( ).
end.(*qq2*)
end.(*ransell*)
end.(*selectl*)

topic select2.
subwin is window(, 14,6,58,22, 'Default Values From
Database', [thinframe,showchildren,visible],?main, ,
lightgray,close_event).
button2 ('Continue',continue,1,1,11,1.4).
window (,1,2.3,58,19.5,, [child,siblings,visible, showchildren],?subwin).
use_font(?boldfont).
text('#e Choose a range to define the criteria').
use_font( ).
range2 is radio_button([['Unconfined Compressive Strength > A',6,4],['A > Unconfined Compressive Strength',6,6]],ransel2).

topic ransel2(range2).
  ransel2 is ?range2.
  if ?ransel2 is 'Unconfined Compressive Strength > A' then rqql( ).
  if ?ransel2 is 'A > Unconfined Compressive Strength' then rqq2( ).

topic rqql.
  text('#x5#y13 Unconfined Compressive Strength > ').
  rsell is set_focus(edit_line (0,,40,12.7,10)).
  wait( ).
  rdssel is get_text(?rsell).
  dfilter is concat('ucs >',?rdssel).
  mtcode( ). (*material code selection*)
  dbgwin( ).
  text('#x2#y3.5 To see the detailed material properties, Press Table Button
  #x5#y15 The Material Codes shown above have the range : #fgreen2 Unconfined Compressive Strength (psf) > #s ',?rdssel,'#d').
  scom is combo_box(?mcode,, 10,5,30,10,,[simple, vertscroll],double_click_event).
  wait ( ).
  svalue is get_combo_box(?scom).
  indfilter is concat(mcode,' = "',?svalue,'"').
  indfault( ).
end.(*rqql*)

topic rqq2.
  text('#x17#y13 > Unconfined Compressive Strength').
  sel2 is set_focus(edit_line (100000,,6,12.7,10)).
  wait ( ).
  dsell is get_text(?sel2).
  dfilter is concat('ucs < ',?dsell).
  mtcode( ). (*material code selection*)
  dbgwin( ).
  text('#x2#y3.5 To see the detailed material properties, Press Table Button
  #x3#y15 The Material Codes shown above have the range : #fgreen2
  #x5#y16 #s', ?dsell, '#xll#y16 > Unconfined
  Compressive Strength').
scom is combo_box(?mcode,, 10,5,30,10,,[simple,
vertscroll],double_click_event).
wait ( ).
svalue is get_combo_box(?scom).
indfilter is concat(mcode,' = ''?svalue,'''').
indfault( ).
end.(*rqq2*)
end.(*ransel2*)
end.(*select2*)
topic select4.
subwin is window(, 14,6,58,22, 'Default Values From
Database', [thinframe,showchildren,visible],?main,,
lightgray,close_event).
button2 ('Continue',continue,1,1,11,1.3 ).
window (,1,2.3,58,19.5,, [child,siblings,visible,
showchildren],?Subwin ).
use_font(?boldfont).
text('#e Choose a range to define the criteria').
use_font( ).
range is radio_button([['Dry Unit Weight > A and
Strength > A',4,4], ['A > Dry Unit Weight and Strength
> A',4,6] ],ransel4).

topic ransel4(range).
ransel4 is ?range.
if ?ransel4 is 'Dry Unit Weight > A and
Strength > A'
then sr4ql( ).
if ?ransel4 is 'A > Dry Unit Weight and Strength > A'
then sr4q2( ).

topic sr4ql.
text(' #x3#y10 Dry Unit Weight (pcf) > ').
sr4duw is set_focus(edit_line (0,,26,9.7,10)).
text(' #x3#y12 Strength (psf) > ').
sr4ucs is edit_line (0,,22,11.7,10).
wait( ).
sr4guw is get_text(?sr4duw).
sr4gucs is get_text(?sr4ucs).
dfilter is concat('dryden >',?sr4duw).
dfilter is concat(?dfilter,' .and. ucs >',?sr4gucs).
mtcode( ).
dbqwin( ).
text('#x2#y3.5 To see the detailed material properties,
Press Table Button
#x3#y15 The Material Codes shown above have the range :
#x10#y16 #fgreen2 Dry Unit Weight (pcf) > #s ',
?sr4duw,
'#x3#y17 Unconfined Compressive Strength (psf) > #s ',
?sr4gucs,'#d').
scom is combo_box(?mcode,, 5,5,30,10,,[simple,
vertscroll],double_click_event).
wait (  ).
    svalue is get_combo_box(?scom).
    indfilter is concat(mcode,' = "',?svalue,'"').
    indfault(  ).
end.(*sr4q1*)

topic sr4q2.
    text(' #xl7#yl0 > Dry Unit Weight (pcf)').
    sr4duw is set_focus(edit_line (100,,6,9.7,10)).
    text(' #x3#yl2 Strength (psf) >').
    sr4ucs is edit_line (0,,22,11.7,10).
    wait(  ).
    sr4gdw is get_text(?sr4duw).
    sr4gucs is get_text(?sr4ucs).
    dfilter is concat('dryden <',?sr4gdw).
    dfilter is concat(?dfilter,' .and. ucs >',?sr4gucs).
    mcode(  ).
    dbqwin(  ).

text('#x1#y3.5 To see the detailed material properties,
    Press Table Button
#x3#yl5 The Material Codes shown above have the range :
    #fgreen2 #x13#yl6 #s',?sr4gdw, '#x18#yl6 > Dry Unit
Weight (pcf)', '#x3#yl7 Unconfined Compressive
    Strength (psf) > #s ',?sr4gucs,'#d'
).
    scom is combo_box(,mccde, ,  5,5,30,10,,[simple,
    vertscroll1,double_click_event).
    wait (  ).
    svalue is get_combo_box(?scom).
    indfilter is concat(mcode,' = "',?svalue,'"').
    indfault(  ).
end.(*sr4q2*)
end.(*ransel4*)
end.(*select4*)
end.(* select *)
end.(*specmat*)
end.(* period *)

topic mtcode.
    db_use_file ('FGDALL').
    db_set_filter (?dfilter).
    db_top_record(  ).
    mcode is db_locate_record(,,,[mcode]).
    db_remove_filter(?dfilter).
    topic error(mcode).
        if ?mcode = ?EOF then mcode is No.
    end.
    if ?mcode is [ ] then
        mcode is 'No Data File for The Range.
            Press Reselect Button'.
    end.(*mtcode*)
topic dbqwin.
  close_window()
  close_window()
  subwin is window(, 14,6,58,22, 'Database Selected', [thinframe,showchildren,visible],?main, , lightgray,close_event).
  button2(Table,dbufault,1,1,11,1.4).
  button2(Reselect,reelect,12,1,11,1.4).
  button2('Continue',continue,23,1,11,1.4).
  window (,1,2,3,58,19.5,, [child,siblings,visible, showchildren],?subwin ).
  use_font(?bolfont).
  text('#e#x4#yl. 5 Please select a specific material to be used').
  end.

end.

topic reselect.
  close_window()
  close_window()
  do(!main:fgd_all:dvalue:period:specmat)
end.

topic dbfault.
  wLeadsl is window (, 40, 5, 45, 22, 'Data Base', [popup, visible,titlebar,siblings,showChildren,thinframe],?main ).
  button2('>>',next,1,1,11,1.4).
  button2('<<',back,12,1,11,1.4).
  button2('Close Me',close,23,1,11,1.4).
  text('Material Code:
Plant Name:
Location:
State:
Capacity(MW)
FGD System:
FGD Process:
Coal Type:
Sorbent Used:
PH:
OMC (%):
Dry Density(pcf):
Strength (psf):
Swell (%):
Permeability(c/s):').

  wLeadsl is window (,20,4,30,17,,[child,siblings, showchildren,visible],?wLeadsl).
  list ()

end.

topic List.
fieldList is [mcode,pname,ploct,pstat,capat,fsyst,fproc,coalt,sorbt,ph,omc,dryden,ucs,swel,perm].
EOF is number_to_char (26).

db_use_file ('FGDALL').
db_set_filter (?dfilter).
db_top_record().
mcode is db_locate_record(,,,[mcode]).

Records is []
rec is db_top_record (?fieldlist).
while ?rec <> ?EOF then
  Records gets [?rec] and
  rec is db_skip_record (,?fieldlist).

index is 1.
set_text (?wLeads3, element (?Records, ?index)).
show_window (?wLeads1).
end.(*List*)

topic Next.
  index is ?index + 1.
  updateLeads ( ).
end.

topic Back.
  index is ?index - 1.
  updateLeads ( ).
end.

topic updateLeads.
  if ?index < 1 then
    index is 1.
  if ?index > list_length (?Records) then
    index is list_length (?Records).
  set_text (?wLeads3, element (?Records, ?index)).
end.
db_remove_filter(?dfilter).
wait( ).
end.(*dbfault*)

topic indfault.
fgdlist is [omc,perm,ucs,dryden,swel,cufr,cuco].
db_use_file ('engdb').
db_set_filter (?indfilter).
db_top_record().
sysdisplay is db_locate_record (,,,[fsyst]).
db_top_record().
prodisplay is db_locate_record (,,,[fproc]).
db_top_record().
plantdisplay is db_locate_record (,,,[pname]).
fgdrex is db_top_record(?fgdlist).
indval gets(?fgdrex).
aomc is element (?indval,1).
aperm is element(?indval,2).
aucs is element (?indval,3).
aden is element(?indval,4).
aswell is element(?indval,5).
acufric is element(?indval,6).
acuh is element(?indval,7).

db_remove_filter(?indfilter).
button2('Press Here Again',leach,5,18,30,1.5).

end.(*indfault*)
end.(*dvalue*)
Material Properties of Leachate, Chemical, & Engineering property.kb

(*== Leachate Properties ==*)
topic leach(item,info,handle).
mainsetup( ).
new_file('fgdeng.in').
use_font(?boldfont).
text('#e #x3#y1 #fbrown Leachate Properties #d').
use_font( ).
text('#x3#y2.5 Enter your data in the following
questions (unit: mg/L):
#x5#y3.5 The Default Values are obtained from Data Base
#x7#y4.5 FGD System : #s',?Sysdisplay,
'#x7#y4.5 FGD Process : #s',?Prodisplay,
'#x7#y5.5 Power Plant : #s', ?Plantdisplay,
'#x8#y8 Arsenic: #x8#y10Barium: #x8#y12Cadmium: #x8#y14Chromium: #x35#y8Lead: #x35#y10Mercury: #x35#y12Selenium: #x35#y14Silver:').
bitmap(?boxl7,4,17).
text('#x6#y7.5 Note:
A primary concern with the use of FGD wastes in any
construction activity is the potential for leaching
hazardous chemical components, and the consequent
pollution of ground and surface water near FGD-based
structures. A waste is categorized as hazardous when
the extract obtained by EP or TCLP test has
concentrations of any of the regulated metals in excess
of the EPA criteria').
pi is set_focus( edit_line (?aars,,20,8,10)).
p2 is edit_line(?abar,,20,10,10).
p3 is edit_line(?acad,,20,12,10).
p4 is edit_line(?achr,,20,14,10).
p5 is edit_line(?alead,,48,8,10).
p6 is edit_line(?amer,,48,10,10).
p7 is edit_line(?asel,,48,12,10).
p8 is edit_line(?asil,,48,14,10).
button2('Press Here To Continue =>>',continue,55,26,30,1.5,,mouse_up_event).
wait( ).
ars is get_text(?p1).
bar is get_text(?p2).
cad is get_text(?p3).
chr is get_text(?p4).
lead is get_text(?p5).
mer is get_text(?p6).
ser is get_text(?p7).
sil is get_text(?p8).

(*Intermediate Conclusion *)
if ?ars > 5 or ?bar >100 or ?cad > 1 or ?chr >5 or
?lead >5 or ?mer >0.2 or ?sel >1 or ?sil >5
then leachcon is ' . Your material is over the EPA
Leachate Criteria.'
else leachcon is ' . Your material is within the EPA
Leachate Criteria.'.

(*====================== Chemical Properties ===============*)
use_font(#boldfont).
text('#e #x3#y1 #fbrown Chemical Properties #d').
use_font( ).
text('#x3#y2.5 Enter your data for the following
questions:#x3#y3.5 The Default Values are obtained
from Data Base #x5#y4.5 FGD System : #s',?Sysdisplay,
'#x37#y4.5 FGD Process : #s',?Prodisplay,
'#x5#y5.5 Power Plant : #s', ?Plantdisplay).
text(' #x5#y7 PH : #x5#y9 Fly Ash (%): #x5#y11 CaO (%):
#x5#y13 Ca(OH)2 (%): #x5#y15 CaCO3 (%): #x35#y7 CaSO3
(%): #x35#y9 CaSO4 (%): #x35#y11 CaMg2 (%): #x35#y13 MgO
(%):').
bitmap(?boxl7,4,17. 5
text('#x6#y8 Note:
#xlO Please type
#xlO If no values
if ?aph is [  ]
then aph = concat(0,.,0).
if ?afly is [ ]
then afly = concat(0,.,0).
if ?acao is [ ]
then acao = concat(0,.,0).
if ?acah2 is [ ]
then acah2 = concat(0,.,0).
if ?acaco is [ ]
then acaco = concat(0,.,0).
if ?acast is [ ]
then acast = concat(0,.,0).
if ?acasf is [ ]
then acasf = concat(0,.,0).
if ?acamg is [ ]
then acamg = concat(0,.,0).
if ?amgo is [ ]
then amgo = concat(0,.,0).

ph is  set_focus( edit_line (?aph,,20,7,10)).
flyah is edit_line(?afly,,20,9,10).
cao is edit_line(?acao,,20,11,10).
caoh2 is edit_line(?acah2,,20,13,10).
caco3 is edit_line(?acaco,,20,15,10).
caso3 is edit_line(?acast,,50,7,10).
caso4 is edit_line(?acasf,,50,9,10).
camg2 is edit_line(?acamg,,50,11,10).
mgo is edit_line(?amgo,,50,13,10).
button2('Press Here To Continue =>>',continue,55,26,
30,1.5).
wait ( ).
cph is get_text(?ph).
ccao is get_text(?cao).
ccao is get_text(?ccao).
ccas3 is get_text(?ccaso3).
ccaso4 is get_text(?ccaso4).
pcao_cah2 is ?ccao + ?ccao.
psso is ?ccaso3 + ?ccaso4.

if ?cph >= 12.0 then conph is ' . Your material may cause Corrosion problem in the structures'
else conph is [ ].

if ?pcao_caoh2 > 15 then chemstr is ' . Your material may increase long term strength'
else chemstr is [ ].

if ?psso >= 20 then psotl is ' . Your material may have Long Term Swelling Potential' and psot2 is ' . Your material may cause Corrosion problem in the structures'
else psotl is [ ] and psot2 is [ ].

(*========= Engineering Property ==================*)
use_font(?boldfont).
text('#e #x3#y3#f brown Engineering Properties #d').

use_font(  ).
text('#x3#y2.5 Enter #fgreen your data #d for the following questions: #x3#y3.5 The Default Values are obtained from Data Base. #x5#y4.5 FGD System : #s',?Sysdisplay, #x37#y4.5 FGD Process : #s',?Prodisplay,'#x5#y5.5 Power Plant : #s',
?Plantdisplay).
text('#x4#y7 #mOMC (%)#m #x4#y9 #mDry Unit Weight(pcf)#m #x4#y11 #mUCS (psf)#m #x4#y13 #mPermeability(cm/sec)#m #x4#y15 #mSwell (%)#m #x40#y7 #mCohesion (psf)#m #x40#y9 #mFriction Angle (deg)#m #x40#y11 #mConsolidation Coef(ft2/day)#m #x40#y13 #mCompression Index#m').

bitmap(?boxl7,4,17.5).
text('#x6#y8 Note: #xlO Please type values in the boxes. #x10 If no values, leave blank or default value.').

omc is set_focus( edit_line (?aomc,,25,7,10)).
duw is edit_line(?aden,,25,9,10).
ucs is edit_line(?aucs,,25,11,12).
perm is edit_line(?aperm,,25,13,13).
swell is edit_line(?aswell,,25,15,10).
coh is edit_line(?acucoh,,67,7,12).
Fric is edit_line(?acufric,,67,9,10).
Conso is edit_line(,,67,11,10).
Comp is edit_line(,,67,13,10).

button2('Press Here To Continue =>>',continue,55,26,30,1.5).
wait(  ).
cduw is get_text(?duw).
cucs is get_text(?ucs).
cswell is get_text(?swell).
cperm is get_text(?perm).
comc is get_text(?omc).
ccoh is get_text(?coh).
cfric is get_text(?fric).
cconso is get_text(?conso).
ccomp is get_text(?comp).

write('fgdeng.in' ,"#n Engineering Property of FGD By-
Product',"#n -------------------------------',
'"#n FGD System : #s',?sysdisplay,
'"#n FGD Process : #s',?prodisplay,
'"#n Power Plant : #s', ?plantdisplay,
'"#n OMC (%): #s',?comc,
'"n Dry Unit Weight(pcf): #s', ?cduw,
'"n UCS (psf): #s', ?cucs,
'"n Permeability(cm/sec): #s', ?cperm,
'"n Swell (%): #s',?cswell,
'"n Cohesion (psf): #s', ?ccoh,
'"n Friction Angle (deg): #s',?cfric,
'"n Consolidation Coef(ft2/day): #s',?cconso,
'"n Compression Index: #s',?ccomp,
'"n -------------------------------').

if ?cswell >= 5 then swell_potential is '. Your
material has high swelling potential'
else swell_potential is [ ].
if ?cperm >= 0.00001 then conperm is '. Your material
may increase Leaching Quantity'
else conperm is [ ].

topic mark(item).
helpwin is window( , 37,5,50,22, ?item,
[thinframe,showchildren,visible],?main, ,lightgray,
close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,50,19.5,,[child,vertscroll,siblings,
visible,showchildren],?helpwin ).
use_font( ).
set_file_pos('property. txt',0,beginning).
prot is read('property.txt',concat('//',?item),'//').
text(' #e',?prot).
end.
topic close.
close_window( ).
close_window( ).
show_window(?main).
end.
topic Figure.
helpwin is window(, 5,13,40,14, 'Property Help',
[thinframe,showchildren,visible],?main, ,
lightgray,close_event).
button2('Close Me',close,1,1,11,1.3).
window (,1,2.3,40,11.5,,[child,siblings,visible,
showchildren],?helpwin ).
bitmap(load_bitmap('omcden.bmp')).
end.
topic Diagram.

helpwin is window(5,6,60,19.5,'Property Help',
    [thinlineframe,showchildren,visible],?main,,lightgray,
    close_event).
button2('Close Me',close,11,1,11,1.3).

window (1,2.3,60,17,,[child,siblings,visible,
    showchildren],?helpwin).
bitmap(load_bitmap('ptmodel.bmp')).

end.
text('#e').
savewin is window(15,7,45,9,'IDSSHED Data Base',
    [controlmenu,thickframe,visible,showchildren],?main,
    ,close_event).
text('#x5#y2 Do you wish to save your current inputs
    on The Source File of Data Base ?').
check_box('YES','savedb',10,5).
check_box('NO',continue,10,6.5).

topic savedb.
text('#e').
text(' #x2#y2 Save Your Inputs:C:\Kpwin2\June\Eng.dbf').
button2('Save',continue,10,5,12,1.8,mouse_up_event).
button2('Cancel',continue,23,5,12,1.8,mouse_up_event).

end.
wait ( ).
close_window(?savewin).
use_font(?boldfont).
text('##x3#y1 #fbrown Intermediate Conclusions for
    Material Properties #d').
bitmap(?box15,3,2.5).
bitmap(?box15,3,8.5).
bitmap(?box15,3,16).
use_font(?boldfontm).
text('#x5#y3 Leachate Properties
    #x5#y9 Chemical Properties
    #x5#y16.5 Engineering Properties').
use_font( ).
cduw =format_number('###.#',?cduw).
cucs =format_number('######.#',?cucs).
text('#x5#y4', ?leachcon, '#x5#y10 .pH : #s',?cph,'#s
    ','?conph,'#x5#y11. Free Lime (%): #s',?pcao_caoh2,'#s
    ','?chemstr,'#x5#y13 . CaSO3 +CaSO4 (%): #s',
    '?ps0','?n#x10',?psot1,'#n#x10',?psot2,'#x5#y17.5 . Dry
    Unit Weight (pcf) : #s ',?cduw,' (Natural Material is
    90 - 130 pcf)', '#x5#y18.5 . UCS(psf) : #s', ?cucs, '
    (Natural Material is Less than 4300 psf)','#x5#y19.5 .
    Swelling Potential (%) : #s', ?cswell, '#s',
    '?swell Potential','#x5#y20.5 . Permeability(cm/sec) :
    #s', ?cperm,'#x15#y21.5 ',?conperm).
button2('Press Here To Continue =>>',found,55,26,30,1.5).
end.*leach*)
Data Base for Soil

found.kb

topic found.
@kpdplib.tpx
use_font(?boldfont).
text('#e #x3#yl#fbrown Data Base For Soils #d').
use_font (  ) .
bitmap(?box10,45,2).

Text('#x46#y3 o. Soil Types by #mODOT Method#m
#x47 A-1-a, A-1-b
#x47 A-3, A-3a
#x47 A-2-4, A-2-5, A-2-6, A-2-7
#x47 A-4a, A-4b, A-5
#x47 A-6a, A-6b
#x47 A-7-5, A-7-6
#x47 OL, OH, Pt, None
#x49 #mUnified Method#m').

bitmap(?box11,6,15).
text('#x3#y3 o. Please input soil types for the layers
#y4 (choose the type from the Table)
#x7#y5 -. Soil Type for Embankment
#x7#y9.5 -. Soil type for Foundation
#x7#yl5 Note:
#x7 o. If one layer of FGD material in Fill,
#x7 type #fgreen None #d in the box.
#x7 o. If more than two layers in Fill,
#x7 type a dominant soil name in the box.
#x7 o. If more than 2 layers in foundation,
#x7 type a dominant soil name in the box.').

emsoiltp is set_focus(edit_line(None,,10,7,10)).
ewin is button2('Continue =>>',continue,25,7,15,1.5).
wait(  ).
close_window(?ewin).

emsoil is get_text(?emsoiltp).
if ?emsoil is 'A-1-a' or ?emsoil is 'A-1-b' or
?emsoil is 'A-3' or ?emsoil is 'A-3a' or
?emsoil is 'A-2-4' or ?emsoil is 'A-2-5' or
?emsoil is 'A-2-6' or ?emsoil is 'A-2-7' or
?emsoil is 'A-4a' or ?emsoil is 'A-4b' or
?emsoil is 'A-5' or ?emsoil is 'A-6a' or
?emsoil is 'A-6b' or ?emsoil is 'A-7-5' or
?emsoil is 'A-7-6' or ?emsoil is OL or
?emsoil is OH or ?emsoil is Pt or ?emsoil is None
then scode is ?emsoil
else soil_errorl(  ).

filter is concat(scode,' = "',?scode,'"').
emsoillist is [stuw,ssuw,ucu,dcu,ufa,dfa].
db_use_file ('soildb.dbf').
db_set_filter (?filter).
rec is db_top_record (?emsoilList).
 esoil gets(?rec).

estuw is element(?esoil,1).
estuw is element(?esoil,2).
estuw is element(?esoil,3).
estuw is element(?esoil,4).
estuw is element(?esoil,5).
estuw is element(?esoil,6).

fdsoiltp is set_focus(edit_line ('A-5',,10,11,10)).

button2('Press Here To Continue =>>',continue,55,26,30,1.5).

wait( ).

if ?fdsoil is 'A-1-a' or ?fdsoil is 'A-1-b' or ?fdsoil is 'A-3' or ?fdsoil is 'A-3a' or ?fdsoil is 'A-2-4' or ?fdsoil is 'A-2-5' or ?fdsoil is 'A-2-6' or ?fdsoil is 'A-2-7' or ?fdsoil is 'A-4a' or ?fdsoil is 'A-4b' or ?fdsoil is 'A-5' or ?fdsoil is 'A-6a' or ?fdsoil is 'A-6b' or ?fdsoil is 'A-7-5' or ?fdsoil is 'A-7-6' or ?fdsoil is OL or ?fdsoil is OH or ?fdsoil is Pt then fcode is ?fdsoil else soil_error2( ).

wait( ).

filter is concat(scode,' = "',?fcode,'"').

fdsoiltp is set_focus(edit_line ('A-5',,10,11,10)).

wait( ).

if ?fdsoil is 'A-1-a' or ?fdsoil is 'A-1-b' or ?fdsoil is 'A-3' or ?fdsoil is 'A-3a' or ?fdsoil is 'A-2-4' or ?fdsoil is 'A-2-5' or ?fdsoil is 'A-2-6' or ?fdsoil is 'A-2-7' or ?fdsoil is 'A-4a' or ?fdsoil is 'A-4b' or ?fdsoil is 'A-5' or ?fdsoil is 'A-6a' or ?fdsoil is 'A-6b' or ?fdsoil is 'A-7-5' or ?fdsoil is 'A-7-6' or ?fdsoil is OL or ?fdsoil is OH or ?fdsoil is Pt then fcode is ?fdsoil else soil_error2( ).

wait( ).
soilwin is window(, 40,12,50,9,, [dialogwindow,visible],
    ?main, , ,).
text(' @ Make sure you type correct name of Soil Type.
    See Table shown above again and retype it.').
emsoiltp is set_focus(edit_line(None,,10,4,10)).
button2('Close Me',Continue,35,6.5,10,1.5).
wait(  ).
scode is get_text(?emsoiltp).
close_window(?soilwin).
show_window(  ).
end.-

topic soil_error2.
soilwin is window(, 40,12,50,9,, [dialogwindow,visible],
    ?main, , ,).
text(' @ Make sure you type correct name of Soil Type.
    See Table shown above again and retype it.').
fdsoiltp is set_focus(edit_line('A-7-6',,10,4,10)).
button2('Close Me',Continue,35,6.5,10,1.5).
wait(  ).
scode is get_text(?fdsoiltp).
close_window(?soilwin).
show_window(  ).
end.-

use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis #d').
use_font(  ).
tesa is load_bitmap('boxl2.bmp').
bitmap(?tesa,5,10).
text(' #x5#y3.5 o. Please select A Type of Slope
    Stability Analysis
    #x7#y11.2 Total Stress Analyses: is used to estimate
    the end-of-construction state with the
    undrained conditions. (c = Su and phi = 0)
    Effective Stress Analyses: is used to
    estimate the long-term state with the drained
    conditions.(c = c and phi = phi)').
tieback is load_bitmap('tieback.bmp').
reinfor is load_bitmap('reinfor.bmp').
noneload is load_bitmap('noneload.bmp').
svalue is radio_button([['Total Stress Analyses',
    8,5,t],[['Effective Stress Analyses',8,6.5]]]).
button2('Press Here To Continue =》',continue,55,26,
    30,1.5,move_up_event).
wait(  ).

styp is stype(?svalue).
topic stype(handle).
   typel is element(?handle,
      where(get_radio_button(?svalue),t)).
   stype is get_text(?typel).
end.

if ?stype is 'Total Stress Analyses'
then esscu is ?esucu and essfa is ?esufa and
   fsscu is ?fsucu and fssfa is ?fsufa.
if ?stype is 'Effective Stress Analyses'
then esscu is ?esdcu and essfa is ?esdfa and
   fsscu is ?fsdcu and fssfa is ?fsdfa.

estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.
estuw = ?estuw*10.
estuw = ?estuw div 1.
estuw = ?estuw/10.

estuw = ?estuw*10 mod 10.
if ?pestuw is 0 then estuw is concat(?estuw,'.').
else estuw = ?estuw.

essuw = ?essuw*10 mod 10.
if ?pessuw is 0 then essuw is concat(?essuw,'.').
else essuw = ?essuw.

esscu = ?esscu*10 mod 10.
if ?pesscu is 0 then esscu is concat(?esscu,'.').
else esscu = ?esscu.

essfa = ?essfa*10 mod 10.
if ?pessfa is 0 then essfa is concat(?essfa,'.').
else essfa = ?essfa.

(* Initial Input for Foundation Soils *)
fstuw = ?fstuw*10.
fstuw = ?fstuw div 1.
fstuw = ?fstuw/10.
fstuw = ?fstuw*10.
fstuw = ?fstuw div 1.
fstuw = ?fstuw/10.
\[ fsscu = \frac{fsscu \times 10}{1} \]
\[ fsscu = \frac{fsscu}{10} \]
\[ fssfa = \frac{fssfa \times 10}{1} \]
\[ fssfa = \frac{fssfa}{10} \]
\[ pfstuw = \frac{pfstuw \times 10 \mod 10}{1} \]
\[ \text{if } \text{pfstuw is } 0 \text{ then } \text{fstuw is } \text{concat}(\text{fstuw},'.'), \text{else } \text{fstuw} = pfstuw. \]
\[ pfssuw = \frac{pfssuw \times 10 \mod 10}{1} \]
\[ \text{if } \text{pfssuw is } 0 \text{ then } \text{fssuw is } \text{concat}(\text{fssuw},'.'), \text{else } \text{fssuw} = pfssuw. \]
\[ pfsscu = \frac{pfsscu \times 10 \mod 10}{1} \]
\[ \text{if } \text{pfsscu is } 0 \text{ then } \text{fsscu is } \text{concat}(\text{fsscu},'.'), \text{else } \text{fsscu} = pfsscu. \]
\[ pfssfa = \frac{pfssfa \times 10 \mod 10}{1} \]
\[ \text{if } \text{pfssfa is } 0 \text{ then } \text{fssfa is } \text{concat}(\text{fssfa},'.'), \text{else } \text{fssfa} = pfssfa. \]

\[ \text{(* FGD for embankment *)} \]
\[ atuw = \frac{aden \times (1+\frac{aomc}{100})}{10} \]
\[ atuw = \frac{atuw\times 10}{1} \]
\[ atuw = \frac{atuw \div 1}{10} \]
\[ acucoh = \frac{acucoh \times 10}{1} \]
\[ acucoh = \frac{acucoh \div 1}{10} \]
\[ acufric = \frac{acufric \times 10}{1} \]
\[ acufric = \frac{acufric \div 1}{10} \]
\[ ptuw = \frac{ptuw \times 10 \mod 10}{1} \]
\[ \text{if } \text{ptuw is } 0 \text{ then } \text{atuw is } \text{concat}(\text{atuw},'.'), \text{else } \text{atuw} = ptuw. \]
\[ acoh = \frac{acucoh \times 10 \mod 10}{1} \]
\[ \text{if } \text{acoh is } 0 \text{ then } \text{acucoh is } \text{concat}(\text{acucoh},'.'), \text{else } \text{acucoh} = acoh. \]
\[ afric = \frac{acufric \times 10 \mod 10}{1} \]
\[ \text{if } \text{afric is } 0 \text{ then } \text{acufric is } \text{concat}(\text{acufric},'.'), \text{else } \text{acufric} = afric. \]

\[ \text{*========Consider loading conditions =========*} \]
\[ \text{use_font}(\text{?boldFont}). \]
text('#e #fbrown Slope Stability Analysis #d, #mPCSTABL6#m').

use_font( ).

text('#x3#y2.5 Which Loading Conditions are you considering in the calculation?').

bitmap(?tieback,5,4).
bitmap(?reinfor,45,4).
bitmap(?noneload,5,15).

Lvalue is radio_button(['Tieback',5,13], ['Reinforcement',45,13], ['None of Them',5,24,t],sload).

topic sload(Lvalue,smethod).

sload is ?Lvalue.
if ?sload is 'Tieback' then ssload is [ ] and Spencert( ).
if ?sload is 'Reinforcement' then ssload is 'Reinforcement' and Bishopr( ).
if ?sload is 'None of Them' then ssload is [ ] and anameth( ).

end.

topic anameth.

use_font(?boldFont).

text('#e #fbrown Slope Stability Analysis').

use_font( ).

text('#x5#y3 o. Please select a Analysis Method of Slope Stability').

mvalue is radio_button(['Simplified Bishop Method',8,5,t],['Simplified Janbu Method',8,7], ['Spencer Method',8,9 ], smethod).

bitmap(?tesa,40,4).

text('#x50#y4.5 Limit Equilibrium Failure Shapes
#x46 Moment Vertical Horizontal Available
#x41 SBM o o x Cir,Spc
#x41 SJM o o o Cir, Irr, Blk, Spc
#x41 SPM o o o Cir, Irr, Blk, Spc
#x41 ------------------------------------------------
#x41 o: satisfied, x: not satisfied
#x41 . Cir: Circular Failure Surface,
#x41 . Irr: Irregular Failure Surface,
#x41 . Blk: Block Failure Surface,
#x41 . Spc: A Specific Failure Surface').

/topic smethod(mvalue).

smethod is ?mvalue.
if ?smethod is 'Simplified Bishop Method' then nmethod is ?smethod and Bishop( ).
if ?smethod is 'Simplified Janbu Method' then nmethod is ?smethod and Janbu( ).
if ?smethod is 'Spencer Method' then nmethod is ?smethod and Janbu( ).
button2('Press Here Again =>>', fstable,55,26,30,1.5).

end.
topic Bishop.

```
text(' #x5#y16 o. Select a Failure Shape In #s',
  ?smethod).
scom is combo_box([Circular, Specific],,10,17.5,27,4,,
  [simple, vertscroll], double_click_event).
bbt is button2(Continue, Continue, 5, 26, 15, 1.5).
wait ( ).
close_window(?bbt).
svalue is get_combo_box(?scom).
if ?svalue is Circular then ssmeth is CIRCL2.
if ?svalue is Specific then ssmeth is SURFIS.
end.
```

delimited_paragraph(topic Janbu.

```
text(' #x5#y16 o. Select a Failure Shape In #s',
?smethod).
scom is combo_box([Circular, Irregular, Specific],
  10,17.5,27,6,,[simple, vertscroll], double_click_event).
jbt is button2(Continue, Continue, 5, 26, 15, 1.5).
wait ( ).
close_window(?jbt).
svalue is get_combo_box(?scom).
if ?svalue is Circular then ssmeth is CIRCLE,
if ?svalue is Irregular then ssmeth is RANDOM,
if ?svalue is Specific then ssmeth is SURFAC.
end.
end.(*anameth*)
```

delimited_paragraph(topic Bishopr.

```
text('#e #fbrown Slope Stability Analysis #d').
use_font( ).
text('#x3#y3 Bishop Method is available for The
Reinforcement #x5#y5 o. Select a Failure Shape In
Bishop Method').
scom is combo_box([Circular],,10,7,27,4,,[simple, vertscroll], double_click_event).
bsbt is button2(Continue, Continue, 5, 26, 25, 1.5).
wait ( ).
close_window(?bsbt).
nmethod is 'Simplified Bishop Method'.
svalue is get_combo_box(?scom).
if ?svalue is Circular then ssmeth is CIRCL2.
button2('Press Here Again =>>', fstable, 55, 26, 30, 1.5).
end.
```

delimited_paragraph(topic Spencert.

```
text('#e #fbrown Slope Stability Analysis #d').
use_font( ).
```
text('#x3#y3 Spencer Method is available for Tieback Load #x3#y5 o. Select a Failure Shape In Spencer Method').
scom is combo_box([Circular, Irregular], ,10,7,27,6,,[simple, vertscroll],double_click_event).
spbt is button2(Continue, Continue,5,26,25,1.5).
wait ( ).
close_window(?spbt).
nmethod is 'Spencer Method'.
svalue is get_combo_box(?scom).
  if ?svalue is Circular then ssmeth is CIRCLE.
  if ?svalue is Irregular then ssmeth is RANDOM.
button2('Press Here To Continue =>>',fstable,55,26,30,1.5).
end.
topic mark(item).
irow is 45 and icol is 3 and ewid is 43 and eht is 27.
helpwin is window(,,?irow,?icol,?ewid,?eht, ?item,
  [thinframe,showchildren,visible],?main,,,close_event).
button2('Close Me',close,1,1,11,1.3).
set_file_pos('stable.txt',0,beginning).
stable is read('stable.txt',concat('//',?item),'//').
text('#x2#y2.5',?stable).
if ?item is PCSTABL6 then
  bitmap(load_bitmap('stable.bmp'),2,14).
end.
topic close.
close_window( ).
show_window( ).
end.(*found*)
Slope stability with FGD material

STAB.KB

topic fstable.
new_file('slope1.in').
new_file('error.out').
new_file('slope1.out').
new_file('slope2.out').
new_file('slope3.out').
new_file('slope4.out').
new_file('fdesign.in').
new_file('fort.out').

zero_int is concat(0,0).
use_font( ?boldFont).
text('#e #fbrown Slope Stability Analysis #d').
use_font( ).
text('#x5#y2.5 ( R: Real Number, I : Integer )
#x2#y4 @ #mEmbankment Boundary#m
#n #x3#y5 Total Number of Boundaries ( I )
#n #x3#y6.5 Number of Surface Boundaries ( I )').

ibnum is 4.
isufnum is 3.
if ?bdll <> [ ] then ibnum is ?numbdo and isufnum is ?numsufo.
numbdi is set_focus(edit_line(?ibnum,,36,4.8,4)).
umsufi is edit_line(?isufnum,,36,6.3,4).
bitmap(load_bitmap('shapl.bmp'),2,7.5).
small1 is button2(Continue,Continue,42,5,15,1.5).
wait( ).
close_window(?small1).

numbdo is get_text(?numbdi).
umsufo is get_text(?numsufo).
if ?numbdo >16 then bderror( ).
if ?numbdo >16 then numbdo = 16.
topic bderror.
bdwin is window(, 40,12,50,6, [dialogwindow,visible] , ?main, , ).
text(' @ Maximum Allowable Number of Total
 Boundaries are 16.').
button2('Close Me',Continue,35,4.5,10,1.5).
wait( ).
close_window(?bdwin).
show_window( ).
end.
comcardo = PROFIL.
titleo is ?project.

s=1.
while ?s <= 16
then concat(bint1,?s) is concat(0,.0,0) and
concat(bint2,?s) is concat(0,.0,0) and
concat(bint3,?s) is concat(0,.0,0) and
concat(bint4,?s) is concat(0,.0,0) and
concat(bint5,?s) is 1 and

s =?s+1.

if ?bdll <> [ ] then initial_input4( ).

if ?numbdo >= ?exnum then minnum is ?exnum and maxnum is
?numbdo.
if ?numbdo < ?exnum then minnum is ?numbdo and maxnum is
?exnum.

sss=1.
while ?sss <= ?minnum
then concat(bint1,?sss) is ?concat(bd1,?sss) and
concat(bint2,?sss) is ?concat(bd2,?sss) and
concat(bint3,?sss) is ?concat(bd3,?sss) and
concat(bint4,?sss) is ?concat(bd4,?sss) and
concat(bint5,?sss) is ?concat(bd5,?sss) and

sss =?sss+1.
end.(* *)

exnum is ?numbdo.

m=8.
yaxis=8m - 2.
par=1.
if ?numbdo > 10 then m=3.5 and yaxis = ?m - 2.

text(concat('#x55#y',?yaxis),'Coordinates of Boundary
(unit: ft) #x55 LX(R) LY(R) RX(R) RY(R) MT(I) ' ).

par=1.
while ?par <= ?numbdo
then text('#n#x52',?par) and
concat(valb1,?par) is set_focus(edit_line
(7concat(bint1,?par),,55,?m,6)) and
set_focus(?valb1) and
concat(valb2,?par) is edit_line
(7concat(bint2,?par),,61,?m,6) and
concat(valb3,?par) is edit_line
(7concat(bint3,?par),,67,?m,6) and
concat(valb4,?par) is edit_line
(7concat(bint4,?par),,73,?m,6) and
concat(valb5,?par) is edit_line
(7concat(bint5,?par),,79,?m,6) and
m =?m+1.5 and
par =?par +1.
ww is button2(Continue,continue,40,26,12,1.5).
wait( ).
close_window(?ww).
write('slopel.in', ?comcardo,?titleo,?numbdo,'#s ',
?numsufo).
write('fdesign.in',?numbdo).
pp = 1.
while ?pp <= ?numbdo
then concat(bdl,?pp) is get_text(?concat(valb1,?pp)) and
concat(bd2,?pp) is get_text(?concat(valb2,?pp)) and
concat(bd3,?pp) is get_text(?concat(valb3,?pp)) and
concat(bd4,?pp) is get_text(?concat(valb4,?pp)) and
concat(bd5,?pp) is get_text(?concat(valb5,?pp)) and
ppp = 1.
if ?bunmo is 0 then bunmo is 1.
ratio is (?bd32 - ?bdl2)/?bunmo.
ratio =format_number('##.#',?ratio).
while ?ppp <= ?numbdo
then write('slopel.in','#n',?concat(bdl,?ppp),'#s ',
?concat(bd2,?ppp),
?concat(bd3,?ppp), ?concat(bd4,?ppp),
?concat(bd5,?ppp)) and
write('fdesign.in','#n',?concat(bdl,?ppp),'#s ',
?concat(bd2,?ppp),?concat(bd3,?ppp),?concat(bd4,?ppp))
and ppp=?ppp+1.

(* ====== Input for Soil Types ===============*)
comsoil =SOIL.
emrow = 10.
fdrow = 18.
use_font(?boldFont).
text('#e #fbrown Slope Stability Analysis #d').
use_font( ).
text('#x4#y3 @ Material Properties ( R: Real Number, I : Integer )
#x5#y4.5 Number of Materials in Embankment (I):
#x5#y6 Number of Materials in Foundation (I):').
umemsoili is set_focus(edit_line(l,,45,4.3,4)).
umfdsoili is edit_line(l,,45,5.8,4).
small3  is button2(Continue,Continue,40,7.5,15,l).
wait( ).
close_window(?small3).
umemsoil is get_text(Pnumemsoili).
umfdsoil is get_text(Pnumfdsoili).
if Pnumemsoil >3 or Pnumfdsoil > 4 then soil_errorl( ).
if Pnumemsoil >3 then numemsoil =  3 .
if Pnumfdsoil >4 then numfdsoil =  4 .
topic soil_errorl.
soilwin is window( , 40,7.5,15,1.5),
wait( ).
close_window(?small3).
umemsoil is get_text(?numemsoili).
umfdsoil is get_text(?numfdsoili).
if ?numemsoil >3 or ?numfdsoil > 4 then soil_errorl( ).
if ?numemsoil >3 then numemsoil = 3.
if ?numfdsoil >4 then numfdsoil = 4.
topic soil_errorl.
soilwin is window( , 40,12,50,8,, [dialogwindow,visible] , ,?main, ,).  
text(' @ Maximum Allowable Number of Material Types in Embankment are 3.')
Maximum Allowable Number of Material Types in Foundation are 4'.

button2('Close Me',Continue,35,6.5,10,1.5).
wait(  ).
close_window(?soilwin).
show_window(  ).
end.

efs=1.
while ?efs <= 4
then  concat(emint1,?efs) is ?atuw and
    concat(emint2,?efs) is ?atuw and
    concat(emint3,?efs) is ?acucoh and
    concat(emint4,?efs) is ?acufric and
    concat(fdint1,?efs) is ?fstuw and
    concat(fdint2,?efs) is ?fssuw and
    concat(fdint3,?efs) is ?fsscu and
    concat(fdint4,?efs) is ?fssfa and
    efs = ?efs+1.
end.

efgd=2.
while ?efgd <= 4
then  concat(emint1,?efgd) is ?estuw and
    concat(emint2,?efgd) is ?essuw and
    concat(emint3,?efgd) is ?esscu and
    concat(emint4,?efgd) is ?essfa and
    efgd = ?efgd+1.
end.

if ?fdsolll <> [ ] then soil_initial(  ).
topic soil_initial.

emx=1.
while ?emx <= ?numemsoil
then  concat(emint1,?emx) is ?concat(emsoll1,?emx) and
    concat(emint2,?emx) is ?concat(emsol2,?emx) and
    concat(emint3,?emx) is ?concat(emsol3,?emx) and
    concat(emint4,?emx) is ?concat(emsol4,?emx) and
    emx = ?emx+1.
end.

fdx=1.
emint11 is ?atuw.
emint21 is ?atuw.
emint31 is ?acucoh.
emint41 is ?acufric.

while ?fdx <= ?numfdsoil
then  concat(fdint1,?fdx) is ?concat(fdsoil1,?fdx) and
    concat(fdint2,?fdx) is ?concat(fdsoil2,?fdx) and
    concat(fdint3,?fdx) is ?concat(fdsoil3,?fdx) and
    concat(fdint4,?fdx) is ?concat(fdsoil4,?fdx) and
    fdx = ?fdx+1.
end.

text('#x5#y8 Embankment Materials (Real Number)
#x10#y9#mMUW#m(pcf)#mSUW#m(pcf)#mCoh#m(psf)
#mPRic#m(deg)#mPPP#m(I)#mPPC#m(I)#mPSN#m(I)').
text('#x5#y16 Foundation Materials')
numsoil is ?numemsoil + ?numfdsoil.

em=1.
while ?em <= ?numemsoil
then text(concat('#x3#y',?emrow),'MT #s',?em) and
concat(emval1,?em) is edit_line
(?concat(emint1,?em),10,?emrow,10) and
set_focus(?emval1) and
concat(emval2,?em) is edit_line
(?concat(emint2,?em),20,?emrow,10) and
concat(emval3,?em) is edit_line
(?concat(emint3,?em),30,?emrow,12) and
concat(emval4,?em) is edit_line
(?concat(emint4,?em),42,?emrow,8) and
concat(emval5,?em) is edit_line
(?zero_int,,50,?emrow,8) and
concat(emval6,?em) is edit_line
(?zero_int,,58,?emrow,8) and
concat(emval7,?em) is edit_line (1,66,?emrow,8)
and emrow =?emrow+2
and em =?em +1.

fd=1.
while ?fd <= ?numfdsoil
then text(concat('#x3#y',?fdrow),'MT #s',?numemsoil+?fd) and
concat(fdval1,?fd) is edit_line
(?concat(fdint1,?fd),10,?fdrow,10) and
concat(fdval2,?fd) is edit_line
(?concat(fdint2,?fd),20,?fdrow,10) and
concat(fdval3,?fd) is edit_line
(?concat(fdint3,?fd),30,?fdrow,12) and
concat(fdval4,?fd) is edit_line
(?concat(fdint4,?fd),42,?fdrow,8) and
concat(fdval5,?fd) is edit_line
(?zero_int,,50,?fdrow,8) and
concat(fdval6,?fd) is edit_line
(?zero_int,,58,?fdrow,8) and
concat(fdval7,?fd) is edit_line (1,66,?fdrow,8)
and fdrow =?fdrow+2
and fd =?fd +1.

button2('Press Here To Continue =>>', Continue, 55, 26, 30, 1.5).
wait(  ).
write('slopel.in','#n',Pcomsoil,'#n',Pnumsoil) .
emtot is 0.
emcoh is 0.
ep = 1.
while ?ep <= ?numemsoil
then concat(emsol1,?ep) is get_text(?concat(emval1,?ep))
    and concat(emsol2,?ep) is get_text(?concat(emval2,?ep))
    and concat(emsol3,?ep) is get_text(?concat(emval3,?ep))
    and concat(emsol4,?ep) is get_text(?concat(emval4,?ep))
    and concat(emsol5,?ep) is get_text(?concat(emval5,?ep))
    and concat(emsol6,?ep) is get_text(?concat(emval6,?ep))
    and concat(emsol7,?ep) is get_text(?concat(emval7,?ep))
    and emtot = ?concat(emsol1,?ep) + ?emtot
    and emcoh = ?concat(emsol3,?ep) + ?emcoh
    and ep = ?ep + 1.

    pemtot = ?emtot/?numemsoil.
    pemcoh = ?emcoh/?numemsoil.

    epp = 1.
    while ?epp <= ?numemsoil
    then write('slopel.in','#n', ?concat(emsol1,?epp), '#s ,
      ?concat(emsol2,?epp),
      ?concat(emsol3,?epp),?concat(emsol4,?epp),
      ?concat(emsol5,?epp),?concat(emsol6,?epp),
      ?concat(emsol7,?epp)) and epp = ?epp+1.

    fp=1.
    while ?fp <= ?numfdsoil
    then concat(fdsoil,?fp) is get_text(?concat(fdval1,?fp))
      and concat(fdsoil2,?fp) is get_text(?concat(fdval2,?fp))
      and concat(fdsoil3,?fp) is get_text(?concat(fdval3,?fp))
      and concat(fdsoil4,?fp) is get_text(?concat(fdval4,?fp))
      and concat(fdsoil5,?fp) is get_text(?concat(fdval5,?fp))
      and concat(fdsoil6,?fp) is get_text(?concat(fdval6,?fp))
      and concat(fdsoil7,?fp) is get_text(?concat(fdval7,?fp))
      and fp = ?fp + 1.

    fpp = 1.
    while ?fpp <= ?numfdsoil
    then write('slopel.in','#n', ?concat(fdsoil1,?fpp), '#s ,
      ?concat(fdsoil2,?fpp), ?concat(fdsoil3,?fpp),
      ?concat(fdsoil4,?fpp), ?concat(fdsoil5,?fpp),
      ?concat(fdsoil6,?fpp), ?concat(fdsoil7,?fpp))

    (*========= Limits and Water ============*)

    use_font(?boldFont).
    text('#e #fbrown  Slope Stability Analysis #d #s',
      ?nmethod).
    use_font(  ).

    text('#x3#y2.5 ( R: Real Number )
    #x3#y3.5 #mWater Table#m (R)#mLimits#m (R) (Bed Rock)
    #x3#y4.5 (Choose 7 points) (Choose 7 points)
    #x4#y5.5 X Co  Y Co  X Co  Y Co').

    wt=1.
    while ?wt <= 7
    then  concat(iwtx,?wt) is ?zero_int and
concat(iwty, ?wt) is ?zero_int and
wt = ?wt + 1.

if ?wtx1 <> [ ] then wt_input( ).
topic wt_input.
  wtt = 1.
  while ?wtt <= 7
       then concat(iwtx, ?wtt) is ?concat(wtxx, ?wtt) and
          concat(iwty, ?wtt) is ?concat(wtyy, ?wtt) and
       wtt = ?wtt + 1.
end.( * * )

wtx1 is set_focus(edit_line (?iwtx1,,4,7,7)).
wty1 is edit_line (?iwty1,,12,7,7).
wtx2 is edit_line (?iwtx2,,4,8.5,7).
wty2 is edit_line (?iwty2,,12,8.5,7).
wtx3 is edit_line (?iwtx3,,4,10,7).
wty3 is edit_line (?iwty3,,12,10,7).
wtx4 is edit_line (?iwtx4,,4,11.5,7).
wty4 is edit_line (?iwty4,,12,11.5,7).
wtx5 is edit_line (?iwtx5,,4,13,7).
wty5 is edit_line (?iwty5,,12,13,7).
wtx6 is edit_line (?iwtx6,,4,14.5,7).
wty6 is edit_line (?iwty6,,12,14.5,7).
wtx7 is edit_line (?iwtx7,,4,16,7).
wty7 is edit_line (?iwty7,,12,16,7).

Lmt = 1.
  while ?Lmt <= 7
       then concat(iLmx, ?Lmt) is ?zero_int and
          concat(iLmy, ?Lmt) is ?zero_int and
       Lmt = ?Lmt + 1.

if ?Lmxx1 <> [ ] then Lm_input( ).
topic Lm_input.
  Lmm = 1.
  while ?Lmm <= 7
       then concat(iLmx, ?Lmm) is ?concat(Lmxx, ?Lmm) and
          concat(iLmy, ?Lmm) is ?concat(Lmyy, ?Lmm) and
       Lmm = ?Lmm + 1.
end.( * * )

Lmx1 is edit_line (?iLmx1,,27,7,7).
Lmy1 is edit_line (?iLmy1,,35,7,7).
Lmx2 is edit_line (?iLmx2,,27,8.5,7).
Lmy2 is edit_line (?iLmy2,,35,8.5,7).
Lmx3 is edit_line (?iLmx3,,27,10,7).
Lmy3 is edit_line (?iLmy3,,35,10,7).
Lmx4 is edit_line (?iLmx4,,27,11.5,7).
Lmy4 is edit_line (?iLmy4,,35,11.5,7).
Lmx5 is edit_line (?iLmx5,,27,13,7).
Lmy5 is edit_line (?iLmy5,,35,13,7).
Lmx6 is edit_line (?iLmx6,,27,14.5,7).
Lmy6 is edit_line (?iLmy6,,35,14.5,7).
Lmx7 is edit_line (?iLmx7,,27,16,7).
Lmy7 is edit_line (?ilmy7,,35,16,7).
if ?ssmeth is CIRCLE then scircle( ).
if ?ssmeth is CIRCL2 then scircl2( ).
if ?ssmeth is RANDOM then srandom( ).
if ?ssmeth is SURFAC or ?ssmeth is SURBIS then sspecific( ).
topic scircle.
text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,' #m ')
#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4 
#x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').
spxll is edit_line (?zero_int,,50,6,7).
spxrl is edit_line (40.1,,58,6,7).
spxl2 is edit_line (50.1,,66,6,7).
spxr2 is edit_line (179.9,,74,6,7).
me is edit_line (?zero_int,,50,9,7).
ls is edit_line (5.5,,58,9,7).
ccsi is edit_line (?zero_int,,66,9,7).
csi is edit_line (?zero_int,,74,9,7).
end.
topic scircl2.
text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,' #m ',
'#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4 
#x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').
spxll is edit_line (?zero_int,,50,6,7).
spxrl is edit_line (40.1,,58,6,7).
spxl2 is edit_line (50.1,,66,6,7).
spxr2 is edit_line (179.9,,74,6,7).
me is edit_line (?zero_int,,50,9,7).
ls is edit_line (4.5,,58,9,7).
ccsi is edit_line (?zero_int,,66,9,7).
csi is edit_line (?zero_int,,74,9,7).
end.
topic srandom.
text('#x50#y2.5 Failure Surface (R) #s #m',?svalue,' #m ',
'#x50#y5 X1 #x58#y5 X2 #x66#y5 X3 #x74#y5 X4 
#x50#y8 L1 #x58#y8 L2 #x66#y8 L3 #x74#y8 L4 ').
spxll is edit_line (10.1,,50,6,7).
spxrl is edit_line (40.1,,58,6,7).
spxl2 is edit_line (130.1,,66,6,7).
spxr2 is edit_line (179.9,,74,6,7).
me is edit_line (?zero_int,,50,9,7).
ls is edit_line (15.1,,58,9,7).
ccsi is edit_line (-15.1,,66,9,7).
csi is edit_line (-44.5,,74,9,7).
end.
topic specific.
text('Failure Surface (R) X Co Y Co X Co Y Co').
sx1 is edit_line (1.2,50,6,7).
sy1 is edit_line (5.1,58,6,7).
sx2 is edit_line (40.1,68,6,7).
sy2 is edit_line (8.5,76,6,7).
sx3 is edit_line (55.5,50,7.5,7).
sy3 is edit_line (8.1,58,7.5,7).
sx4 is edit_line (110.1,68,7.5,7).
sy4 is edit_line (7.2,76,7.5,7).
sx5 is edit_line (125.1,50,9,7).
sy5 is edit_line (17.1,58,9,7).
sx6 is edit_line (140.5,68,9,7).
sy6 is edit_line (20.1,76,9,7).
sx7 is edit_line (179.1,50,10.5,7).
sy7 is edit_line (7.5,58,10.5,7).
sx8 is edit_line (179.1,68,10.5,7).
sy8 is edit_line (7.5,76,10.5,7).
end.

if ?ssload is 'Reinforcement' then reinf( ).
if ?ssload is Tieback then tieback( ).

topic reinf.
small5 is button2(Continue,Continue,55,12,15,1.5).
wait( ).
close_window(?small5).
text('Number of Reinforcement (I): Number of Points (I):').
numreinf is set_focus(edit_line(1,80,14.3,4)).
numpont is edit_line(3,80,15.8,4).
rein is button2(Continue,Continue,75,17.5,10,1.5).
wait( ).
close_window(?rein).
nreinf is get_text(?numreinf).
npoint is get_text(?numpont).
if ?npoint >=5 then npoint = 4 and npnum( ).

topic npnum.
reinfwin is window(, 45,15,41,4,, [dialogwindow,visible] , ?main,, ).
Text('Maximum Allowable Number of Points are 4').
button2('Close Me',Continue,5,3.5,15,1).
wait( ).
close_window(?reinfwin).
show_window( ).
end.
text(′#x50#y17.5 X Co(R) Y Co(R) Force(Lb/ft) I-Fac
(R)′).
iwx = concat(40,,0).
iwy = concat(35,,0).
rp = 1.
rr = 1.
m = 18.5.
iforce1 = concat(0,,0).
iforce2 = concat(503,,0).
iforce3 = concat(503,,0).

while ?rr <= ?ntie
then concat(rvalbl,?rr) is edit_line (?iwx,,50,?m,6)
and set_focus(?rvalbl) and
concat(rvalb2,?rr) is edit_line (?iwy,,58,?m,6) and
concat(rvalb3,?rr) is edit_line(#concat(iforce1,
?rr),, 66,?m,10) and
concat(rvalb4,?rr) is edit_line (?zero_int,,78,?m,6)
and m =?m+2 and rr =?rr +1.
end.(*reinf*)

topic tieback.
small6 is button2(Continue,Continue,55,12,15,1.5).
wait( ).
close_window(?small6).
text(′#x47#y13 #mTieback Load#m (R)
#x47#y14.5 Number of Tieback Loads (I):′).
numtie is set_focus(edit_line(1,,77,14.3,4)).
tie is button2(Continue,Continue,68,16,10,1.5).
wait( ).
close_window(?tie).
ntie is get_text(?numtie).
if ?ntie >=6 then ntie = 6 and ntienum( ).

topic ntienum.
tiewin is window(, 45,15,41,4, [dialogwindow,visible] ,
?main, ,).
Text(′Maximum Allowable Number of Tieback Loads are 5′).
button2(′Close Me′,Continue,5,3.5,15,1).
wait( ).
CLOSEwンドow(?tiewin).
show_wındow( ).
end. (*ntienum*)

text(′#x47#y16 N(I) Y (R) Load(R) Sp(R) In(R) Lth(R)′).
tb=1.
m=17.
while ?tb <= ?ntie
then concat(tvalbl,?tb) is edit_line (1,,47,?m,4) and
set_focus(?tvalbl) and
concat(tvalb2,?tb) is edit_line (,,52,?m,6) and
concat(tvalb3,?tb) is edit_line (,,59,?m,8) and
concat(tvalb4,?tb) is edit_line (,,68,?m,6) and
concat(tvalb5,?tb) is edit_line (,,75,?m,6) and
concat(tvalb6,?tb) is edit_line (,,82,?m,6) and
m =?m+1.5 and
end.(*tieback*)

button2('Press Here To Continue =》',Continue,55,26,
30,1.5).
wait(  ) .

wtg=1.
while ?wtg <= 7
then concat(wtxx,?wtg) is get_text (?concat(wtx,?wtg))
and concat(wtyy,?wtg) is get_text (?concat(wty,?wtg))
and wtg = ?wtg +1.

write('slopet.in','#n', WATER,'#n', l,'#s ',62.4,'#n',7,
'n '#n ?wtxxl,'#s',?wtyyl,'#n','
,wtxx2,'#s',?wtyy2,'#n','
,wtxx3,'#s',?wtyy3,'#n','
,wtxx4,'#s',?wtyy4,'#n','
,wtxx5,'#s',?wtyy5,'#n','
,wtxx6,'#s',?wtyy6,'#n','
,wtxx7,'#s',?wtyy7).
write('fdesign.in','#n', 7,'#n',
,wtxxl,'#s ',Pwtyyl,'#n',
,wtxx2,'#s',?wtyy2,'#n',
,wtxx3,'#s',?wtyy3,'#n',
,wtxx4,'#s',?wtyy4,'#n',
,wtxx5,'#s',?wtyy5,'#n',
,wtxx6,'#s',?wtyy6,'#n',
,wtxx7,'#s',?wtyy7).

Lm=1.
while ?Lm <= 7
then concat(Lmxx,?Lm) is get_text (?concat(Lmx,?Lm))
and concat(Lmyy,?Lm) is get_text (?concat(Lmy,?Lm))
and Lm = ?Lm +1.

write('slopet.in','#n',LIMITS,'#n',6,'#s ',6,'#n',
,Lmxx1,'#s',?Lmyyl,?Lmxx2,?Lmyy2,'#n',
,Lmxx3,'#s',?Lmyy2,?Lmxx3,?Lmyy3,'#n',
,Lmxx3,'#s',?Lmyy3,?Lmxx4,?Lmyy4,'#n',
,Lmxx4,'#s',?Lmyy4,?Lmxx5,?Lmyy5,'#n',
,Lmxx5,'#s',?Lmyy5,?Lmxx6,?Lmyy6,'#n',
,Lmxx6,'#s',?Lmyy6,?Lmxx7,?Lmyy7).
write('fdesign.in','#n',7,'#n',
,Lmxx1,'#s ',?Lmyyl,'#n',
,Lmxx2,'#s ',?Lmyy2,'#n',
,Lmxx3,'#s ',?Lmyy3,'#n',
,Lmxx4,'#s ',?Lmyy4,'#n',
,Lmxx5,'#s ',?Lmyy5,'#n',
,Lmxx6,'#s ',?Lmyy6,'#n',
if ?ssload is Reinforcement then reinfwt( ) .
if ?ssload is Tieback then tiewt( ) .

topic reinfwt.
write('slopel.in', '#n', REINF, '#n', nreinf, '#n', npoint).
while ?rp <= ?npoint
then write('slopel.in', '#n',
    [get_text(?concat(rvalbl,?rp)),'#s ',
    get_text(?concat(rvalb2,?rp)),
    get_text(?concat(rvalb3,?rp)),
    get_text(?concat(rvalb4,?rp)) ] ) and
    rp =?rp +1.
end.
topic tiewt.
trp =1.
write('slopel.in', '#n', TIES, '#n', ntie).
while ?trp <= ?ntie
then write('slopel.in', '#n',
    [get_text(?concat(tvalbl,?trp)),'#s ',concat(0,0),
    get_text(?concat(tvalb2,?trp)),
    get_text(?concat(tvalb3,?trp)),
    get_text(?concat(tvalb4,?trp)),
    get_text(?concat(tvalb5,?trp)),
    get_text(?concat(tvalb6,?trp)) ] ) and
    trp =?trp +1.
end.
if Pssmeth is CIRCLE or Pssmeth is CIRCL2 or Pssmeth is
RANDOM then wcircle( ).
if Pssmeth is SURFAC or Pssmeth is SURBIS then
wspecific( ).
topic wcircle.
write('slopel.in', '#n', Pssmeth, '#n', 10, '#s ', 10,'#n',
    [get_text (pspxl1),'#s', get_text (pspxrl),
    get_text (pspxl2), get_text (pspxr2),'#n',
    get_text (pm), get_text (?ls), '#s',
    get_text (?ccs1), get_text (?csi) ], '#n' ).
end.
topic wspecific.
write('slopel.in', '#n', Pssmeth, '#n8',
    get_text (?sx1), '#s ', get_text (?syl), '#n',
    get_text (?sx2), '#s', get_text (?sy2), '#n',
    get_text (?sx3), '#s', get_text (?sy3), '#n',
    get_text (?sx4), '#s', get_text (?sy4), '#n',
    get_text (?sx5), '#s', get_text (?sy5), '#n',
    get_text (?sx6), '#s', get_text (?sy6), '#n',
    get_text (?sx7), '#s', get_text (?sy7), '#n',
    get_text (?sx8), '#s', get_text (?sy8), '#n', EXECUT).
end.(*wspecific*)

key_stable is 999.
write('error.out',1).
set_file_pos('slopel.in',0,beginning).
run('stable.exe').

if ?ssload is Reinforcement then wreinf() else woutren().
topic wreinf.
rrpp=1.
write('slope4.out',?npoint).
while ?rrpp <= ?npoint
then write('slope4.out','#n',
[get_text(?concat(rvalbl,?rrpp)),'#s ',
 get_text(?concat(rvalb2,?rrpp))]) and
 rrpp =?rrpp +1.
end.
topic woutren.
write('slope4.out',0).
end.

set_file_pos('slopel.out',0,beginning).
set_file_pos('slope2.out',0,beginning).
set_file_pos('slope3.out',0,beginning).
set_file_pos('slope4.out',0,beginning).

use_font (?boldfont).
text('#e #fbrown Failure Surfaces for FGD Embankment #d').
use_font( ).
run('vbslope.exe').
set_file_pos('slopel.out',-10,end).
FFOS is read line('slopel.eut').
set_file_pos('error.out',0,beginning).
error_read is read_line('error.out').
bitmap(?box19,15,7).
text('#x18#y9.5 o. Please select the following option.').

check_box('Continue to Settlement Analysis',settle,20,13).
if ?error_read <> 1 then text('#x22#y15 Error in the Program is #s#m',?error_read,'#m')
else text( ).
topic recal_stabl.
do(!main:fgd_all:fstable).
end.
topic mark(item).
irow is 10 and icol is 17 and ewid is 60 and eht is 10.
subtitle is concat('Error Code ',?item).
if ?item is 'Embankment Boundary' then subtitle is ?item.
and irow is 42 and icol is 4 and ewid is 45 and eht is 22.
if ?item is MUW or ?item is SUW or ?item is Coh or ?item
is Fric or ?item is PPP
or ?item is PPC or ?item is PSN then subtitle is
'Material Properties' and irow is 2 and icol is 5 and
ewid is 47 and eht is 23.
if ?item is Circular or ?item is Irregular
then irow is 2 and icol is 5 and ewid is 42 and eht is 20
and subtitle is 'Range of Failure Surface Generation'.
if ?item is 'Reinforcement' then subtitle is ?item and
irow is 10 and icol is 5 and ewid is 37 and eht is 20.
if ?item is 'Water Table' or ?item is Limits or ?item is
'Tieback Load' or ?item is Specific then subtitle is
?item and irow is 2.5 and icol is 19 and ewid is 45 and
eht is 11.5.
if ?item is 'Correction Factor' then subtitle is
concat(?item,' for Su') and irow is 2 and icol is 12
and ewid is 55 and eht is 15.5.

helpwin is window(,?irow,?icol,?ewid,?eht, ?subtitle,
[thinframe,showchildren,visible],?main, ,
lighgray,close_event).
button2('Close Me',close,1,1,1,1.3).
window (,1,2.3,?ewid,?eht-2.5,, [child,siblings,visible,
vertscroll,showchildren],?helpwin ).
set_file_pos('stable.txt',0,beginning).
stabl is read('stable.txt',concat('//',?item),'//').
text('#e',?stabl).
if ?item is PCSTABL6
then bitmap(load_bitmap('stable.bmp'),2,15).
if ?item is 'Embankment Boundary'
then bitmap(load_bitmap('shape2.bmp'),1,1).
if ?item is Reinforcement
then bitmap(load_bitmap('reinf.bmp'),1,1).
if ?item is 'Water Table'
then bitmap(load_bitmap('waterhlp.bmp'),2,2).
if ?item is Limits
then bitmap(load_bitmap('limithlp.bmp'),2,2).
if ?item is 'Tieback Load'
then bitmap(load_bitmap('tiehlp.bmp'),2,2).
if ?item is Specific
then bitmap(load_bitmap('spechlp.bmp'),2,2).
if ?item is Circular or ?item is Irregular
then bitmap(load_bitmap('circle.bmp'),2,2)
and recir is button2('Press Here For
More',cir_more,7,16.5,25,1.5).
if ?item is 'Correction Factor'
then bitmap(load_bitmap('sufactor.bmp'),1,1).
end.
topic cir_more.
close_window(?recir).
window(,2,6.2,38,22.5, 'Generation of Trial Failure
Surface', [thinframe,showchildren,visible],?main,
lightgray, close_event).
bitmap(load_bitmap('circle2.bmp'),2,2).
button2('Close Me', close5,1,1,11,1.3).

topic close5.
  close_window( )
end.
end.

topic close.
  close_window( )
  close_window( )
  show_window( )
end.
end. (*numeric*)
Settlement Calculation
settle.kb

(topic settle.
new_file('settle.in').
new_file('settle1.out').
new_file('settle2.out').
new_file('pcset.in')

use_font(?boldfont).
text('#e#x3#y3 Settlement Analysis #d')
use_font( ).
text('#x3#y3 What type of drainage conditions do you have in the field ?').

bitmap(load_bitmap('boxl3.bmp'),4,5).
bitmap(load_bitmap('drain.bmp'),6,7).

drainList is [ 
    'Type I - A',
    'Type I - B',
    'Type I - C',
    'Type I - D',
    'Type II - A',
    'Type III - A'].

drainList is [ 
    'Type I - A',
    'Type I - B',
    'Type I - C',
    'Type I - D',
    'Type II - A',
    'Type III - A'].

drainList is [ 
    'Type I - A',
    'Type I - B',
    'Type I - C',
    'Type I - D',
    'Type II - A',
    'Type III - A'].

drainList is [ 
    'Type I - A',
    'Type I - B',
    'Type I - C',
    'Type I - D',
    'Type II - A',
    'Type III - A'].

drainList is [ 
    'Type I - A',
    'Type I - B',
    'Type I - C',
    'Type I - D',
    'Type II - A',
    'Type III - A'].

if ?drain is 'Type I - A' or ?drain is 'Type I - B' or 
    ?drain is 'Type I - C' then type = 1 and path = 1.
if ?drain is 'Type I - D' then type = 1 and path = 2.
if ?drain is 'Type II - A' then type = 2 and path = 2.
if ?drain is 'Type III - A' then type = 3 and path = 2.

ise is 1.
isf is 1.
if ?key_settle > 0 then ise is ?numse and isf is ?numsf.
use_font(?boldfont).
text('#e#x3#y3 Settlement Analysis #d')
use_font( ).
text('#e#x3#y3 Define number of layers
    #x3#y6 Number of Layers in Embankment
    #x3#y7.5 Number of Layers in Foundation
    #x47#y12 Number of Layers in Fill : #fgreen 2 #d
    #x47#y14 Number of Layers in Foundation : #fgreen 3 #d
    #x47#y14.5 o. Note:
    #x47 Split many layers of foundation
to get accurate settlement
calculation even if the foundation
includes one uniform soil property '.
numsei is set_focus(edit_line(?ise,,36,5.8,4)).
umsfi is edit_line(?isf,,36,7.3,4).
bitmap(load_bitmap('boxl3.bmp'),45,4).
bitmap(load_bitmap('laynum.bmp'),47,5).

button2('Press Here To Continue =>>',continue,55,26,
30,1.4).
wait( ).
numse is get_text(?numsei).
umsfi is get_text(?numsfi).

if ?numse >=4 or ?numsfi >=4 then sethlp( ).
if ?numse >=4 then numse = 3.
if ?numsfi >=4 then numsfi = 3.

topic sethlp.

setwin is window(, 40,12,50,8,, [dialogwindow,visible] ,
?main, ,,).
text(' @ Maximum Allowable Number of Layers in Embankment are 3.
@ Maximum Allowable Number of Layers in Foundation are 3').
button2('Close Me',Continue,35,6.5,10,1.5).
wait( ).
close_window(?setwin).
show_window( ).
end.

use_font(?boldfont).
text('#e#x3#y1.5 #fbrown Settlement Analysis #d').
use_font( ).
text('#x3#y4 @ Determine OCR value at the middle
of each depth.').
text(' #x50#y5.5 Average Stress History Profile
#x55#y22 OCR = Svm / Svo').

ooint =1.
while ?ooint <=3
then concat(ocrint,?ooint) is concat(1,,0) and
ooint=?ooint+1.

if ?key_ssettle > 0 then ocr_init( ).
topic ocr_init.
if ?numse >= ?senum then eminnum is ?senum and emaxnum is
?numse.
if ?numse < ?senum then eminnum is ?numse and emaxnum is
?senum.
if ?numsfi >= ?sfnum then fminnum is ?sfnum and fmaxnum is
?numsfi.
if ?numsf < ?sfnum then fminnum is ?numsf and fmaxnum is ?sfnum.

ocrit=1.
while ?ocrit <= ?fminnum
then concat(ocrichto, ?ocrit) is ?concat(ocr, ?ocrit) and
ocrit = ?ocrit + 1.
end.

ocrmm = 10.
ocrm=1.
while ?ocrm <= ?numsf
then concat(ocred, ?ocrm) is edit_line (?concat(ocrint, ?ocrm), ?,?ocrmm,7,4,7) and
set_focus(ocred1) and
ocrm = ?ocrm + 1 and
ocrmm = ?ocrmm+7.

bitmap(load_bitmap('boxl3.bmp'),45,4).
bitmap (load_bitmap('ocr.bmp'),47,8).
button2('Press Here To Continue =>>',continue,55,26,30,1.4).
wait( ).

ocrmg=1.
while ?ocrmg <= ?numsf
then concat(ocr, ?ocrmg) is
get_text(ocr, ?ocrmg) and
ocrmg = ?ocrmg + 1.

use_font() .
text('#e #fbrown Settlement Calculation #d, #mSETFGD#m').
text('#x2#y2.5 @ Shape of Embankment #x3#y4 @ Slope Ratio(SR)#x3#y5.5 Height of Embankment (ft) #x3#y7 Depth of Foundation (ft)(DF)#x3#y8.5 Water Table (ft) (WT) #x3#y10 Top half width (ft)(TW) ').
text('#x2#y13 @ Material Properties of Embankment. #x2#y14.5 mDepth of each layer#m (EDEL) #x2#y16 #mTotal Unit Weight(pcf)#m').
text('#x45#y2.5 @ Material Properties of Foundation Soil.').
text('#x45#y4 #mDepth of each layer#m (FDEL) #x45#y5.5 #mTotal Unit Weight(pcf)#m #x45#y7 #mSaturated Unit Weight(pcf)#m #x45#y8.5 #mCompression Index#m #x45#y10 #mRecompression Index#m #x45#y11.5 #mVoid Ratio#m').
text('#x45#y13 #mConsolidation Coef.(ft2/day)#m #x50#y15 #m2nd Compression Index#m #x45#y18 @ #mConstruction Period (days)#m #x45#y19.5 @ #mConsolidation Degree (%)#m').
text('#x45#y22 The default values were obtained from #x45 Data Base or from Previous Input').

bitmap(load_bitmap('setshp.bmp'),5,18).

rat is ?ratio.
  higi is ?height.
  depi is ?depth.
  wat is concat(10,,0).
  widi is ?width.
  drpi is concat(20,,0).

if ?key_ssettle > 0 then initial_shape( )
  topic initial_shape.
    rat is ?ratio.
    higi is ?height.
    depi is ?depth.
    wat is ?water.
    widi is ?width.
    drpi is ?draind.
  (*
   drgdep is edit_line(?drpi,,34,11.5,7).*)
end.

sem =1.
while ?sem <=3
  then concat(seintl,?sem) is concat(10,,0) and
    concat(seint2,?sem) is ?emsolll and
  sem=sem+1.

s=1.
while ?s <= 3
  then concat(sfint1,?s) is concat(10,,0) and
    concat(sfint2,?s) is ?fndsolll and
    concat(sfint3,?s) is ?fndsoll2 and
    concat(sfint4,?s) is ?fsc and
    concat(sfint5,?s) is ?fscrl and
    concat(sfint6,?s) is ?fsv and
    concat(sfint7,?s) is ?fscv and
  s =?s+1.

if ?key_ssettle > 0 then initial_sf2( ).
  topic Initial_sf2.
  se=1.
  while ?se <= ?eminnum
    then concat(seintl,?se) is ?concsegval1,?se and
      concat(seint2,?se) is ?emsolll and
    se =?se +1.

sf=1.
while ?sf <= ?fminnum
  then concat(sfint1,?sf) is ?concsegfval1,?sf and
concat(sfint2,?sf) is ?concat(sfgval2,?sf) and
concat(sfint3,?sf) is ?concat(sfgval3,?sf) and
concat(sfint4,?sf) is ?concat(sfgval4,?sf) and
concat(sfint5,?sf) is ?concat(sfgval5,?sf) and
concat(sfint6,?sf) is ?concat(sfgval6,?sf) and
concat(sfint7,?sf) is ?concat(sfgval7,?sf) and
sf =?sf+1.
end.(**

senum is ?numse.
sfnum is ?numsf.
semm = 28.
set=1.
while ?set <= ?numse
then concat(seval1,?set) is edit_line (?concat(seint1,
   ?set),?,semm,14.5,7) and
   concat(seval2,?set) is edit_line (?concat(seint2,
   ?set),?,semm,16,7) and

sfm = 69.
sft=1.
while ?sft <= ?numsf
then concat(sfval1,?sft) is edit_line
   (?concat(sfint1,?sft),?,sfm,4,7) and
concat(sfval2,?sft) is edit_line
   (?concat(sfint2,?sft),?,sfm,5.5,7) and
concat(sfval3,?sft) is edit_line
   (?concat(sfint3,?sft),?,sfm,7,7) and
concat(sfval4,?sft) is edit_line
   (?concat(sfint4,?sft),?,sfm,8.5,7) and
concat(sfval5,?sft) is edit_line
   (?concat(sfint5,?sft),?,sfm,10,7) and
concat(sfval6,?sft) is edit_line
   (?concat(sfint6,?sft),?,sfm,11.5,7) and
concat(sfval7,?sft) is edit_line
   (?concat(sfint7,?sft),?,sfm,13,7) and
sfm =?sfm+7 and
sft =?sft +1.

sendc is edit_line(0.02,,75,15,7).
timei is edit_line (200,,71,18,7).
condeg is edit_line(90,,71,19.5,7).

button2('Press Here To Continue =>>',continue,55,26,
30,1.5).
wait( ).

ratio is get_text(?rat).
height is get_text(?hig).
depth is get_text(?dep).
water is get_text(?wat).
width is get_text(?topwid).
emhigh = 0.

setq = 1.
while ?setq <= ?numse
then concat(segval1, ?setq) is get_text
    (?concat(sevall, ?setq)) and
    concat(segval2, ?setq) is get_text
    (?concat(seval2, ?setq)) and
    emhigh = ?concat(segval1, ?setq) + ?emhigh and
setq = ?setq + 1.

fdhigh = 0.
sftq = 1.
while ?sftq <= ?numsf
then concat(sfgval1, ?sftq) is get_text
    (?concat(sfval1, ?sftq)) and
    concat(sfgval2, ?sftq) is get_text
    (?concat(sfval2, ?sftq)) and
    concat(sfgval3, ?sftq) is get_text
    (?concat(sfval3, ?sftq)) and
    concat(sfgval4, ?sftq) is get_text
    (?concat(sfval4, ?sftq)) and
    concat(sfgval5, ?sftq) is get_text
    (?concat(sfval5, ?sftq)) and
    concat(sfgval6, ?sftq) is get_text
    (?concat(sfval6, ?sftq)) and
    concat(sfgval7, ?sftq) is get_text
    (?concat(sfval7, ?sftq)) and
    fdhigh = ?concat(sfgval1, ?sftq) + ?fdhigh and
sftq = ?sftq + 1.

sendcom is get_text(?sendc).

cond is get_text(?condeg).
time is get_text(?timei).
if ?height <> ?emhigh then emhigh_hlp( ).
if ?depth <> ?fdhigh then fdhigh_hlp( ).
if ?water < 0 or ?water >= ?fdhigh then water_hlp( ).

topic emmhigh_hlp.
emhwin is window(, 40,12,50,9,, [dialogwindow,visible],
?main, ,,).
text( '
  @ Total height of Embankment should be equal to sum of
  each layer in Embankment. HE = Sum of EDEL' ).

semh = 5.
seh = 1.
while ?seh <= ?numse
then concat(seval1, ?seh) is edit_line (concat(0,,0)
    ,, ?semh,6.5,7) and
set_focus(?seval11) and
seh = ?seh + 1 and
semh=semh+7.
button2('Close Me',Continue,35,6.5,10,1.5).
   wait( ).
seth=1.
   while ?seth <= ?numse
       then concat(segval1,?seth) is get_text (?concat
           (segval1, ?seth)) and
       seth= ?seth+1.
   close_window(?emhwin).
show_window( ).
end.

topic fdhigh_hlp.
   fdhwin is window(, 40,12,50,9,[dialogwindow,visible],
      main, ,,).
   text(' @ Total Depth of Foundation should be equal
to sum of each layer in Foundation.
DF = Sum of FDEL' ).
sfh =5.
sfth=1.
   while ?sfth <= ?numsf
       then concat(sfvall,?sfth) is edit_line (concat(0,,0)
          ,,sfth,6.5,7) and
       set_focus(sfvall1) and
       sfth = ?sfth +1 and
       sfhm =sfhm+7.
button2('Close Me',Continue,35,6.5,10,1.5).
   wait( ).
sftgh=1.
   while ?sftgh <= ?numsf
       then concat(sfgvall,?sftgh) is get_text (?concat
          (sfvall,?sftgh)) and
       sftgh = ?sftgh +1.
   close_window(?fdhwin).
show_window( ).
end.

topic water_hlp.
   wtw = window(, 40,12,50,9, [dialogwindow,visible]
      ,main, ,,).
   text('@ Depth of Water Table is assumed within
the depth of foundation.
0 <= water table <= depth of foundation ' ).
   wat is set_focus(edit_line(0,,7,6.5,7)).
button2('Close Me',Continue,35,6.5,10,1.5).
   wait( ).
   water is get_text (?wat).
   close_window(?wtwin).
   show_window( ).
end.
(*============ write in files============*)

write('settle.in', ?ratio,'#s ',?height,?depth,
?water,?width, ?sendcom,'#n',?numse).
segm=1.
while ?segm <= ?numse
then write('settle.in','#n',?concat(segval1,?segm),'#s',
?concat(segval2,?segm)) and
segm=segm+1.
write('settle.in','#n',?type,'#s ',?path,?cond,?time,
'#n',?numse).

sftgw=1.
while ?sftgw <= ?numsf
then write('settle.in','#n',?concat(ocr,?sftgw),'#s ',
?concat(sfgval1,?sftgw), ?concat(sfgval2,?sftgw),
?concat(sfgval3,?sftgw), ?concat(sfgval4,?sftgw),
?concat(sfgval5,?sftgw), ?concat(sfgval6,?sftgw),
?concat(sfgval7,?sftgw)) and
sftgw =?sftgw +1.

(*====== Write for Report ===========*)

write('pcset.in','#n Input Data for SETFDG',
'#n -----------------------------',
'#n Slope Ratio: #s',?ratio,
'#n Height of Embankment(ft): #s',?height,
'#n Thickness of Foundation(ft): #s',?depth,
'#n Water Table from the bottom of fill(ft): #s',
?water,
'#n Width of Top half(ft): #s',?width,
'#n Number of Layers of Embankment: #s', ?numse,
'#n Thickness Each Layer Total Unit Weight #n').
segm=1.
while ?segm <= ?numse
then write('pcset.in','#s ',?concat(segval1,?segm),
' #s ',?concat(segval2,?segm),'#n') and
segm=segm+1.

write('pcset.in','#s Drainage Type: #s',?type,
'#n Consolidation Degree(%); #s',?cond,
'#n Construction Period(days): #s',?time,
'#n Number of Layers of Foundation: #s',?numsf,
'#n OCR TL TUW SUW Cc Cr Void Cv #n').
sftgw=1.
while ?sftgw <= ?numsf
then write('pcset.in','#s ',?concat(ocr,?sftgw),'#s',
?concat(sfgval1,?sftgw), ?concat(sfgval2,?sftgw),
?concat(sfgval3,?sftgw), ?concat(sfgval4,?sftgw),
?concat(sfgval5,?sftgw), ?concat(sfgval6,?sftgw),
?concat(sfgval7,?sftgw),'#n') and
sftgw =?sftgw +1.
write('pcset.in','#n -----------------------------').
key_settle is 999.
set_file_pos('settle.in',0,beginning).
run('settle.exe').

set_file_pos('settle2.out',0,beginning).
[STOT,TP,conset,contime,consol,remset] is_c
read('settle2.out','//SUNG','//HWAN').
(*Actually STOT is remaining settlement after
construction period*)

if ?seratio is 0 then seratio =4.
write('fdesign.in','#n',?seratio,?ratio,?FFOS,?OFOS).
use_font (?boldfont).
text('#e #x3#y2.5 #fbrown The Relationships of Elapsed
Time and Settlement #d').
use_font ( ).
set_file_pos('settle1.out',0,beginning).
set_file_pos('settle2.out',0,beginning).

run ('setplt.exe').

bitmap(?box19,6,6).
text('#x8#y7.5 o. Please select the following option.
check_box('Recalculate Settlement',return_settle,10,10).
check_box('See Conclusions',fgdcon,10,12).

topic return_settle.
  do(! main:fgd_all:settle).
end.
topic mark(item).
  helpwin is window( , 40,8,50,18, ?item,
    [thinframe,showchildren,visible],?main, ,lightgray,
    close_event).
  button2('Close Me',close,1,1,11,1.3).
  window (,1,2.3,50,15.5, [child,siblings,visible,
    vertscroll, showchildren],?helpwin ).
  set_file_pos('settle.txt',0,beginning).
  settle is read('settle.txt',concat('//',?item),'//').
  text('#e',?settle).
end.

topic close.
  close_window( ).
  close_window( ).
  show_window( ).
end.

end.(*settle*)
Conclusions with FGD

fgdcon.kb

topic fgdcon.
new_file('summary.out').
material_evaluation is normal.
if ?material_evaluation is well then CFOS is 1.2.
if ?material_evaluation is normal then CFOS is 1.4.

con1 is 'o. No more modification'.
con2 is 'o. Steepen slope ratio'.
con3 is 'o. Increase stability'.
con4 is 'o. Reduce settlement'.
con6 is 'o. Need foundation treatment'.
con7 is 'o. Increase Stability and Reduce Settlement'.
sugg1 is ' -.Use higher strength FGD materials'.
sugg2 is ' -.Use lighter weight FGD materials'.
sugg3 is ' -.Reduce water table'.
sugg4 is ' -.Use lower permeable FGD materials'.

FTUW= ?pemtot.
FUCU = ?pemcoh.
SSRE = ?ratio.
DF= ?depth.

use_font( ?boldfont).
text('#e #x7#y2.5 #fbrown Summary of the Numerical Analysis #d').
use_font ( ).
bitmap(?box6,7,5).
if 70FOS is 0 then OFOS is 'No Value'.
if 70seratio is 0 then seratio is 'No Value'.
if 70sestot is 0 then sestot is 'No Value'.
text('#x40#y6 Soil Fill FGD Fill',
' #x10#y8 o. Factor of Safety #s ', ?OFOS,
' #s#x63', ?FFOS,' #x10#y9.5 o. Maximum Slope Ratio
#s', ?seratio,' #s#x63',?ratio,
' #x10#y11 o. Total Settlement (ft) : #s',?sestot,' #s
#x63',?stot,
' #x10#y12.5 o. Time required for 90%
consolidation(days): #s',?tp,
' #x10#y14 o. % Consolidation at End-Construction :
#s', ?consol, ' #x10#y15.5 o. Remaining Settlement after
End-Construction (ft): #s',?remset).

write('summary.out','#n Summary of the Numerical Analysis',
' #n ---------------------------------------------------------------',
' #n Soil Fill FGD Fill',
' #n o. Factor of Safety: #s ', ?OFOS, ' #s ', ?FFOS,
' #n o. Maximum Slope Ratio: #s ', ?seratio, ' #s ', ?ratio,
' #n o. Total Settlement (ft): #s ?sestot, ' #s ', ?stot,
# n o. Time required for 90%
'Time consolidation(days): - #s', ?tp,
# n o. Consolidation Degree',
# n at End-Construction(%): - #s', ?consol,
# n o. Remaining Settlement',
# n after End-Construction (ft): - #s', ?remset,
# n --------------------------------------

button2('Press Here To Continue =>>', continue, 55, 26, 30, 1.5).
wait( ).
use_font(?boldfont).
text('e#x5#y1.5 #fbrown Conclusion #d').
use_font( ).
bitmap(load_bitmap('box6.bmp'), 5, 1).
text('x7#9 Factor of Safety : #s', ?FFOS,
'x7#10 Total Settlement (ft) #s', ?STOT,
'x7#11 Total Unit Weight of Fill Soil (pcf): #s', ?FTUW,
'x7#12 Cohesion of Fill Soil (psf): #s', ?FUCU,
'x7#13 Side Slope Ratio: #s', ?ratio).

fill_type1 is "Full Thickness of FGD Waste".
fill_type2 is "Mixed Layers of Soil and FGD Waste".

if ?FFOS >= ?CFOS
and ?STOT <= 1
and ?FTUW < 100
and ?FUCU >= 10000
and ?SSRE <= 2
then text('x7#4', ?con1)
and write('fdesign.in', '#n', ?fill_type1)
and bitmap(load_bitmap('dntype1.bmp'), 45, 4).

if ?FFOS >= ?CFOS
and ?STOT <= 1
and ?FTUW >= 100
and ?SSRE > 2
then text('x7#4', ?con2, 'x7#6', ?sugg2)
and write('fdesign.in', '#n', ?fill_type1)
and bitmap(load_bitmap('dntype3.bmp'), 45, 4).

if ?FFOS >= ?CFOS
and ?STOT <= 1
and ?FUCU < 10000
and ?SSRE > 2
then text('x7#4', ?con2, 'x7#6', ?sugg1)
and write('fdesign.in', '#n', ?fill_type1)
and bitmap(load_bitmap('dntype3.bmp'), 45, 4).

if ?FFOS >= ?CFOS
and ?STOT <= 1
and ?FUCU < 10000
and ?SSRE =< 2
then text('#x7#y4',?con1)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype1.bmp'),45,4).

if ?FFOS =? CFOS
and ?STOT < = 1
and ?FTUW = = 100
and ?SSRE =< 2
then text('#x7#y4',?con1)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype1.bmp'),45,4).

if ?FFOS =? CFOS
and ?STOT > 1
and ?FTUW < 100
then text('#x7#y4',?con6)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype6.bmp'),45,4).

if ?FFOS < ?CFOS
and ?STOT < = 1
and ?FTUW < 100
and ?FUCU <10000
then text('#x7#y4',?con3,'#x7#y6',?sugg1)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype5.bmp'),45,4).

if ?FFOS < ?CFOS
and ?STOT < = 1
and ?FTUW < 100
and ?FUCU =>=10000
then text('#x7#y4',?con6)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype6.bmp'),45,4).

if ?FFOS < ?CFOS
and ?STOT < = 1
and ?FTUW > = 100
and ?FUCU <10000
then text('#x7#y4',?con3,'#x8',?sugg1,'#x8',?sugg2)
and write('fdesign.in','#n',?fill_type1)
and bitmap(load_bitmap('dntype10.bmp'),45,4).

if ?FFOS < ?CFOS
and ?STOT < = 1
and \( \text{FTUW} \geq 100 \)
and \( \text{FUCU} \geq 10000 \)
then text('#x7#y4',\text{con3}, '#x8',\text{sugg2})
and write('fdesign.in','#n',\text{fill_type1})
and bitmap(load_bitmap('dntype8.bmp'),45,4).

if \( \text{FFOS} < \text{CFOS} \)
and \( \text{STOT} > 1 \)
and \( \text{FTUW} < 100 \)
and \( \text{FUCU} < 10000 \)
then text('#x7#y4',\text{con6})
and write('fdesign.in','#n',\text{fill_type1})
and bitmap(load_bitmap('dntype6.bmp'),45,4).

if \( \text{FFOS} < \text{CFOS} \)
and \( \text{STOT} > 1 \)
and \( \text{FTUW} < 100 \)
and \( \text{FUCU} < 10000 \)
then text('#x7#y4',\text{con6})
and write('fdesign.in','#n',\text{fill_type1})
and bitmap(load_bitmap('dntype6.bmp'),45,4).

if \( \text{FFOS} < \text{CFOS} \)
and \( \text{STOT} > 1 \)
and \( \text{FTUW} \geq 100 \)
and \( \text{FUCU} \geq 10000 \)
then text('#x7#y4',\text{con7}, '#x8',\text{sugg1}, '#x8',\text{sugg2})
and write('fdesign.in','#n',\text{fill_type1})
and bitmap(load_bitmap('dntype9.bmp'),45,4).

if \( \text{FFOS} < \text{CFOS} \)
and \( \text{STOT} > 1 \)
and \( \text{FTUW} \geq 100 \)
and \( \text{FUCU} \geq 10000 \)
then text('#x7#y4',\text{con7}, '#x8',\text{sugg2})
and write('fdesign.in','#n',\text{fill_type1})
and bitmap(load_bitmap('dntype9.bmp'),45,4).

bitmap(?box19,6,16).
text('#x8#y18 o. Please select the following option.').
check_box('Consider Reconsultation',return_fgd,9,20).
check_box('See Final Design',fdesign,9,22).

topic return_fgd.
    do(!main:fgd_all:fgd_db).
end.
end.(*fgd_con*)
Design of Embankment Using FGD
fdesign.kb

topic fdesign.
text('#e').
bitmap(?boxl0,3,3).
bitmap(?boxl0,48,3).
bitmap(?boxl0,3,15).
bitmap(?boxl0,48,15).

use_font(?boldfont).
text('#x5#y1 #fbrown Design of Fills #d').

use_font( ).
text('#x3#y4 o. By Proposed Highway Grade Line
 #x4#y5.5 - Height of Embankment(ft): #s',?height,
 #x4#y7 - Top Width of Embankment(ft): #s',?width,
 #x4#y8.5 - Right-Of-Way Available(ft): #s', ?row,
 #x50#y4 o. By Stability Analysis
 #x51#y5.5 - Height of Embankment(ft): #s',?height,
 #x51#y7 - Foreslope Ratio: #s',?ratio,
 #x51#y8.5 - Factor of Safety: #s',?ffos,
 #x5#y16 o. By Standard Guidline (after AASHTO)
 #x50#y16 o. Input Parameters for Slope Design
 #x50#y17 (choose them from above three results)
 #x51#y19 @ Height of Embankment(ft):
 #x51#y20.5 @ Side Slope:').
dnl is set_focus(edit_line(?height,,80,18.8,7)).
dn2 is edit_line(?ratio,,80,20.3,7).
stslope is read('slope.txt').
text('#x1#y17',?stslope).

button2('Press Here To Continue=>>',continue,55,26,
30,1.5).
wait( )
dheight is get_text(?dnl).
dslope is get_text(?dn2).

use_font(?boldfont).
text('#e #x2#y1 #fbrown Slope Design #d').
set_file_pos('fdesign.in',0,beginning).
run('fdesign.exe').

use_font( ).
window(, 15,9,48,7,,[visible,dialogwindow],?main,
',close_event).
text('#x5#y2.5 Press the following button to continue').

button2('Press Here To Continue =>>',continue,8,4.5,
30,1.5).
wait( ).
close_window( ).
show_window( ).
use_font(?boldfont).
text('#e #x2#y1 #fbrown Side Slope Cover Design #d').

bitmap(load_bitmap('side.bmp'),2,4).

use_font( ) .

bitmap(?box11,50,5).

text('#x52#y6 Choose one of the Side Cover Types').

sidetp is radio_button([[ 'Type I',55,7.5,t],[ 'Type II',55,9]]).

button2('Press Here To Continue =>>',continue,55,26,30,1.5).

wait( ) .

sdloc is side(?sidetp).

topic side(handle).

sidep is element(?handle, where(get_radio_button(?sidetp),t)).

side is get_text(?sidep).

end.

text('#e').

use_font(?boldfont).

bitmap(?box12,2,2).

bitmap(?box12,2,15).

bitmap(?box13,52,3).

text('#x2#y2 Side Slope Design
#x2#y3 Side Cover Design
#x54#y5 Design Information').

n=?dslope.

a=?width.

b=?dheight.

c=?depth.

d=?water.

use_font( ).

text('#x5#y4 (unit : ft)',

' #x30#y2.6 #fgreen2 #s =',?a,' #x20#y6.3 #s =',?n,

' #x42#y6.3 #s =',?b,' #x42.3#y9 #s =',?c,' #x30.5#y7.7

#s =',?d,' #d ').

emshape is load_bitmap('emshape.bmp').

bitmap(?emshape,6,2.5).

if ?side is 'Type I' then side is sidetp1.

if ?side is 'Type II' then side is sidetp2.

sidefil is concat(?side,'.bmp').

sidefig is load_bitmap(?sidefil).

bitmap(?sidefig,3,15.5).

text('#x54#y5 Height (ft)',

' #x54#y6.5 Slope',

' #x54#y8 Safety Factor',

' #x54#y9.5 Total Settlement (ft)',

' #x54#y11 Total Volume of Embankment (ft3)').
button2('Press Here To Continue =>>', continue, 55, 26, 30, 1.5).
wait().

use_font(?boldfont).
text('e ').

bitmap(?box6, 2, 1.2).
text('#f brown Construction Procedures #d').
use_font().
text('#y3.7 o. Range of Slope : greater than
'.'s', ?dslope,
'#y4.9 o. #m Optimum Moisture Content#m :
'.'s', ?aomc, '('%, ')
'#y6.1 o. Dry unit Weight : 's', ?aden, '(pcf)',
'#y7.3 o. Compaction Required in the field : 100 '
'.'s', ?den,
'#y8.5 o. Compaction Thickness of each layer: 6 to 8
'.' inches',
'#y9.7 o. #m Compaction Equipment#m',
'#y10.9 o. #m Erosion Control#m',
'#y12.1 o. #m Compaction During Wet Weather#m',
'#y13.3 o. #m Construction During Freezing
Weather#m').
conpro is load_bitmap('const.bmp').
bitmap(?box7, 2, 16.3).
bitmap(?conpro, 5, 17).

button2(Report, report, 76, 23, 5, 15, 1.5).
button2(Restart, restart, 76, 25, 15, 1.5).
button2(End, end, 76, 26.5, 15, 1.5).

topic mark(item).
markwin is window(, 40, 5, 50, 24, 'Design - Construction
Help', [thinframe, showchildren, visible], ?main, ,
lightgray, close_event).
button2('Close Me', close, 1, 1, 11, 1.3).
window(, 1, 2, 3, 50, 21.5, [child, siblings, visible,
showchildren], ?markwin ).
set_file_pos('design.txt', 0, beginning).
design is read('design.txt', concat('//', ?item), '//').
text('e', ?design).

topic close.
close_window( ).
close_window( ).
show_window(?main).
end.
end.(*mark*)
end.(*fdesign*)
topic report.
repwin is window(, 10, 4, 60, 24, 'Report Format of the
Consultation', [thinframe, showchildren, visible], ?main).
button2(Material, material_out, 1, 1, 12, 1.3).
button2('Stability',stable_out,13,1,12,1.3).
button2('Settlement',settle_out,25,1,12,1.3).
button2('Summary',sum_cont,37,1,12,1.3).
button2('Close Me',close,49,1,12,1.3).

window (.3,2.3,58,21.5,, [child,siblings,horzscroll,
vertscroll,visible,showchildren],?repwin).
use_font(?boldfont).
text('#e #x5#y5 #fred List and Print the Consultation
Results.#d#x5#y7 Please select a button shown above.').
use_font(  ).
topic material_out.
text('#e #y3 Properties of FGD By-product Used'),
prot is radio_button([['Leachate Property',8,5,t],
['Chemical Property',8,6.5], ['Engineering
Property',8,8]]).
button2('Press Here Again =>>',continue,5,18,30,1.5,
mov_up_event).
wait( ).
mtyp is mtype(?m_prot).
topic mtype(handle).
mtypel is element(?handle,
where(get_radio_button(?m_prot),t)).
fgd_prot is get_text(?mtypel).
end.

if ?fgd_prot is 'Leachate Property'
then lea_result( ).

if ?fgd_prot is 'Chemical Property'
then chem_result( ).

if ?fgd_prot is 'Engineering Property'
then eng_result( ).
end.(*material_out*)

(*===== stability ===================*)
topic stableout.
text('#e #y3 Output Files of Stability Calculation').
sta_prot is radio_button([['Input File',8,5,t],
['Summary of Stability Calculation',8,6.5],
['Output File',8,8], ['Graphic of Failure
Surface',8,9.5]]).
button2('Press Here Again =>>',continue,5,18,30,1.5,
mov_up_event).
wait( ).
staout is stablout(?sta_prot).
topic stablout(handle).
staout is element(?handle,
where(get_radio_button(?sta_prot),t)).
sta_out is get_text(?staout).
end.

if ?sta_out is 'Input File'
then sta_input( ).

if ?sta_out is 'Summary of Stability Calculation'
then sta_sum( ).

if ?sta_out is 'Output File'
then sta_output( ).

if ?sta_out is 'Graphic of Failure Surface'
then sta_graph( ).
end.(*stable_out*)

(*========settlement====================*)
topic settle_out.
text('#e #y3 Output Files of Settlement Calculation').
sett_prot is radio_button([[Input File',8,5,t],
               ['Summary of Settlement Calculation',8,6.5],
               ['Output File',8,8],
               ['Graphic of Time - Settlement',8,9.5]]).
button2('Press Here Again =>>,continue,5,18,30,1.5,
        move_up_event).
wait( ).

setout is settout(?sett_prot).
topic settout(handle).
settleout is element(?handle,
               where(get_radio_button(?sett_prot),t)).
settle_out is get_text(?settleout).
end.

if ?settle_out is 'Input File'
then sett_input( ).

if ?settle_out is 'Summary of Settlement Calculation'
then sett_sum( ).

if ?settle_out is 'Output File'
then sett_output( ).

if ?settle_out is 'Graphic of Time - Settlement'
then sett_graph( ).
end.(*settle_out*)

topic lea_result.
text('#e #y3 Leachate Property').
print('#x3#y2 Leachate Property').
end.

topic chem_result.
text('#e Chemical Property').
print('#x3#y2 Chemical Property').
end.
topic eng_result.
  set_file_pos('fgdeng.in',0,beginning).
  eng_file is read('fgdeng.in').
  text('#e', ?eng_file).
  print(?eng_file).
end.

topic sta_input.
  set_file_pos('slopel.in',0,beginning).
  sta_file is read('slopel.in').
  text('#e',?sta_file).
  print(?sta_file).
end.

topic sta_sum.
  text('#e Engineering Summary').
  print('#x3#y2 Engineering Summary').
end.

topic sta_output.
  pcstaout is read('fort.out').
  text('#e', ?pcstaout).
end.

topic sta_graph.
  text('#e Engineering graphic') .
  print('#x3#y2 Engineering Property').
end.

topic sett_input.
  set_file_pos('pcset.in',0,beginning).
  sett_file is read('pcset.in').
  text('#e',?sett_file).
  print('#n #n#n#n#n',?sett_file).
end.

topic sett_sum.
  text('#e Settlement Summary').
  (* print('#x3#y2 Engineering Summary').*)
end.

topic sett_output.
  setfile_out is read('pcset.out').
  text('#e', ?setfile_out).
  (* print('#x3#y2',?setfile_out).*
end.

topic sett_graph.
  text('#e Settlement graphic').
  (* print('#x3#y2 Engineering Property').*)
end.
(*====== All of Summary==================*)

**topic sum_cont.**

- `set_file_pos('summary.out',0,begiining).`
- `all_summ is read('summary.out').`
- `text('#e', ?all_summ).`
- `print('#x3#y2',?all_summ).`

**end.**

**topic close.**

- `close_window( ).`
- `close_window( ).`
- `show_window(?main).`

**end.**

(*report*)

**topic end.**

- `clear( ).`

**end.**

**topic restart.**

- `close_window( ).`
- `close_window( ).`
- `do(!main).`

**end.**

(* The end of the Program *)
new_file('arsenic.in').
filter is concat(fsyst,' = "',?gsystem,'"').
if ?gprocess <> [ ] then
  filter is concat (?filter, ' .and. fproc = "',
                     ?gprocess,'"').
if ?gplant <> [ ] then
  filter is concat (?filter, ' .and. pname = "',
                   ?gplant,'"').

db_use_file ('leachdb.dbf').
db_set_filter (?filter).
db_top_record().
kkk is db_locate_record (rest,,'arsenic').
psum = 0.
  repeat
    psum is ?psum +first(?kkk) and
    kkk is rest (?kkk)
  until ?kkk is [ ].
if ?psum <> 0 then filter is concat(?filter,' .and. '
                                    ',arsenic <> 0.0').'.

db_set_filter (?filter).
db_top_record().
kkk is db_locate_record (rest,,'arsenic').
sqars = 0.
  repeat
    sqars is ?sqars +(first(?kkk)*first(?kkk)) and
    kkk is rest (?kkk)
  until ?kkk is [ ].

(* Total Number of Field Values *)
db_top_record().
tot is db_locate_record (,,'arsenic').
x=total(?tot).
topic total (list).
num = 0.
  apply (addto,?list).
  total = ?num.

  topic addto (x).
  num =?num +1.
end.
end.
(* Summation of field values *)
sum = 0.
repeat
  sum is ?sum + first(?tot) and
  tot is rest (?tot)
until ?tot is [ ].

(* Average of Field Values *)
aars is ?sum/?num.

(* Standard Deviation *)
stdaars = user(?library,sqrt,[?sqars/?num - ?aars*?aars]).
write('arsenic.in','#s', ?aars,stdaars).
db_remove_filter(?filter).

(*===============================================================
  Data Base of Chemical Properties (CaCO3)
  cacovar.kb
===============================================================*)
filter is concat(fsyst,' = "',?gsystem,'"').
if ?gprocess <> [ ] then
  filter is concat (?filter, ' .and. fproc = "',
                  ?gprocess,'"').
if ?gplant <> [ ] then
  filter is concat (?filter, ' .and. pname = "',
                  ?gplant,'"').

db_use_file ('chemdb.dbf').
db_set_filter (?filter).
db_top_record().
kkk is db_locate_record (rest,,,[caco]).
  psum = 0.
  repeat
    psum is ?psum + first(?kkk) and
    kkk is rest (?kkk)
  until ?kkk is [ ].
if ?psum <> 0 then filter is concat(?filter,' .and.
                                   'caco <> 0.0').

  db_set_filter (?filter).
  db_top_record().
  kkk is db_locate_record (rest,,,[caco]).

s qcaco = 0.
repeat
  sqcaco is ?sqcaco +(first(?kkk)*first(?kkk)) and
  kkk is rest (?kkk)
until ?kkk is [ ].
(*) Total Number of Field Values *)
  db_top_record().
  tot is db_locate_record (,,,[caco]).
  x=total(?tot).
  topic total (list).
  num = 0.
  apply (addto,?list).
  total = ?num.
  topic addto (x).
  num =?num +1.
  end.
end.

(*) Summation of field values *)
cnum=?num.
sum = 0.
repeat
  sum is ?sum +first(?tot) and
  tot is rest (?tot)
until ?tot is [].

(*) Average of Field Values *)
acaco is ?sum/?num.

(*) Standard Deviation *)
stdcaco = user(?library,sqrt,[?sqcaco/?num -
  ?acaco*?acaco]).
db_remove_filter(?filter).
if ?acaco is 0 then acaco is ' ' and stdcaco is ' '.

(*=================================================================
 Data Base of Engineering Properties
 (Unconfined Compressive Strength)
 ucsvar.kb
=================================================================*")
new_file('ucs.in').
filter is concat(fsyst, ' = "',?gsystem, '"').
if ?gprocess <> [ ] then
  filter is concat (?filter, ' .and. fproc = "',
              ?gprocess, '"').
if ?gplant <> [ ] then
  filter is concat (?filter, ' .and. pname = "',
              ?gplant, '"').
db_use_file ('engdb.dbf').
db_set_filter (?filter).
db_top_record().
kk is db_locate_record (,,,[ucs]).
psum = 0.
  repeat
    psum is ?psum +first(?kk) and
    kk is rest (?kk)
  until ?kk is [ ].
if ?psum <> 0 then filter is concat(?filter,' .and. '
  ,'ucs <> 0.0').
  db_top_record().
  db_set_filter (?filter).
  kkk is db_locate_record (,,,[ucs]).

squcs = 0.
  repeat
    squcs is ?squcs +(first(?kkk)*first(?kkk)) and
    kkk is rest (?kkk)
  until ?kkk is [ ].

(* Total Number of Field Values *)
  db_top_record().
  tot is db_locate_record (,,,[ucs]).

  x=total(?tot).
  topic total (list).

  num = 0.
  apply (addto,?list).
  total = ?num.

  topic addto (x).
    num =?num +1.
  end.
end.

(* Summation of field values *)
  sum = 0.
  repeat
    sum is ?sum +first(?tot) and
    tot is rest (?tot)
  until ?tot is [ ].

(* Average of Field Values *)
  aucs is ?sum/?num.

(* Standard Deviation *)
  stducs = user(?library,sqrt,?[squcs/?num - ?aucs*?aucs]).
  db_top_record().
  ucsall is db_locate_record (,,,[ucs]).
  ucsall=format_number('#######.#',?ucsall).
  aucs=format_number('#######.#',?aucs).
  stducs=format_number('######.#',?stducs).
  ucsall=format_number('#######.#',?ucsall).
  stducs=format_number('######.#',?stducs).
if ?gprocess is [ ] then wprocess is All.
if ?gprocess <> [ ] then wprocess is ?gprocess.

write('ucs.in',?gsystem,?wprocess, ?num,?aucs, ?stducs,?ucssall).
db_remove_filter(?filter).
APPENDIX C

DATA BASE FOR FGD BY-PRODUCTS AND SOILS
<table>
<thead>
<tr>
<th>PCODE#</th>
<th>PNAME</th>
<th>PLOCT</th>
<th>PSTAT</th>
<th>CAPAT</th>
<th>FSYST</th>
<th>FPROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>McCracken</td>
<td>Columbus</td>
<td>Ohio</td>
<td>63 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>2</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>63 MW</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>3</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>4</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>Ohio</td>
<td>70 MW</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>5</td>
<td>B&amp;W ARC</td>
<td>Alliance</td>
<td>Ohio</td>
<td>1 MW</td>
<td>Dry</td>
<td>LIDS</td>
</tr>
<tr>
<td>6</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>Michigan</td>
<td>300,000 pph</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>8</td>
<td>Northern States Power</td>
<td>Minneapolis</td>
<td>Minnesota</td>
<td>860 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>9</td>
<td>PEDCO</td>
<td>Elizabethton</td>
<td>Tennessee</td>
<td>Dry</td>
<td></td>
<td>RCBB</td>
</tr>
<tr>
<td>10</td>
<td>Quaker State</td>
<td>Elizabethton</td>
<td>Tennessee</td>
<td>Dry</td>
<td></td>
<td>AFBC</td>
</tr>
<tr>
<td>11</td>
<td>Rockport</td>
<td>Rockport</td>
<td>Indiana</td>
<td>Dry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Staley</td>
<td>Decatur</td>
<td>Illinois</td>
<td>62 MW</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>13</td>
<td>TVA Shawnee</td>
<td>Paducah</td>
<td>Kentucky</td>
<td>160 MW</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>14</td>
<td>Burger</td>
<td>Shadyside</td>
<td>Ohio</td>
<td>5 MW</td>
<td>Dry</td>
<td>SNRB</td>
</tr>
<tr>
<td>15</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>100 MW</td>
<td>Dry</td>
<td>Coolside</td>
</tr>
<tr>
<td>MCODE</td>
<td>PNAME</td>
<td>PLOCT</td>
<td>PSTAT</td>
<td>CAPAT</td>
<td>FSYST</td>
<td>FPROC</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>BAB-LID-01</td>
<td>BW ARC</td>
<td>Alliance</td>
<td>Ohio</td>
<td>1 MW</td>
<td>Dry</td>
<td>LIDS</td>
</tr>
<tr>
<td>BAB-LID-02</td>
<td>BW ARC</td>
<td>Alliance</td>
<td>Ohio</td>
<td>1 MW</td>
<td>Dry</td>
<td>LIDS</td>
</tr>
<tr>
<td>BAB-LID-03</td>
<td>B&amp;W ARC</td>
<td>Alliance</td>
<td>Ohio</td>
<td>1 MW</td>
<td>Dry</td>
<td>LIDS</td>
</tr>
<tr>
<td>BEV-DUC-02</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-03</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-05</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-06</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-07</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-08</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>Ohio</td>
<td>585 MW</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BUR-SNR-01</td>
<td>Burger</td>
<td>Shadyside</td>
<td>Ohio</td>
<td>5 MW</td>
<td>Dry</td>
<td>SNRB</td>
</tr>
<tr>
<td>EDG-CLS-01</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>63 MW</td>
<td>Dry</td>
<td>CLS</td>
</tr>
<tr>
<td>EDG-LIM-03</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-04</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-05</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-06</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-07</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-08</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-09</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-10</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-11</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-12</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-13</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>EDG-LIM-14</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>Ohio</td>
<td>Dry</td>
<td>LIMB</td>
<td>Bituminous</td>
</tr>
<tr>
<td>QMC-FLB-01</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>Michigan</td>
<td>Dry</td>
<td>AFBC</td>
<td>Bituminous</td>
</tr>
<tr>
<td>QMC-FLB-02</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>Michigan</td>
<td>Dry</td>
<td>AFBC</td>
<td>Bituminous</td>
</tr>
<tr>
<td>QMC-FLB-03</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>Michigan</td>
<td>Dry</td>
<td>AFBC</td>
<td>Bituminous</td>
</tr>
<tr>
<td>HST-SPD-01</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-02</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-03</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-04</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-05</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>New York</td>
<td>4 MW</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>Code</td>
<td>Plant Name</td>
<td>Location</td>
<td>Capacity</td>
<td>Fuel Type</td>
<td>Type</td>
<td>Additive</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------</td>
<td>----------------</td>
<td>----------</td>
<td>-----------</td>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>NSP-SPD-01</td>
<td>Northern States</td>
<td>Minneapolis</td>
<td>860 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td>Slaked lime</td>
</tr>
<tr>
<td>NSP-SPD-02</td>
<td>Northern States</td>
<td>Minneapolis</td>
<td>860 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-01</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-02</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-03</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-04</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-05</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-06</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-07</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-08</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-09</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-10</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>OSU-SPD-11</td>
<td>McCracken</td>
<td>Columbus</td>
<td>25 MW</td>
<td>Bituminous</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td>PED-RCB-01</td>
<td>PEDCO</td>
<td>Elizabethton</td>
<td>Dry</td>
<td>Bituminous</td>
<td>RCB</td>
<td>Calcitic lime</td>
</tr>
<tr>
<td>QST-FLB-01</td>
<td>Quaker State</td>
<td>Elizabethton</td>
<td>Dry</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Economize</td>
</tr>
<tr>
<td>QST-FLB-02</td>
<td>Quaker State</td>
<td>Elizabethton</td>
<td>Dry</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Baghouse</td>
</tr>
<tr>
<td>QST-FLB-03</td>
<td>Quaker State</td>
<td>Elizabethton</td>
<td>Dry</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Bottom ash</td>
</tr>
<tr>
<td>RPT-FLY-01</td>
<td>Rockport</td>
<td>Rockport</td>
<td>Dry</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Fly ash</td>
</tr>
<tr>
<td>STA-FLB-01</td>
<td>Staley</td>
<td>Decatur</td>
<td>Dry</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Fly ash</td>
</tr>
<tr>
<td>TID-FLB-01</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TID-FLB-02</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Boiler bed</td>
</tr>
<tr>
<td>TID-FLB-03</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TID-FLB-04</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TID-FLB-05</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TID-FLB-06</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TID-FLB-07</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>70 MW</td>
<td>Bituminous</td>
<td>PFBC</td>
<td>Dolomitic lime Primary cyc</td>
</tr>
<tr>
<td>TVA-FLB-01</td>
<td>TVA Shawnee</td>
<td>Paducah</td>
<td>160 MW</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Char overfl</td>
</tr>
<tr>
<td>TVA-FLB-02</td>
<td>TVA Shawnee</td>
<td>Paducah</td>
<td>160 MW</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Fly ash</td>
</tr>
<tr>
<td>TVA-FLB-03</td>
<td>TVA Shawnee</td>
<td>Paducah</td>
<td>160 MW</td>
<td>Bituminous</td>
<td>AFBC</td>
<td>Limestone Fly ash</td>
</tr>
<tr>
<td>CON-WET-01</td>
<td>Consville</td>
<td>Coshocton</td>
<td>Wet</td>
<td>Bituminous</td>
<td>Lime</td>
<td></td>
</tr>
<tr>
<td>MCODE#</td>
<td>PNAME</td>
<td>PLOC</td>
<td>PSTAT</td>
<td>FSYST</td>
<td>FPROC</td>
<td>PH</td>
</tr>
<tr>
<td>------------</td>
<td>----------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-----</td>
</tr>
<tr>
<td>OSU-SPD-01</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
<td>12</td>
</tr>
<tr>
<td>OSU-SPD-02</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
<td>10.53</td>
</tr>
<tr>
<td>OSU-SPD-03</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
<td>12.26</td>
</tr>
<tr>
<td>OSU-SPD-05</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
<td>10.44</td>
</tr>
<tr>
<td>EDG-LIM-06</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIM</td>
<td>10.48</td>
</tr>
<tr>
<td>EDG-LIM-07</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIM</td>
<td>12.14</td>
</tr>
<tr>
<td>EDG-LIM-14</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIM</td>
<td>10.48</td>
</tr>
<tr>
<td>HST-SPD-01</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
<td>11.68</td>
</tr>
<tr>
<td>HST-SPD-02</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
<td>12.23</td>
</tr>
<tr>
<td>HST-SPD-03</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
<td>12.53</td>
</tr>
<tr>
<td>HST-SPD-04</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
<td>12.56</td>
</tr>
<tr>
<td>HST-SPD-05</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
<td>12.1</td>
</tr>
<tr>
<td>TID-FBC-03</td>
<td>Tid</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
<td>11.42</td>
</tr>
<tr>
<td>TID-FLB-04</td>
<td>Tid</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
<td>11.77</td>
</tr>
<tr>
<td>TID-FLB-05</td>
<td>Tid</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
<td>8.44</td>
</tr>
<tr>
<td>BEV-DUC-02</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
<td>12.22</td>
</tr>
<tr>
<td>BEV-DUC-05</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
<td>10.66</td>
</tr>
<tr>
<td>BFR-SNR-01</td>
<td>Shadyside</td>
<td>OH</td>
<td>Dry</td>
<td>SNRB</td>
<td>12.2</td>
<td>0.08</td>
</tr>
<tr>
<td>GMC-FLB-01</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>MI</td>
<td>Dry</td>
<td>AFBC</td>
<td>11.83</td>
</tr>
<tr>
<td>GMC-FLB-02</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>MI</td>
<td>Dry</td>
<td>AFBC</td>
<td>11.1</td>
</tr>
<tr>
<td>GMC-FLB-03</td>
<td>General Motors</td>
<td>Pontiac</td>
<td>MI</td>
<td>Dry</td>
<td>AFBC</td>
<td>12.01</td>
</tr>
<tr>
<td>PED-RCB-01</td>
<td>PEDCO</td>
<td>Elizabethton</td>
<td>TN</td>
<td>Dry</td>
<td>RCBB</td>
<td>12.64</td>
</tr>
<tr>
<td>QST-FLB-02</td>
<td>Quaker State</td>
<td>Dry</td>
<td>AFBC</td>
<td>10.45</td>
<td>0.08</td>
<td>0.25</td>
</tr>
<tr>
<td>QST-FLB-03</td>
<td>Quaker State</td>
<td>Dry</td>
<td>AFBC</td>
<td>10.55</td>
<td>0.08</td>
<td>0.23</td>
</tr>
<tr>
<td>TVA-FLB-03</td>
<td>TVA Shawnee</td>
<td>Paducah</td>
<td>KY</td>
<td>Dry</td>
<td>AFBC</td>
<td>10.48</td>
</tr>
</tbody>
</table>

- Northern States Power: Dry SPD
- Rockport: Dry SPD
- Staley: Dry AFBC
- AEP No1: Wet
- Consville: Wet
- B&W ARC: Dry LIDS
### LEAD (continued)

<table>
<thead>
<tr>
<th>CHROMIUM</th>
<th>LEAD</th>
<th>MERCURY</th>
<th>SELENIUM</th>
<th>SILVER</th>
<th>TDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.011</td>
<td>0.012</td>
<td>0.0002</td>
<td>0.005</td>
<td>0.125</td>
<td>3516</td>
</tr>
<tr>
<td>0.026</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>3194</td>
<td></td>
</tr>
<tr>
<td>0.017</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>0.119</td>
<td></td>
</tr>
<tr>
<td>0.015</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.001</td>
<td>0.0002</td>
<td>0.005</td>
<td>4830</td>
<td></td>
</tr>
<tr>
<td>0.014</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>4838</td>
<td></td>
</tr>
<tr>
<td>0.066</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>1642</td>
<td></td>
</tr>
<tr>
<td>0.059</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>1908</td>
<td></td>
</tr>
<tr>
<td>0.083</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>2508</td>
<td></td>
</tr>
<tr>
<td>0.07</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>2190</td>
<td></td>
</tr>
<tr>
<td>0.077</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>2198</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.001</td>
<td>0.0002</td>
<td>0.005</td>
<td>1.261</td>
<td>3136</td>
</tr>
<tr>
<td>0.011</td>
<td>0.001</td>
<td>0.0002</td>
<td>0.005</td>
<td>5154</td>
<td></td>
</tr>
<tr>
<td>0.02</td>
<td>0.093</td>
<td>0.04</td>
<td>0.3</td>
<td>3.076</td>
<td>9044</td>
</tr>
<tr>
<td>0.026</td>
<td>0.001</td>
<td>0.0002</td>
<td>0.021</td>
<td>2916</td>
<td></td>
</tr>
<tr>
<td>0.027</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>2676</td>
<td></td>
</tr>
<tr>
<td>0.004</td>
<td>0.04</td>
<td>0.04</td>
<td>0.27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>4870</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>5065</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.001</td>
<td>0.0002</td>
<td>0.005</td>
<td>4825</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>5039</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>3626</td>
<td></td>
</tr>
<tr>
<td>0.011</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>4581</td>
<td></td>
</tr>
<tr>
<td>0.024</td>
<td>0.093</td>
<td>0.04</td>
<td>0.27</td>
<td>4603</td>
<td></td>
</tr>
<tr>
<td>MCODE</td>
<td>PNAME</td>
<td>PLOT</td>
<td>PSTAT</td>
<td>FSYS T</td>
<td>FPROC</td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>--------</td>
<td>-------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>OSU-SPD-01</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-02</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-03</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-05</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-10</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>EDG-LIM-03</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-04</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-05</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-06</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-07</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-08</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-09</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-10</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-11</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-12</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-13</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-14</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>TID-FLB-01</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FLB-02</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FLB-04</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FLB-05</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FLB-06</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FLB-07</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>BEV-DUC-02</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-03</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-05</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-06</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-07</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-08</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>Code</td>
<td>Company</td>
<td>Location</td>
<td>Type</td>
<td>Fuel</td>
<td>CF1</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------</td>
<td>-----------</td>
<td>------</td>
<td>------</td>
<td>-----</td>
</tr>
<tr>
<td>GMC-FLB-02</td>
<td>General Motors</td>
<td>Pontiac MI</td>
<td>Dry</td>
<td>FBC</td>
<td>49</td>
</tr>
<tr>
<td>GMC-FLB-03</td>
<td>General Motors</td>
<td>Pontiac MI</td>
<td>Dry</td>
<td>FBC</td>
<td>19</td>
</tr>
<tr>
<td>EDG-CLS-01</td>
<td>Edgewater</td>
<td>Lorain OH</td>
<td>Dry</td>
<td>CLS</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>EPRI HSTC</td>
<td></td>
<td>Dry</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>B&amp;W ARC</td>
<td></td>
<td>Dry</td>
<td>LIDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Northern States Power</td>
<td></td>
<td>Dry</td>
<td>SPD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PEDCO</td>
<td></td>
<td>Dry</td>
<td>RCBB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Quaker State</td>
<td></td>
<td>Dry</td>
<td>AFBC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rockport</td>
<td></td>
<td>Dry</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Staley</td>
<td></td>
<td>Dry</td>
<td>AFBC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TVA Shawnee</td>
<td></td>
<td>Dry</td>
<td>AFBC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Burger</td>
<td></td>
<td>Dry</td>
<td>SNRB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AEP No1</td>
<td></td>
<td>Wet</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Consville</td>
<td></td>
<td>Wet</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MCODE#</td>
<td>PNAME</td>
<td>PLOC T</td>
<td>PSTAT</td>
<td>FSYST</td>
<td>FPROC</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>--------</td>
<td>-------</td>
<td>-------</td>
<td>---------</td>
</tr>
<tr>
<td>OSU-SPD-01</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-02</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-03</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-11</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>OSU-SPD-12</td>
<td>McCracken</td>
<td>Columbus</td>
<td>OH</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>EDG-LIM-06</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-07</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-08</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-09</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-10</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-11</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-12</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-13</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>EDG-LIM-14</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>LIMB</td>
</tr>
<tr>
<td>HST-SPD-01</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-03</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>HST-SPD-05</td>
<td>EPRI HSTC</td>
<td>Somerset</td>
<td>NY</td>
<td>Dry</td>
<td>SPD</td>
</tr>
<tr>
<td>TID-FB-01</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FB-02</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FB-03</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FB-05</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>TID-FB-06</td>
<td>Tidd</td>
<td>Brilliant</td>
<td>OH</td>
<td>Dry</td>
<td>PFBC</td>
</tr>
<tr>
<td>BEV-DUC-05</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>BEV-DUC-06</td>
<td>Muskingum River</td>
<td>Beverly</td>
<td>OH</td>
<td>Dry</td>
<td>DUCT</td>
</tr>
<tr>
<td>EDG-CLS-01</td>
<td>Edgewater</td>
<td>Lorain</td>
<td>OH</td>
<td>Dry</td>
<td>CLS</td>
</tr>
<tr>
<td>PED-RCB-01</td>
<td>PEDCO</td>
<td>Elizabethon</td>
<td>TN</td>
<td>Dry</td>
<td>RCBB</td>
</tr>
<tr>
<td>BUR-SNR-01</td>
<td>Burger</td>
<td>Shadyside</td>
<td>OH</td>
<td>Dry</td>
<td>SNRB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BW ARC</th>
<th>Dry</th>
<th>LIDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rockport</td>
<td>Dry</td>
<td></td>
</tr>
<tr>
<td>General Motors</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>TVA Shawnee</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>Quaker State</td>
<td>Dry</td>
<td>AFBC</td>
</tr>
<tr>
<td>UCS28</td>
<td>OMC</td>
<td>DRYDEN</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>--------</td>
</tr>
<tr>
<td>7502.4</td>
<td>50</td>
<td>56.6</td>
</tr>
<tr>
<td>10252.8</td>
<td>68</td>
<td>51.9</td>
</tr>
<tr>
<td>7344</td>
<td>63</td>
<td>52.6</td>
</tr>
<tr>
<td>12643.2</td>
<td>40</td>
<td>65.8</td>
</tr>
<tr>
<td>39052.8</td>
<td>68</td>
<td>51.9</td>
</tr>
<tr>
<td>74592</td>
<td>43</td>
<td>74.8</td>
</tr>
<tr>
<td>69840</td>
<td>48</td>
<td>69.8</td>
</tr>
<tr>
<td>125568</td>
<td>28</td>
<td>81</td>
</tr>
<tr>
<td>38448</td>
<td>50</td>
<td>65.7</td>
</tr>
<tr>
<td>38592</td>
<td>43</td>
<td>66.6</td>
</tr>
<tr>
<td>42912</td>
<td>50</td>
<td>63.5</td>
</tr>
<tr>
<td>29088</td>
<td>42</td>
<td>69.6</td>
</tr>
<tr>
<td>35424</td>
<td>45</td>
<td>64.9</td>
</tr>
<tr>
<td>8755.2</td>
<td>40</td>
<td>49.2</td>
</tr>
<tr>
<td>7891.2</td>
<td>53</td>
<td>60.7</td>
</tr>
<tr>
<td>4896</td>
<td>65</td>
<td>54.9</td>
</tr>
<tr>
<td>15408</td>
<td>50</td>
<td>58.1</td>
</tr>
<tr>
<td>317088</td>
<td>18</td>
<td>113.1</td>
</tr>
<tr>
<td>185760</td>
<td>18</td>
<td>114</td>
</tr>
<tr>
<td>34272</td>
<td>35</td>
<td>84.7</td>
</tr>
<tr>
<td>220608</td>
<td>25</td>
<td>98.5</td>
</tr>
<tr>
<td>34560</td>
<td>42</td>
<td>70.5</td>
</tr>
<tr>
<td>15984</td>
<td>40</td>
<td>67.5</td>
</tr>
<tr>
<td>59040</td>
<td>32</td>
<td>68.5</td>
</tr>
<tr>
<td>186336</td>
<td>35</td>
<td>79.2</td>
</tr>
<tr>
<td>10497.6</td>
<td>35</td>
<td>71.4</td>
</tr>
<tr>
<td>scode</td>
<td>swc</td>
<td>stuw</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>A-1-a</td>
<td>15</td>
<td>135</td>
</tr>
<tr>
<td>A-1-b</td>
<td>15</td>
<td>135</td>
</tr>
<tr>
<td>A-3</td>
<td>15</td>
<td>135</td>
</tr>
<tr>
<td>A-3-a</td>
<td>15</td>
<td>135</td>
</tr>
<tr>
<td>A-2-4</td>
<td>15</td>
<td>125</td>
</tr>
<tr>
<td>A-2-5</td>
<td>15</td>
<td>125</td>
</tr>
<tr>
<td>A-2-6</td>
<td>15</td>
<td>125</td>
</tr>
<tr>
<td>A-2-7</td>
<td>15</td>
<td>125</td>
</tr>
<tr>
<td>A-4a</td>
<td>16.6</td>
<td>120</td>
</tr>
<tr>
<td>A-4b</td>
<td>23.8</td>
<td>120</td>
</tr>
<tr>
<td>A-5</td>
<td>24</td>
<td>115</td>
</tr>
<tr>
<td>A-6a</td>
<td>24</td>
<td>115</td>
</tr>
<tr>
<td>A-6b</td>
<td>27.5</td>
<td>115</td>
</tr>
<tr>
<td>A-7-5</td>
<td>27.6</td>
<td>115</td>
</tr>
<tr>
<td>A-7-6</td>
<td>40</td>
<td>115</td>
</tr>
<tr>
<td>OL</td>
<td>96</td>
<td>110</td>
</tr>
<tr>
<td>OH</td>
<td>96</td>
<td>110</td>
</tr>
<tr>
<td>Pt</td>
<td>96</td>
<td>110</td>
</tr>
<tr>
<td>None</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX D

FORTRAN PROGRAM FOR SETTLEMENT CALCULATION

(SETFGD)
D1. Settlement Calculation By SETFGD

The following equations were used in the program SETFGD.

D1.1 Primary Consolidation

\[ r_f = \sum H_0 \left( RR \log \frac{S_{vm}}{S_{vo}} + CR \log \frac{S_{vf}}{S_{vm}} \right) \quad \text{Eq. d1} \]

where \( r_f \) = final consolidation settlement
\( H_0 \) = initial height of individual strata
\( CR = \frac{C_C}{1 + e_0} \) = virgin compression ratio
\( RR = \frac{C_r}{1 + e_0} \) = recompression ratio
\( C_C \) = compression index
\( C_r \) = recompression index
\( e_0 \) = initial void ratio
\( S_{vo} \) = initial net overburden stress
\( S_{vm} \) = maximum past stress
\( S_{vf} \) = final vertical effective stress

For normally consolidated soil, \( S_{vm} = S_{vo} \).

For overconsolidated clay, \( S_{vf} = S_{vm} \).

D1.2 Load Distribution by Embankment Fill

Stress increment due to a vertical embankment loading on the surface of a semi-infinite mass was calculated by the equations presented by Poulos and Davis [1974].
where

\[ \Delta p = \frac{q_0}{\pi} \frac{B + A}{A} (\alpha_1 + \alpha_2) - \frac{B}{A} \alpha_1 \]

\[ q_0 = \gamma H \]

\[ \alpha_1 \text{ (radian)} = \tan^{-1} \left( \frac{B}{z} \right) \]

\[ \alpha_2 \text{ (radian)} = \tan^{-1} \left( \frac{A + B}{z} \right) - \tan^{-1} \left( \frac{B}{z} \right) \]

\[ g = \text{unit weight of the embankment soil} \]

\[ H = \text{height of the embankment} \]

**D1.3 Secondary Consolidation**

\[ \rho_S = \sum H_o C_a \log \left( \frac{t}{t_p} \right) \quad \text{Eq d2} \]

where \( \rho_S \) = secondary compression

\( C_a \) = coefficient of secondary compression

\( t \) = time under consideration

\( t_p \) = time for primary consolidation

**D1.4 Rate of Settlement**

\[ U = 1 - \sum_{m=0}^{\infty} \frac{2}{M^2} e^{-M^2 T_v} \quad \text{Eq d3} \]

where \( U \) = average degree of consolidation

\( T_v \) = time factor

\( m \) = an integer

\( M = (2m + 1)p/2 \)

\[ t_p = \frac{(H_d)^2 T_v}{C_v} \quad \text{Eq d4} \]

where \( H_d \) = average longest drainage path during consolidation
Equivalent thickness due to variation of $C_V$ was calculated by the following equation:

$$H' = H_1 \sqrt{\frac{C_V(2)}{C_V(1)}}$$  \hspace{1cm} \text{Eq d 5}$$

where $H'$ = equivalent thickness of $H_1$
$H_1 = \text{top layer}$
$C_V(1) = C_V \text{ at } H_1$
$C_V(2) = C_V \text{ at } H_2$
$H_2 = \text{bottom layer which has } C_V \text{ less than that of } H_1$
Total thickness to evaluate average drainage path = $H_2 + H'$

D2. Source Code For SETFGD

C ********************************************
C SETFGD.FOR
C ********************************************

DIMENSION EMBH(3),EMBT(3),FOUNDH(3),FOUND1(3),
+FOUND2(3),FOUND3(3),FOUND4(3),CR(3),RR(3),
+FOUND5(3),VERTINC(3),OCR(3),CV(3)

OPEN(5,FILE='SETTLE.IN',STATUS='unknown')
OPEN(6,FILE='SETTLE.out',STATUS='unknown')
OPEN(7,FILE='SETTLE2.out',STATUS='unknown')

READ(5,*)SA, THEIGHT, DEPTH, WATERTB, TOPWID,
+CALPHA, NUMEMN
DO 5 I=1, NUMEMN
READ(5,*) EMBH(I),EMBT(I)
5 CONTINUE
READ(5,*) NTYPE, NPATH, COND, TIME, NOUNDN

DO 7 J=1, NOUNDN
READ(5,*) OCR(J), FOUNDH(J), FOUND1(J), FOUND2(J),
+FOUND3(J), FOUND4(J), FOUND5(J), CV(J)
7 CONTINUE

C INITIALIZATION

PI = 3.14159
TOP = TOPWID
BOTTOM = TOP + SA * THEIGHT

DO 156 I =1, NOUNDN
10 CR(I) = FOUND3(I)/(1.0 + FOUND5(I))
RR(I) = FOUND4(I)/(1.0 + FOUND5(I))
156 CONTINUE

C
C CALCULATE DRAIN LENGTH CONSIDERING VARIATION OF CV
C

IF(OUNDN-2) 210, 220, 230

210 DRAIND= FOUNDH(1)
GO TO 399

220 ANEWH1 = FOUNDH(1) * SQRT(CV(2)/CV(1))
DRAIND=FOUNDH(2) + ANEWH1
CVN=CV(2)
GO TO 399

230 ANEWH1 =FOUNDH(1) * SQRT(CV(3)/CV(1))
ANEWH2 =FOUNDH(2) * SQRT(CV(3)/CV(2))
DRAIND = FOUNDH(3) + ANEWH1 + ANEWH2
CVN = CV(3)
GO TO 399

399 IF (NTYPE - 2) 400,410,420

400 IF (NPATH - 1) 450,450,460

450 HD = DRAIND/2.0
TV = 0.848
GO TO 490

460 HD = DRAIND
TV = 0.848
GO TO 490

410 HD = DRAIND
TV =0.94
GO TO 490

420 HD = DRAIND
TV = 0.72

490 TP = TV*(HD)**2.0/CVN
C

C Calculate time - total settlement relation
C

QEMB=0. 0
A = SA * THEIGHT
B = BOTTOM - A
DO 91 I = 1 , NUMEMN
QEMB = QEMB + EMBH(I) * EMBT(I)
91 CONTINUE

FOUNDH(0) =0
DO 200 I = 1 , NOUNDN
FOUNDHH=FOUNDH(I-2)+FOUNDH(I-1)+FOUNDH(I)/2
ALPHA1 = ATAN(B/FOUNDHH)
ALPHA2 = ATAN((A+B)/FOUNDHH) - ALPHA1
EFFI = ((A + B)/A*(ALPHA1 + ALPHA2) - B/A*ALPHA1)/PI
VERTINC(I) =2.0 *QEMB *EFFI
200 CONTINUE

C WATER TABLE - EFFECTIVE STRESS
HH1= FOUNDH(1)/2.
HH2= FOUNDH(1)+FOUNDH(2)/2
HH3= FOUNDH(1) + FOUNDH(2) + FOUNDH(3)/2
HL1 = FOUNDH(1)
HL2 = FOUNDH(1) + FOUNDH(2)
HL3 = FOUNDH(1) + FOUNDH(2) + FOUNDH(3)

IF (NOUNDN .EQ. 1) HH2 = 0
HH3 = 0
HL2 = 0
HL3 = 0
GO TO 551
IF (NOUNDN .EQ. 2) HH3 = 0
HL3 = 0
GO TO 551

551 IF (WATERTB - HH1) 101, 101, 102
101 SIGMVO1 = WATERTB * FOUND1(1) + (HH1 - WATERTB) *
+ (FOUND2(1) - 62.4)
SIGMVM1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)

SIGMVO2 = WATERTB * FOUND1(1) + (HL1 - WATERTB) *
+ (FOUND2(1) - 62.4) + FOUNDH(2) / 2 * (FOUND2(2) - 62.4)
SIGMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)

SIGMVO3 = WATERTB * FOUND1(1) + (HL1 - WATERTB) *
+ (FOUND2(1) - 62.4) + FOUNDH(2) * (FOUND2(2) - 62.4) +
+ FOUNDH(3) / 2 * (FOUND2(3) - 62.4)
SIGMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)
GO TO 1000

102 IF (WATERTB - HH2) 201, 201, 202
201 SIGMVO1 = HH1 * FOUND1(1)
SIGMVM1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)

SIGMVO2 = WATERTB * FOUND1(1) + (HL1 - WATERTB) *
+ (FOUND2(1) - 62.4) + FOUNDH(2) / 2 * (FOUND2(2) - 62.4)
SIGMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)

SIGMVO3 = WATERTB * FOUND1(1) + (HL1 - WATERTB) *
+ (FOUND2(1) - 62.4) + FOUNDH(2) * (FOUND2(2) - 62.4) +
+ FOUNDH(3) / 2 * (FOUND2(3) - 62.4)
SIGMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)
GO TO 1000

202 IF (WATERTB - HH2) 301, 301, 302
301 SIGMVO1 = HH1 * FOUND1(1)
SIGMVM1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)
SIGMVO2 = HL1 * FOUND1(1) + (WATERTB-HL1)*FOUND1(2) ++(HH2-WATERTB)*(FOUND2(2)-62.4)
SIGVMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)

SIGMVO3 = HL1 * FOUND1(1) + (WATERTB-HL1)*FOUND1(2) ++(HL2-WATERTB)*(FOUND2(2)-62.4)+FOUNDH(3)/2*+(FOUND2(3)-62.4)
SIGVMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)
GO TO 1000

302 IF (WATERTB - HL2) 401, 401, 402
401 SIGMVO1 = HH1*FOUND1(1)
SIGVMV1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)

SIGMVO2 = HL1 * FOUND1(1) + (HH2-HL1)*FOUND1(2)
SIGVMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)
SIGMVO3 = HL1 * FOUND1(1) + (WATERTB-HL1)*FOUND1(2) ++(HL2-WATERTB)*(FOUND2(2)-62.4)+FOUNDH(3)/2*+(FOUND2(3)-62.4)
SIGVMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)
GO TO 1000

402 IF (WATERTB - HH3) 501, 501, 502
501 SIGMVO1 = HH1*FOUND1(1)
SIGVMV1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)
SIGMVO2 = HL1 * FOUND1(1) + (HH2-HL1)*FOUND1(2)
SIGVMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)
SIGMVO3 = HL1 * FOUND1(1) + (HH2-HL1)*FOUND1(2) ++(WATERTB-HH3)*FOUND1(3)+ (HH3-WATERTB)*(FOUND2(3)-+62.4)
SIGVMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)
GO TO 1000

502 IF (WATERTB - HL3) 601, 601, 601
601 SIGMVO1 = HH1*FOUND1(1)
SIGVMV1 = OCR(1) * SIGMVO1
SIGMVF1 = SIGMVO1 + VERTINC(1)
SIGMVO2 = HL1 * FOUND1(1) + (HH2-HL1)*FOUND1(2)
SIGVMVM2 = OCR(2) * SIGMVO2
SIGMVF2 = SIGMVO2 + VERTINC(2)
SIGMVO3 = HL1 * FOUND1(1) + (HL2-HL1)*FOUND1(2) ++(HH3-HL2)*FOUND1(3)
SIGVMVM3 = OCR(3) * SIGMVO3
SIGMVF3 = SIGMVO3 + VERTINC(3)

1000 IF(FOUNDN - 2) 701, 702, 703
701 IF (SIGVMV1 .GT. SIGMVF1) SIGMVM1 = SIGMVF1
PRISET = FOUNDH(1) * (RR(1)*ALOG10(SIGMVM1/SIGMVO1)  
++CR(1)*ALOG10(SIGMVFL/SIGMVM1))
DELEP1 = CR(1)*ALOG10(SIGMVFL/SIGMVO1)
EP1 = FOUND5(1) - DELEP1
GO TO 800

702 IF (SIGMVM1 .GT. SIGMVFL) SIGMVM1 = SIGMVFL
IF (SIGMVFL .GT. SIGMVFL2) SIGMVFL = SIGMVFL2
PRISET1 = FOUNDH(1) * (RR(1)*ALOG10(SIGMVM1/SIGMVO1)  
++CR(1)*ALOG10(SIGMVFL/SIGMVFL1))
PRISET2 = FOUNDH(2) * (RR(2)*ALOG10(SIGMVFL2/SIGMVFL2))
PRISET = PRISET1 + PRISET2
DELEP2 = CR(l)*ALOG10(SIGMVFL/SIGMVO1)+
+CR(2)*ALOG10(SIGMVFL2/SIGMVO2)
EP = FOUND5(1) + FOUND5(2) - DELEP2
GO TO 800

703 IF (SIGMVM1 .GT. SIGMVFL) SIGMVM1 = SIGMVFL
IF (SIGMVFL .GT. SIGMVFL2) SIGMVFL = SIGMVFL2
IF (SIGMVFL2 .GT. SIGMVFL3) SIGMVFL2 = SIGMVFL3
PRISET1 = FOUNDH(1) * (RR(1)*ALOG10(SIGMVM1/SIGMVO1)  
++CR(1)*ALOG10(SIGMVFL/SIGMVFL1))
PRISET2 = FOUNDH(2) * (RR(2)*ALOG10(SIGMVFL2/SIGMVFL2))
PRISET3 = FOUNDH(3) * (RR(3)*ALOG10(SIGMVFL3/SIGMVFL3))
PRISET = PRISET1 + PRISET2 + PRISET3
DELEP3 = CR(l)*ALOG10(SIGMVFL/SIGMVO1)+
+CR(2)*ALOG10(SIGMVFL2/SIGMVO2)+
+CR(3)*ALOG10(SIGMVFL3/SIGMVO3)
EP = FOUND5(1) + FOUND5(2) + FOUND5(3) - DELEP3

C SECONDARY CONSOLIDATION SETTLEMENT
C

800 TEP = 10.0 *TP
SECSET = CALPHA/(1.0+EP)*DEPTH*ALOG10(TEP/TP)

C TOTAL SETTLEMENT
TOTSET = PRISET + SECSET

C TOTAL SETTLEMENT - ELAPSED TIME

NLAYER = TP/50
IF (NLAYER.GT.20) NLAYER = 20
TLAYER = NLAYER * 50
WRITE(6,*),NLAYER+1
DO 550 J = 0, TP, 50
UTOO = 0.0
TFAC = CVN * J/(HD)**2.0
DO 250 K = 0, 500
VAR = PI /2.0*(2.0 * K + 1.0)
UOO = 2.0/VAR**2.0*EXP(-VAR**2.0*TFAC)
UTOO = UOO + UTOO

250 CONTINUE
UOO = 1.0 - UTOO
SETRAT = TOTSET * UOO
WRITE(6, 600) J,SETRAT
550 CONTINUE

C calculate the settlement at construction time
TiFa = CVN * time/(HD)**2.0
tUTO = 0.0
DO 910 I= 0,100
VARM = PI /2.0*(2.0*I + 1.)
tUO = 2.0/VARM**2.0*EXP(-VARM**2.0*TiFa)
tUTO = tUO + tUTO
910 CONTINUE
tU = 1.0 - tUTO
CONSET=TOTSET*tU
CONUU = tU*100
REMAIN = TOTSET - CONSET
WRITE(7,949)'/SUNG'
WRITE(6, 900) TOTSET,TP,CONSET,TIME,CONUU,REMAIN
WRITE(7, 950) TOTSET,TP,CONSET,TIME,CONUU,REMAIN
WRITE(7,949)'/HWAN'
100 FORMAT (3X,I10,F10.2)
600 FORMAT (3X,I10,F10.2)
700 FORMAT (3X,F10.2)
900 FORMAT (3X,2F10.2)
949 FORMAT (6A)
950 FORMAT (F10.2)
STOP
END
APPENDIX E

SOURCE CODES FOR GRAPHICAL DESIGN

307
E.1 Drawing for Leachate Properties

Sub Form_Load()
    Show
    Static leacht(20), leach(50), leachq(50), indval(50), qwer(30)
    Const PI = 3.14159 * 2

    Open "leach.in" For Input As #1
    Input #1, title1, title2

    label1.Caption = title1
    label2.Caption = title2

    For I = 1 To 8
        Input #1, leacht(I)
    Next I

    DrawWidth = 5
    For I = 1 To 8
        Input #1, leach(I)
    Next I

    Form1.Print Space(3); "#"; I; leacht(I); Tab(17); leach(I); "(mg/L)"
    Next I
    'Leachate Limit
    lars = leach(1) / 1.5 * 100
    lbar = leach(2) / 30# * 100
    lcad = leach(3) / .3 * 100
    lchr = leach(4) / 1.5 * 100
    lled = leach(5) / 1.5 * 100
    lmer = leach(6) / .06 * 100
    lsel = leach(7) / .3 * 100
    lsil = leach(8) / 1.5 * 100
    'Primary Drinking Water Limit
    pars = leach(1) / .05 * 100
    pbar = leach(2) / 1# * 100
    pcad = leach(3) / .01 * 100
    pchr = leach(4) / .05 * 100
    pled = leach(5) / .05 * 100
    pmer = leach(6) / .002 * 100
    psel = leach(7) / .01 * 100
    psil = leach(8) / .05 * 100

    xspace = 400
    Line (600 + 1 * xspace, 4300)-(600 + 1 * xspace, 4300 - lars * 25), QBColor(2)'arsenic
    Line (600 + 2 * xspace, 4300)-(600 + 2 * xspace, 4300 - lbar * 25), QBColor(2)'barium
    Line (600 + 3 * xspace, 4300)-(600 + 3 * xspace, 4300 - lchr * 25), QBColor(2)'arsenic
    Line (600 + 4 * xspace, 4300)-(600 + 4 * xspace, 4300 - lld * 25), QBColor(2)'barium
E.2 Drawing for Chemical Properties

Sub Form_Load ()
Show
Static compt(20), chem(50), chemq(50), indval(50), qwer(30)
Const PI = 3.14159 * 2
Open "chem.in" For Input As #1
Input #1, title1, title2
label1.Caption = title1
label2.Caption = title2
sum = 0
Input #1, n
For I = 1 To n
Input #1, compt(I)
Next I
DrawWidth = 5
For I = 1 To n
Input #1, chem(I)
s = sum + chem(I)
Form1.Print Space(5); compt(I); Tab(15); chem(I); "(%)"
Next I
'ratio
For I = 1 To n
    indival(I) = chem(I) / sum
Next I

'Circle Drawing
qwer(0) = 0
For I = 0 To n - 1
    qwer(I + 1) = qwer(I) + indival(I + 1)
Next I

For I = 0 To n - 1
    m = I + 1
    If m = n + 1 Then m = 0
    Circle (2000, 2700), 1000 + I * 100, QBColor(I), -PI * qwer(I), -PI * qwer(m)
    Line (2500, 100 + 200 * I)-(2800, 100 + 200 * I), QBColor(I)
Next I

'Circle (2000, 2000), 1200, QBColor(0), -PI * qwer(0), -PI * qwer(1)
'Line (1000, 100)-(1200, 100), QBColor(0)

'Circle (2000, 2000), 1200, QBColor(1), -PI * qwer(1), -PI * qwer(2)
'Line (1000, 100 + 200)-(1200, 100 + 200), QBColor(1)

'Circle (2000, 2000), 1200, QBColor(2), -PI * qwer(2), -PI * qwer(3)
'Line (1000, 100 + 200 * 2)-(1200, 100 + 200 * 2), QBColor(2)

'Circle (2000, 2000), 1200, QBColor(3), -PI * qwer(3), -PI * qwer(4)
'Line (1000, 100 + 200 * 3)-(1200, 100 + 200 * 3), QBColor(3)

'Circle (2000, 2000), 1200, QBColor(4), -PI * qwer(4), -PI * qwer(5)
'Line (1000, 100 + 200 * 4)-(1200, 100 + 200 * 4), QBColor(4)

End Sub

E.3 Drawing for Engineering Properties

Optimum Moisture Content

Sub Command1_Click ()
End
End Sub
Sub Form_Load ()
    Show
    Static omc(50), omcq(50)
    Open "omc.in" For Input As #1
    Input #1, title1, title2

    Input #1, n
    For I = 1 To n + 2
        Input #1, omc(I)
    Next I
    max = 0
    min = 100
    For I = 3 To n + 2
        If omc(I) > max Then max = omc(I)
        If omc(I) < min Then min = omc(I)
    Next I
    label8.Caption = max
    label9.Caption = min

    'y-scale arrangement
    If max <= 50 Then
        yspace = 60
        label2.Caption = 12.5
        label3.Caption = 25
        label4.Caption = 37.5
        label5.Caption = 50
    End If
    If max > 50 Then
        yspace = 37.5
        label2.Caption = 20
        label3.Caption = 40
        label4.Caption = 60
        label5.Caption = 80
    End If

    'X-scale arrangement
    If n <= 5 Then xspace = 500
    If n <= 10 And n > 5 Then xspace = 300
    If n > 10 And n <= 20 Then xspace = 150
    If n > 20 Then xspace = 120

    For I = 1 To n + 2
        DrawWidth = 5
        omcq(I) = 3360 - omc(I) * yspace
        'Line drawing
        Line (600 + I * xspace, 3360)-(600 + I * xspace, omcq(I)), QBColor(3) 'ucs-bar
    Next I

    Line (600 + 1 * xspace, 3360)-(600 + 1 * xspace, omcq(1)), QBColor(5) 'ucs-bar
Dry Unit Weight

Sub Form_Load()
    Show
    Static den(50), denq(50)
    Open "den.in" For Input As #1
    Input #1, title1, title2
    Input #1, n
    For I = 1 To n + 2
        Input #1, den(I)
    Next I
    max = 0
    min = 1000
    For I = 3 To n + 2
        If den(I) > max Then max = den(I)
        If den(I) < min Then min = den(I)
    Next I
    label10.caption = max
    label11.caption = min
    For I = 1 To n + 2
        DrawWidth = 5
        If max <= 80 Then
            yspace = 30
            label2.caption = 100
            label3.caption = 40
            label4.caption = 20
            label8.caption = 80
            label9.caption = 60
        End If
    Next I
End If

If max > 80 Then
    yspace = 20
    label2.caption = 150
    label3.caption = 60
    label4.caption = 30
    label8.caption = 120
    label9.caption = 90
End If

denq(I) = 3360 - den(I) * yspace
Next I

'X-Scale arrangement
If n <= 5 Then xspace = 500
If n <= 10 And n > 5 Then xspace = 300
If n > 10 And n <= 20 Then xspace = 150
If n > 20 Then xspace = 120

'Line drawing
For I = 3 To n + 2
    Line (720 + I * xspace, 3360)-(720 + I * xspace, denq(I)), QBColor(3)'den-bar
Next I

Line (720 + 1 * xspace, 3360)-(720 + 1 * xspace, denq(1)), QBColor(5)'Average
Line (720 + 2 * xspace, 3360)-(720 + 2 * xspace, denq(2)), QBColor(6)'Standard Deviation
Line (720, 3360)-(720 + (n + 3) * xspace, 3360)'X-axis
Line (720, 3360)-(720, 360)'Y-axis
Line (2900, 3600)-(3100, 3600), QBColor(5)'average
Line (2900, 3800)-(3100, 3800), QBColor(6)'standard deviation

DrawStyle = 2
DrawWidth = 1
For I = 1 To 10
    Line (720, 3360 - 300 * I)-(720 + (n + 3) * xspace, 3360 - 300 * I), QBColor(1)'standard deviation
Next I

label6.caption = title1
label7.caption = title2

End Sub

Permeability

Sub Command1_Click ()
End
End Sub
Sub Form_Load()
    Dim max As Double
    Static perm(50), permq(50)
    Open "perm.in" For Input As #1
    Input #1, title1, title2
    Input #1, n
    For I = 1 To n + 2
        Input #1, perm(I)
    Next I
    max = 0
    min = 10
    For I = 3 To n + 2
        If perm(I) > max Then max = perm(I)
        If perm(I) < min Then min = perm(I)
    Next I
    max = 10 ' (max - 10)
    min = 10 ' (min - 10)
    labell1.Caption = max
    labell2.Caption = min
    For I = 1 To n + 2
        DrawWidth = 5
        permq(I) = 3360 - perm(I) * 500
    Next I
    'Y-Scale arrangement
    For I = 3 To n + 2
        Line (720 + I * xspace, 3360)-(720 + I * xspace, 3360) 'perm-bar
        Line (720 + 1 * xspace, 3360)-(720 + 1 * xspace, permq(1)) 'average
        Line (720 + 2 * xspace, 3360)-(720 + 2 * xspace, permq(2)) 'standard deviation
    Next I
    Line (720, 3360)-(720 + (n + 3) * xspace, 3360)'X-axis
    Line (720, 3360)-(720, 360)'Y-axis
    Line (2900, 3600)-(3100, 3600), QBColor(5)'average deviation
    Line (2900, 3800)-(3100, 3800), QBColor(6)'standard deviation
    DrawStyle = 2
    DrawWidth = 1
Sub Command1_Click()
    End Sub

Sub Form_Load()
    Show
    Static swell(50), swellq(50)
    Open "swell.in" For Input As #1
    Input #1, title1, title2
    Input #1, n

    For I = 1 To n + 2
        Input #1, swell(I)
        Next I

    'Check maximum values of swell Value
    max = 0
    min = 1000
    For I = 3 To n + 2
        If swell(I) > max Then max = swell(I)
        If swell(I) < min Then min = swell(I)
    Next I
    labell3.Caption = max
    labell0.Caption = min

    For I = 1 To n + 2
        DrawWidth = 5
        If max > 40 Then
            yspace = 30
            labell2.Caption = 100
            labell3.Caption = 40
            labell4.Caption = 20
            labell8.Caption = 80
            labell9.Caption = 60
End If

If max <= 40 And max >= 20 Then
  yspace = 60
  label2.Caption = 50
  label3.Caption = 20
  label4.Caption = 10
  label8.Caption = 40
  label9.Caption = 30
End If

If max < 20 And max >= 10 Then
  yspace = 150
  label2.Caption = 20
  label3.Caption = 8
  label4.Caption = 4
  label8.Caption = 16
  label9.Caption = 12
End If

If max < 10 Then
  yspace = 300
  label2.Caption = 10
  label3.Caption = 4
  label4.Caption = 2
  label8.Caption = 8
  label9.Caption = 6
End If

swellq(I) = 3360 - swell(I) * yspace
Next I

'X-Scale arrangement
If n <= 5 Then xspace = 500
If n <= 10 And n > 5 Then xspace = 300
If n > 10 And n <= 20 Then xspace = 150
If n > 20 Then xspace = 120

'Line drawing
For I = 3 To n + 2
  Line (720 + I * xspace, 3360)-(720 + I * xspace, swellq(I)), QBColor(3)'swell-bar
Next I

Line (720 + 1 * xspace, 3360)-(720 + 1 * xspace, swellq(1)), QBColor(5)'Average
Line (720 + 2 * xspace, 3360)-(720 + 2 * xspace, swellq(2)), QBColor(6)'Standard Deviation
Line (720, 3360)-(720 + (n + 3) * xspace, 3360)'X-axis
Line (720, 3360)-(720, 360)'Y-axis
Line (3100, 3600)-(3400, 3600), QBColor(5)'average
Line (3100, 3800)-(3400, 3800), QBColor(6)'standard deviation
Unconfined Compressive Strength

Sub Form_Load ()
    Static ucs(50), ucsq(50)
    Open "ucs.in" For Input As #1
    Input #1, title1, title2
    Input #1, n
    For I = 1 To n + 2
        Input #1, ucs(I)
    Next I
    'Check maximum values of UCS Value
    max = 0
    min = 10000000
    For I = 3 To n + 2
        If ucs(I) > max Then max = ucs(I)
        If ucs(I) < min Then min = ucs(I)
    Next I
    label9.Caption = max
    label10.Caption = min

    If max > 100000 Then
        yspace = .01
        label2.Caption = 400
        label3.Caption = 200
        label4.Caption = 100
    End If

    If max <= 100000 And max > 60000 Then
        yspace = .03
        label2.Caption = 100
        label3.Caption = 50
        label4.Caption = 25
    End If
If max <= 60000 And max > 20000 Then
  yspace = .05
  label2.Caption = 60
  label3.Caption = 30
  label4.Caption = 15
End If

If max <= 20000 And max > 10000 Then
  yspace = .15
  label2.Caption = 20
  label3.Caption = 10
  label4.Caption = 5
End If

If max <= 10000 And max > 5000 Then
  yspace = .3
  label2.Caption = 10
  label3.Caption = 5
  label4.Caption = 2.5
End If

If max <= 5000 Then
  yspace = .6
  label2.Caption = 5
  label3.Caption = 2.5
  label4.Caption = 1.25
End If

End If

DrawWidth = 5

For I = 1 To n + 2
  ucsq(I) = 3360 - ucs(I) * yspace
Next I

'X-Scale arrangement
If n <= 5 Then xspace = 500
If n <= 10 And n > 5 Then xspace = 300
If n > 10 And n <= 20 Then xspace = 150
If n > 20 Then xspace = 120

'Line drawing
For I = 3 To n + 2
  Line (600 + I * xspace, 3360)-(600 + I * xspace, ucsq(I)), QBColor(3)'ucs-bar
  Line (600 + 1 * xspace, 3360)-(600 + 1 * xspace, ucsq(1)), QBColor(5)
  Line (600 + 2 * xspace, 3360)-(600 + 2 * xspace, ucsq(2)), QBColor(6)
Next I

Line (600, 3360)-(600 + (n + 3) * xspace, 3360)'X-axis
Line (600, 3360)-(600, 360)'Y-axis
Line (2900, 3600)-(3100, 3600), QBColor(5)'average
deviation
Line (2900, 3800)-(3100, 3800), QBColor(6)'standard
deviation
DrawStyle = 2
DrawWidth = 1
Line (600, 1860)-(600 + (n + 3) * xspace, 1860), QBColor(1)'standard deviation
Line (600, 1110)-(600 + (n + 3) * xspace, 1110), QBColor(1)'standard deviation
Line (600, 2610)-(600 + (n + 3) * xspace, 2610), QBColor(1)'standard deviation

label6.Caption = title1
label7.Caption = title2

End Sub

E.4 Drawing of Failure Surfaces

Sub Command1_Click ()
End
End Sub

Sub Form_Load ()
Show
Dim n As Integer
Static xx1(20), yy1(20), x1(20), y1(20)
Static xx2(20), yy2(20), x2(20), y2(20)
Static ww1(20), ww2(20), w1(20), w2(20)
Static z1(20), z2(20), zz1(20), zz2(20)
Static xxL1(20), yyL1(20), xL1(20), yL1(20)
Static xxL2(20), yyL2(20), xL2(20), yL2(20)

Open "fdesign.in" For Input As #1
'profile
Input #1, n
For I = 1 To n
  Input #1, x1(I), y1(I), x2(I), y2(I)
Next I

'Check maximum values of X and Y coordinators
xmax = 0
ymax = 0
For I = 1 To n
  If x2(I) > xmax Then xmax = x2(I)
  If y2(I) > ymax Then
    ymax = y2(I)
    xcood = x2(I)
    ycood = ymax
  End If
Next I

'X-scale
If xmax <= 50 Then
  xspace = 120
  text1.Text = -10
  text1.Text = 0
texts.Text = 10
text5.Text = 20
label7.Caption = 30
label8(0).Caption = 40
label8(1).Caption = 50

End If
If xmax > 50 And xmax <= 100 Then
xspace = 60
text2.Text = -20
text1.Text = 0
text3.Text = 20
text5.Text = 40
label7.Caption = 60
label8(0).Caption = 80
label8(1).Caption = 100

End If
If xmax > 100 And xmax <= 150 Then
xspace = 40
text2.Text = -30
text1.Text = 0
text3.Text = 30
text5.Text = 60
label7.Caption = 90
label8(0).Caption = 120
label8(1).Caption = "150 (ft)"

End If
If xmax > 150 And xmax <= 200 Then
xspace = 30
text2.Text = -30
text1.Text = 0
text3.Text = 40
text5.Text = 80
label7.Caption = 120
label8(0).Caption = 160
label8(1).Caption = "200 (ft)"

End If
If xmax > 200 Then
xspace = 24
text2.Text = -50
text1.Text = 0
text3.Text = 50
text5.Text = 100
label7.Caption = 150
label8(0).Caption = 200
label8(1).Caption = "250 (ft)"

End If

' Y Scale
If ymax <= 50 Then
yspace = 56
text12.Text = 0
text11.Text = 10
text10.Text = 20
text9(0).Text = 30
text8(0).Text = 40
text7(1).Text = 50
End If
If ymax > 50 And ymax <= 80 Then
  yspace = 35
  text12.Text = 0
  text11.Text = 16
  text10.Text = 32
  text9(0).Text = 48
  text8(0).Text = 64
  text7(1).Text = 80
End If
If ymax > 80 And ymax <= 100 Then
  yspace = 28
  text12.Text = 0
  text11.Text = 20
  text10.Text = 40
  text9(0).Text = 60
  text8(0).Text = 80
  text7(1).Text = 100
End If
If ymax > 100 And ymax <= 120 Then
  yspace = 23.33
  text12.Text = 0
  text11.Text = 24
  text10.Text = 48
  text9(0).Text = 72
  text8(0).Text = 96
  text7(1).Text = 120
End If
If ymax > 120 Then
  yspace = 18.667
  text12.Text = 0
  text11.Text = 30
  text10.Text = 60
  text9(0).Text = 90
  text8(0).Text = 120
  text7(1).Text = 150
End If

'Draw X and Y scales
For K = 1 To 10
  DrawSSStyle = 2
  drawWidth = 1
  Line (600, 3800 - K * 280)-(8600, 3800 - K * 280),
    QBColor(7)
Next K
For K = 1 To 21
Line (600 + K * 400, 3800)-(600 + K * 400, 1000), QBColor(7)
Next K

'Draw Boundary Surface
drawWidth = 4
For I = 1 To n
  xx1(I) = Val(x1(I)) * xspace + 2600
  y1(I) = 3800 - Val(y1(I)) * yspace
  xx2(I) = Val(x2(I)) * xspace + 2600
  y12(I) = 3800 - Val(y2(I)) * yspace
  Line (xx1(I), y1(I))-(xx2(I), y12(I)), QBColor(8)
Next I

'Water table
Input #1, L
For I = 1 To L
  Input #1, zl(I), z2(I)
drawWidth = 3
  zzl(I) = Val(zl(I)) * xspace + 2600
  zz2(I) = 3800 - Val(z2(I)) * yspace
Next I
For I = 1 To L - 1
  Line (zzl(I), zz2(I))-(zzl(I + 1), zz2(I + 1)), QBColor(9)
Next I

'Limits (bed rock boundary)
Input #1, LL
For I = 1 To LL
  Input #1, xL(I), yL(I)
drawWidth = 5
  xxL(I) = Val(xL(I)) * xspace + 2600
  yL1(I) = 3800 - Val(yL(I)) * yspace
Next I
For I = 1 To LL - 1
  Line (xxL(I), yL(I))-(xxL(I + 1), yL(I + 1)), QBColor(5)
Next I

'a dot line at 0 of x-axis
drawWidth = 1
drawStyle = 3
Line (2600, 3800)-(2600, 1000)

'Line draw for slope of Soil Embankment
Input #1, sratio, ratio, fos1, fos2
drawWidth = 3
xleft = (yl(1) - (ycood - xcood / sratio)) * sratio
xlcood = xleft * xspace + 2600
Line (xlcood, 3800 - yl(1) * yspace)-(2600 + xcood * xspace, 3800 - ycood * yspace), QBColor(3)
Line (xlcood - xl(2) * xspace, 3800 - yl(1) * yspace)-(xl(2) * xspace + 2600, 3800 - yl(1) * yspace), QBColor(3)
unitvol = (xl(2) - xleft) * (ycoord - yl(1)) / 2

If fos2 = 0 Then fos2 = "No Value"
Forecolor = QBColor(2)
label5.Caption = unitvol
Labell.Caption = ratio
Labell3.Caption = sratio
labell1.Caption = fos2
labell2.Caption = fos1
End Sub

E.5 Drawing of the Relationship of Time and Settlement

Sub Command1_Click ()
End
End Sub

Sub Command2_Click ()
End
End Sub

Sub Form_Load ()
Show
Static SHT(101), SST(101), SHX(101), SHY(101)
Open "settle1.out" For Input As #1
Open "settle2.out" For Input As #2
Input #1, m
For i = 1 To m
Input #1, SHT(I), SST(I)
Next I

DrawWidth = 5

'x scale for settlement
If SHT(m) <= 200 Then
  xset = 20
  text2.Text = 40
  text3.Text = 80
  text4.Text = 120
  text5.Text = 160
  text6.Text = "200 (days)"
End If

If SHT(m) > 200 And SHT(m) <= 500 Then
  xset = 8
  text2.Text = 100
  text3.Text = 200
  text4.Text = 300

texts.Text = 400

text6.Text = "500 (days)"
End If
If SHT(m) > 500 And SHT(m) <= 800 Then
    xset = 5
    text2.Text = 160
    text3.Text = 320
    text4.Text = 480
    text5.Text = 640
    text6.Text = "800 (days)"
End If
If SHT(m) > 800 Then
    xset = 4
    text2.Text = 200
    text3.Text = 400
    text4.Text = 600
    text5.Text = 800
    text6.Text = "1000 (days)"
End If
'y scale for settlement
If SST(m) <= 1 Then
    yset = 2000
    text8.Text = .2
    text9.Text = .4
    text10.Text = .6
    text14.Text = .8
    text15.Text = 1
End If
If SST(m) > 1 And SST(m) <= 2 Then
    yset = 1000
    text8.Text = .4
    text9.Text = .8
    text10.Text = 1.2
    text14.Text = 1.6
    text15.Text = 2#
End If
If SST(m) > 2 And SST(m) <= 3 Then
    yset = 666.67
    text8.Text = .6
    text9.Text = 1.2
    text10.Text = 1.8
    text14.Text = 2.4
    text15.Text = 3
End If
If SST(m) > 3 And SST(m) <= 5 Then
    yset = 400
    text8.Text = 1
    text9.Text = 2
    text10.Text = 3
    text14.Text = 4
text15.Text = 5
End If

If SST(m) > 5 Then
    yset = 200
    text8.Text = 2
    text9.Text = 4
    text10.Text = 6
    text14.Text = 8
    text15.Text = 10
End If

' Vertical and horizontal lines
For K = 1 To 5
    DrawWidth = 1
    Line (680, 600 + K * 400)-(4680, 600 + K * 400), QBColor(7)
    Line (680 + K * 800, 600)-(680 + K * 800, 2600), QBColor(7)
Next K

' Curve for Time and Settlement
DrawWidth = 3
SHX(0) = 680
SHY(0) = 600
For I = 1 To m
    SHX(I) = 680 + SHT(I) * xset
    SHY(I) = 600 + SST(I) * yset
    Line (SHX(I - 1), SHY(I - 1))-(SHX(I), SHY(I)), QBColor(1)
Next I

Input #2, tl, rl, r2, p1, p2, q1, q2
ForeColor = QBColor(1)
Form1.FontSize = 10

label5.Caption = rl
label6.Caption = r2
label7.Caption = p1
label8.Caption = q1
label10.Caption = q2
End Sub

E.6 Drawing of Embankment Shape

Sub Command1_Click ()
    SetFocus
    Form2.Show
End Sub

Sub Command2_Click ()
    End
End Sub

Sub Command3_Click ()
Sub Form_Load()
    Dim n As Integer
    Static xx1(20), yy1(20), x1(20), y1(20)
    Static xx2(20), yy2(20), x2(20), y2(20)
    Static wW1(200), WW2(200), WL(200), W2(200)
    Static z1(20), z2(20), z1(20), z2(20)
    Static xxL1(20), yyL1(20), xL1(20), yL1(20)
    Static xxL2(20), yyL2(20), xL2(20), yL2(20)

    Open "slopel.out" For Input As #1

    'profile
    Input #1, n
    For I = 1 To n
        Input #1, x1(I), y1(I), x2(I), y2(I)
    Next I

    'Check maximum values of X and Y coordinators
    xmax = 0
    ymax = 0
    For I = 1 To n
        If x2(I) > xmax Then xmax = x2(I)
        If y2(I) > ymax Then ymax = y2(I)
    Next I

    'X-scale
    If xmax <= 50 Then
        xspace = 120
        text2.Text = 5
        text3.Text = 10
        text4.Text = 15
        text5.Text = 20
        text6.Text = 25
        text17.Text = 30
        text18.Text = 35
        text19.Text = 40
        text20.Text = 45
        label2.Caption = 50
    End If
    If xmax > 50 And xmax <= 100 Then
        xspace = 60
        text2.Text = 10
        text3.Text = 20
        text4.Text = 30
        text5.Text = 40
        text6.Text = 50
        text17.Text = 60
        text18.Text = 70
        text19.Text = 80
        text20.Text = 90
    End If
label2.Caption = 100
End If
If xmax > 100 And xmax <= 150 Then
   xspace = 40
   text2.Text = 15
   text3.Text = 30
   text4.Text = 45
   text5.Text = 60
   text6.Text = 75
   text17.Text = 90
   text18.Text = 105
   text19.Text = 120
   text20.Text = 135
   label2.Caption = 150
End If

If xmax > 150 And xmax <= 200 Then
   xspace = 30
   text2.Text = 20
   text3.Text = 40
   text4.Text = 60
   text5.Text = 80
   text6.Text = 100
   text17.Text = 120
   text18.Text = 140
   text19.Text = 160
   text20.Text = 180
   label2.Caption = 200
End If
If xmax > 200 And xmax <= 250 Then
   xspace = 24
   text2.Text = 25
   text3.Text = 50
   text4.Text = 75
   text5.Text = 100
   text6.Text = 125
   text17.Text = 150
   text18.Text = 175
   text19.Text = 200
   text20.Text = 225
   label2.Caption = 250
End If

' Y Scale
If Ymax <= 50 Then
   yspace = 60
   text10.Text = 10
   text8.Text = 20
   text7.Text = 30
   text9.Text = 40
   text11.Text = 50
End If
If Ymax > 50 And Ymax <= 80 Then
   yspace = 37.5
text10.Text = 16
text8.Text = 32
text7.Text = 48
text9.Text = 64
text11.Text = 80
End If
If Ymax > 80 And Ymax <= 100 Then
  yspace = 30
  text10.Text = 20
  text8.Text = 40
  text7.Text = 60
  text9.Text = 80
  text11.Text = 100
End If
If Ymax > 100 And Ymax <= 120 Then
  yspace = 25
  text10.Text = 24
  text8.Text = 48
  text7.Text = 72
  text9.Text = 96
  text11.Text = 120
End If
If Ymax > 120 Then
  yspace = 20
  text10.Text = 30
  text8.Text = 60
  text7.Text = 90
  text9.Text = 120
  text11.Text = 150
End If

'Draw X and Y scales
For K = 1 To 5
  DrawSSStyle = 2
drawWidth = 1
  Line (600, 3800 - K * 600)-(6600, 3800 - K * 600), QBColor(7)
Next K
For K = 1 To 11
  Line (600 + K * 600, 3800)-(600 + K * 600, 800), QBColor(7)
Next K

'Draw Boundary Surface
drawWidth = 3
For I = 1 To n
  xx1(I) = Val(x1(I)) * xspace + 600
  yy1(I) = 3800 - Val(y1(I)) * yspace
  xx2(I) = Val(x2(I)) * xspace + 600
  yy2(I) = 3800 - Val(y2(I)) * yspace
  Line (xx1(I), yy1(I))-(xx2(I), yy2(I)), QBColor(8)
Next I
'Water table
Input #1, L
For I = 1 To L
Input #1, z1(I), z2(I)
drawWidth = 2
zz1(I) = Val(z1(I)) * xspace + 600
zz2(I) = 3800 - Val(z2(I)) * yspace
Next I
For I = 1 To L - 1
Line (zz1(I), zz2(I))-(zz1(I + 1), zz2(I + 1)), QBColor(9)
Next I

'Limits (bed rock boundary)
Input #1, LL
For I = 1 To LL
Input #1, xLl(I), yLl(I), xL2(l), yL2(l)
drawWidth = 5
xxLl(I) = Val(xLl(I)) * xspace + 600
yyLl(I) = 3800 - Val(yLl(I)) * yspace
xxL2(I) = Val(xL2(I)) * xspace + 600
yyL2(I) = 3800 - Val(yL2(I)) * yspace
Line (xxLl(I), yyLl(I))-(xxL2(I), yyL2(I)), QBColor(5)
Next I

'Failure Surface
Input #1, mm
For K = 1 To mm
Input #1, M
For I = 1 To M
Input #1, wl(I), w2(I)
wwl(I) = Val(wl(I)) * xspace + 600
ww2(I) = 3800 - Val(w2(I)) * yspace
Next I
For I = 1 To M - 1
drawWidth = 2
Line (wwl(I), ww2(I))-(wwl(I + 1), ww2(I + 1)), QBColor(K)
Next I
Next K
'draw legends for failure lines
For L = 1 To 10
drawWidth = 5
Line (2280 + (L - 1) * 48, 600)-(2280 + 48 * L, 600), QBColor(L)
Next L
Input #1, sf
Labell.Caption = sf
End Sub
APPENDIX F

SCREENS OF CONSULTATION
Embankment Design Procedures

Generally, the Design of Embankment is developed through following stages [NCHRP Synthesis 33]:

In this program, Geometric conditions (height of embankment, right-of-way width) are provided from stages (1) and (2). Foundation conditions (depth, underlying soil types, water table) are obtained from the preliminary stage.

As an embankment fill material, FGD by-products is selected. Data base incorporated into this system supplies the required information for the embankment and foundation materials. Based on the above information, the engineering behavior of embankment (slope stability and settlement) is predicted. The design of the fill begins considering the Standard Specification, Geometric conditions, and the results of numerical analyses. Finally, construction procedures for the embankment are listed.

Figure F.1 Introduction Screen to IDSSHED

Figure F.2 Procedures of embankment design
Enter Your Information For This Project

Title of the Project:
Ohio Route 541 Embankment Design

Date: 05/05/94

User Name: Sung-Hwan Kim
Position: Section Chief
Company Name: Korea Highway Corporation

Figure F.3 User's information screen

Site Information Obtained from Corridor Study and Route Selection

- Geometric Conditions by Proposed Grade Line
  - Slope of Embankment (a): 2.0
  - Height of Embankment (b): 25.0
  - Depth of Foundation (c): 17.0
  - Half Width of Top (d): 30.0
  - Width of Right-Of-Way (e): 120.0

Figure F.4 Site information screen
Figure F.5 Selection of material type for fill

Figure F.6 Data base for material properties
Figure F.7 Stress analysis methods

Figure F.8 Loading conditions in embankment
Figure F.9 Methods of slope stability analysis

Figure F.10 Profile for slope analysis
Figure F.11 Material properties for slope analysis

Figure F.12 Boundary conditions for slope analysis
Failure Surfaces for Soil Embankment

Possible Failure Surfaces

Critical Failure Surface

Lowest Safety Factor: 1.461

Factors of Safety

- 1.46
- 1.51
- 1.55
- 1.56
- 1.57
- 1.57
- 1.61
- 1.64
- 1.66

Figure F.13 Possible failure surfaces in soil fill

Figure F.14 A critical failure surface in soil fill
Figure F.15 Drainage types

Figure F.16 Number of layers in embankment and foundation
1. Determine OCR value at the middle of each depth.

OCR = $S_{vm}/S_{vo}$

Figure F.17 Average stress history profile

Figure F.18 Inputs for settlement calculation
The Relationship of Elapsed Time and Settlement

Elapsed Time (days) vs Settlement

- Total Settlement: 1.22 [ft]
- Consolidation Time Required: 306.34 [days]
- % at End-Construction: 79.32 [%]
- Settlement down at End-Construction: .96 [ft]
- Remaining Settlement: .25 [ft]

Figure F.19 Relationship of time and settlement

Summary of the Numerical Analysis

- Factor of Safety: 1.461
- Maximum Slope Ratio: 2.0
- Time required for 90% consolidation (days): 306.34
- % Consolidation at End-Construction: 79.32
- Remaining Settlement after End Construction [ft]: .25

Figure F.20 Summary of consultation in Path I
Conclusion In Soil Fill

- Increase stability
- Use higher strength FGD materials
- Use lighter weight FGD materials

Factor of Safety: 1.461
Total Settlement [in]: 1.22
Total Unit Weight of Fill Soil [pcf]: 120
Cohesion of Fill Soil [psf]: 800
Side Slope Ratio: 2.0

- Please select the following option.
  - [ ] Consider FGD Wastes as Fill Material
  - [ ] See Final Design

Figure F.21 Conclusion in Path I

Figure F.22 Selection of FGD material from data base
Figure F.23 Operating information of a material selected

<table>
<thead>
<tr>
<th>Operating Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Material Code:</td>
</tr>
<tr>
<td>Plant Name:</td>
</tr>
<tr>
<td>Location:</td>
</tr>
<tr>
<td>State:</td>
</tr>
<tr>
<td>Capacity(MW)</td>
</tr>
<tr>
<td>FGD System:</td>
</tr>
<tr>
<td>FGD Process:</td>
</tr>
<tr>
<td>Coal Type:</td>
</tr>
<tr>
<td>Sorbent Used:</td>
</tr>
<tr>
<td>Sampling Point:</td>
</tr>
<tr>
<td>Sampling Date:</td>
</tr>
</tbody>
</table>

Figure F.24 Leachate property of a material selected

<table>
<thead>
<tr>
<th>Leachate Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leachate Compounds</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>pH</td>
</tr>
<tr>
<td>Arsenic</td>
</tr>
<tr>
<td>Barium</td>
</tr>
<tr>
<td>Cadmium</td>
</tr>
<tr>
<td>Chromium</td>
</tr>
<tr>
<td>Lead</td>
</tr>
<tr>
<td>Mercury</td>
</tr>
<tr>
<td>Selenium</td>
</tr>
<tr>
<td>Silver</td>
</tr>
</tbody>
</table>
Figure F.25 Chemical property of a material selected

Figure F.26 Engineering property of a material selected
Figure F.27 Selection of default values

Figure F.28 Inputs of leachate properties

A primary concern with the use of FGD wastes in any construction activity is the potential for leaching hazardous chemical components, and the consequent pollution of ground and surface water near FGD-based structures. A waste is categorized as hazardous when the extract obtained by EP or TCLP test has concentrations of any of the regulated metals in excess of the EPA criteria.
Figure F.29 Inputs of chemical properties

Figure F.30 Inputs of engineering properties
Intermediate Conclusions for Material Properties

Leachate Properties
- Your material is within the EPA Leachate Criteria.

Chemical Properties
- pH: 10.5
- Free Lime (%): 5.7
- CaSO4 x CaSO4 (%): 32.7
- Your material may have Long Term Swelling Potential
- Your material may cause Corrosion problem in the structures

Engineering Properties
- Dry Unit Weight (pcf): 104.6 (Natural Material is 90 - 130 pcf)
- UCS (psf): 169432.0 (Natural Material is Less than 4300 psf)
- Swelling Potential (%): 0.9
- Permeability (cm/sec): 0.000000435

You may have Long Term Swelling Potential.
Your material may cause Corrosion problem in the structures.

Figure F.31 Intermediate conclusion

Slope Stability Analysis

Please select a Analysis Method of Slope Stability

- Simplified Bishop Method
- Simplified Janbu Method
- Spencer Method

Limit Equilibrium

<table>
<thead>
<tr>
<th>Moment</th>
<th>Vertical</th>
<th>Horizontal</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBM</td>
<td>0</td>
<td>x</td>
</tr>
<tr>
<td>SJM</td>
<td>x</td>
<td>0</td>
</tr>
<tr>
<td>SPM</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Failure Shapes

- Ctr: Circular Failure Surface
- Irr: Irregular Failure Surface
- Blik: Block Failure Surface
- Spc: A Specific Failure Shape

Select a Failure Shape In Simplified Janbu Method

- Irregular
- Circular
- Irregular
- Specific

Figure F.32 Methods of slope stability analysis (with FGD material)
Figure F.33 Profile for slope analysis (with FGD material)

Figure F.34 Material properties for slope analysis (with FGD material)
Figure F.35 Boundary conditions for slope analysis (with FGD material)

Figure F.36 Possible failure surfaces in FGD fill
Figure F.37 A critical failure surface in FGD fill

Figure F.38 Number of layers in embankment and foundation (with FGD material)
Figure F.39 Inputs for settlement calculation (with FGD material)

Figure F.40 Relationship of time and settlement (with FGD material)
Summary of the Numerical Analysis

<table>
<thead>
<tr>
<th>Soil Fill</th>
<th>FGD Fill</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor of Safety</td>
<td>1.461</td>
</tr>
<tr>
<td>Maximum Slope Ratio</td>
<td>2.0</td>
</tr>
<tr>
<td>Total Settlement (ft):</td>
<td>1.22</td>
</tr>
<tr>
<td>Time required for 90% consolidation (days):</td>
<td>306.34</td>
</tr>
<tr>
<td>% Consolidation at End-Construction:</td>
<td>79.32</td>
</tr>
<tr>
<td>Remaining Settlement after End-Construction (ft):</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Figure F.41 Summary of numerical analysis

Conclusion

- No more modification

- Factor of Safety: 7.900
- Total Settlement (ft): 1.21
- Total Unit Weight of Fill FGD (pcf): 120.4
- Cohesion of Fill FGD (psi): 94741
- Side Slope Ratio: 2.0

- Please select the following option.
  - [ ] Consider Reconsultation
  - [ ] See Final Design

Figure F.42 Conclusion in Path III
Figure F.43 Factors of embankment design

Figure F.44 Design of FGD-based embankment
Choose one of the Side Cover Types

- Type I
- Type II

Figure F.45 Selection of side slope cover

Figure F.46 Summary of design information
Construction Procedures

- Range of Slope: greater than 2.0
- Minimum Moisture Content: 22.8 (%)  
- Dry unit Weight: 184.6 [pcf]
- Compaction Required in the field: 100%
- Compaction Thickness of each layer: 6 to 8 inches
- Compaction Equipment
- Erosion Control
- Compaction during Wet Weather
- Construction during Freezing Weather

Figure F.47 Construction information

List and Print the Consultation Results.
Please select a button shown above.

Figure F.48 Report of Consultation results
APPENDIX G

EVALUATION RESULTS OF IDSSHED
Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are made of, among other features, on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good (Good) Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good (Good) Fair Poor Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td><strong>Very-Good</strong> Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good (Good) Fair Poor Very-Poor</td>
</tr>
</tbody>
</table>
Validation Sheet

Name: Pedro Aymaya
Position: Engineer
Company: American Electric Power
Degree: M.S. Geotechnical Engineering
Area of Experience (years): 8

Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
</tbody>
</table>
Department of Civil Engineering
470 Hitchcock Hall
2070 Neil Avenue
Columbus, Ohio 43210

March 29, 1994

Dear Mr. Kim:

After reviewing the documentation provided, the following comments are offered for your consideration:

*1. Stability analysis program: I believe that the limitation to 14 boundaries restricts the use of the program in common practical problems. For instance, in my experience, most embankment problems are associated with weak foundation soils rather than with the man-made fill portion. Consequently, the use of FGD waste with its high strength and low weight is a plus, as indicated in your write up. However, for important new embankments, such as dikes or even landfills, the stability is controlled by the strength of the foundation soils as we observed in the example run at my office, using parameters and geometry of the Gavin FGD Waste Landfill. Also, in many remediation cases for highway embankments and other type of unstable hillsides, 14 boundaries will limit the ability to model the stratigraphy of the embankment or sliding mass, and of the foundation soils.

*2. Reinforced embankment analysis: It appears that this program does not make a differentiation between the internal stability and the global stability of the reinforced embankment. This is in my opinion and important shortcoming since only the modeling and analysis of the internal stability of a reinforced earth system will lead to the selection of the most efficient (economic) reinforcement element (fabric vs. geogrids, or high vs. low strength geogrids, etc.). In my experience, the use of reinforced systems are only welcome when the economics of the alternative can be fully demonstrated.
Mr. Kim  
March 29, 1994  
Page 2

It is possible that the above concerns or apparent shortcomings would have been overcome should I have had more time to review your work. I also hope these comments are not seen as critical but that they may lead to the upgrading of the present program by additional research and effort. For instance, re-analyzing with this program the embankment on I-77 studied by R. L. Williams - through - B. Randolph. Comparing its result with actual observations would be a valuable evaluation since parameters for I-77 embankments are fully documented in different Ph.D. thesis. Also, I believe that the data base for FGD (both wet and dry) should continue to be upgraded. It may be educational to maintain contact with different utilities and obtain testing information as it becomes available.

Finally, I would like to thank you and Dr. Wolfe for believing that I could contribute to the evaluation of this very-good program that you have developed. I wish you luck in your professional practice.

Very truly yours,

Pedro J. Amaya, P. E.
Civil Engineering

PJA:df/Kirn

cc: Dr. W. E. Wolfe
Validation Sheet

Name: Chien-Tan Chang, P.E. Date: April 4, 1994
Position: Highway Engineer
Company: Federal Highway Administration
Degree: M.S., C.E.
Area of Experience (years):
Twenty-year experience in highway related geotechnical applications

Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good  Good  Fair  Poor  Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good  Good  Fair  Poor  Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good  Good  Fair  Poor  Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good  Good  Fair  Poor  Very-Poor</td>
</tr>
</tbody>
</table>
Recommendations:

1. The proposed intelligent decision support system for highway embankment design using FGD by-product is needed in order to encourage engineers to use this harmless solid waste in their design with confidence. The success of this expert system will rely on a well-defined domain. It is recommended that the researcher should describe clearly about the "facts" and "rules" of the data base. The process of collecting and organizing the facts and rules that are used by experts are of importance to the development.

2. It will be very helpful to include several case studies in Chapter 6 for validation of the system. The cases selected should have detailed engineering records about the performance of the embankment within reasonable years, even be better include some failure cases caused by various reasons.
Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good  Good Fair Poor Very-Poor</td>
</tr>
</tbody>
</table>
Recommendations

1. The system has a very user-friendly interface for engineering modeling and analysis. The program may be further enhanced by using a CAD-type module to draw the boundary and geometry of the embankment and automatically records the coordinates. Because it requires several steps for a complete operation, it will be helpful if the program can have a tutorial demo included in the system.

2. The system may add components to provide interface facilities for both the developer who build the system and for those who use the system, such as "Knowledge acquisition subsystem".
Validation Sheet

Name: SAL NODJOMIAN, CAPT
Company: U.S. AIR FORCE
Degree: M.S.
Area of Experience: GEOTECHNICAL ENGINEERING

Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good</td>
</tr>
<tr>
<td></td>
<td>Good</td>
</tr>
<tr>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td></td>
<td>Poor</td>
</tr>
<tr>
<td></td>
<td>Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good</td>
</tr>
<tr>
<td></td>
<td>Good</td>
</tr>
<tr>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td></td>
<td>Poor</td>
</tr>
<tr>
<td></td>
<td>Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good</td>
</tr>
<tr>
<td></td>
<td>Good</td>
</tr>
<tr>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td></td>
<td>Poor</td>
</tr>
<tr>
<td></td>
<td>Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good</td>
</tr>
<tr>
<td></td>
<td>Good</td>
</tr>
<tr>
<td></td>
<td>Fair</td>
</tr>
<tr>
<td></td>
<td>Poor</td>
</tr>
<tr>
<td></td>
<td>Very-Poor</td>
</tr>
</tbody>
</table>

Recommendations:

1. **User-friendliness**: The system seems a little difficult to use for first time users without the aid of a manual. Ease greatly increases with practice.

2. **Usability**: System should allow user to save file as a different name at various stages.
Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good Good Fair Poor Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good Good Fair Poor Very-Poor</td>
</tr>
</tbody>
</table>
Factors of primary interest in validation process are:

1. **Completeness** refers to the thoroughness of the system and includes checks to determine if the system can address all desired problems within its problem domain.

2. **Efficiency** checks on how well the system makes use of the available knowledge, data, hardware, software, and time in solving problems within its specified domain. For example, an efficient knowledge-based system should be able to reach a useful conclusion on the basis of limited data.

3. **User-friendliness** is a measure of the ease of use of the system. Here evaluations are a function of on-screen help, illustrations, and explanations given to conclusions.

4. **Usability** involves an evaluation of how the system might meet users' needs.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Grades</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completeness</td>
<td>Very-Good, Good, Fair, Poor, Very-Poor</td>
</tr>
<tr>
<td>Efficiency</td>
<td>Very-Good, Good, Fair, Poor, Very-Poor</td>
</tr>
<tr>
<td>User-friendliness</td>
<td>Very-Good, Good, Fair, Poor, Very-Poor</td>
</tr>
<tr>
<td>Usability</td>
<td>Very-Good, Good, Fair, Poor, Very-Poor</td>
</tr>
</tbody>
</table>