COMMONWEALTH OF KENTUCKY
DEPARTMENT OF HIGHWAYS
FRANKFORT, KENTUCKY 40601

CHARLES PRYOR, JR.
COMMISSIONER OF HIGHWAYS

ADDRESS REPLY TO:
DEPARTMENT OF HIGHWAYS
DIVISION OF RESEARCH
533 SOUTH LIMESTONE STREET
LEXINGTON, KENTUCKY 40508
TELEPHONE 606-254-4475

February 28, 1973

H.3.38

MEMORANDUM TO: J. R. Harbison
State Highway Engineer
Chairman, Research Committee

SUBJECT: Research Report No. 358; "Slope Stability Analysis: A Computerized Solution of Bishop's Simplified Method of Slices," KYP-72-38; HPR-1(8); Part III.

In 1965, while lecturing to a graduate class in soil mechanics at the University of Kentucky, R. C. Deen suggested a *schema* for computerizing the solution of the Swedish circle, earth stability problem. A class project ensued but was not completed. Several embankment failures were then being analyzed by tedious graphical methods by our soils engineers in the Research Division -- some were also students in the class. One of the students, H. F. Southgate, was employed part-time in the Division and started the development of the program under Dr. Deen's direction. Perhaps the incentive then was merely to avoid the tedious labor confronting themselves at that time. Those working on landslides then were G. D. Scott, T. C. Hopkins and W. W. McGraw.

Early in 1966, a major fill failure occurred during embankment construction on I 64 in Bath County [I 64 - 6 - (6)117]; the computer program enabled rapid analyses and decisions to be made -- so that construction could proceed *(1, 2)*. By December 1966, three additional slides had been analyzed *(3)*. Those three slides were: 1) US 23, one mile north of Louisa; 2) MP 83, West Kentucky Parkway; and 3) MP 75, West Kentucky Parkway.

As an outgrowth of the computer program, it became more feasible to impress embankment analyses into the design of highways at the very outset. The Division of Materials had the responsibility then for reviewing soil and subsurface reports. It was necessary to staff-up and equip for this added work. It was also necessary to draft new guidelines for subsurface exploration; theretofore drillings were made at bridge sites and to determine qualtities of rock excavation; only limited borings were done otherwise. Little exploration was done in low ground such as culvert sites or for fill foundations. New guides were adopted; the computer program was made available to all consultants; and stability analyses were required for all embankments 20 feet or more in height.

---

[1] "Proposed Remedial Design for Unstable Highway Embankment Foundation;" G. D. Scott and R. C. Deen, April 1966.

[2] "Stability Analyses of Earth Masses," Interim Report on Study No. KYHPR-63-16, HPR-1(2), Part II; R. C. Deen, G. D. Scott, and W. W. McGraw; September 1966.

[3] "Investigation of the Safety Factors Predicted by Theoretical Stability Analyses for Earth Slopes Which Have Failed," W. W. McGraw, MS in CE Thesis (U. of KY., December 1966).

W. W. McGraw transferred to Materials in late 1966. For some time, analyses were made in Research; some were done jointly. Some existing plans were scanned; a site which came under intensive study was the Bull Fork bridge sites on I 64 in Rowan County. There, construction of the embankment on the west side was found to be perilous. The foundation soil in the valley would surely have failed if construction had proceeded without the constraints recommended. The east embankment at the same site was suspected (intuitively) of presenting problems some time in the future but was not analyzed. It is worthy of note, here, that the east embankment has since become the subject of study because of severe settlement of the approach and indications of a slip failure in the upper reaches of the fill.

In several instances, the stability analysis has been used in conjunction with settlement analyses. The I 71 crossing of the Kentucky River at Carrollton, the Big Eddy Creek crossing of I 24 over Barkley Lake, the Green River bridge at Sebree, and the Parkway bridge over the Green River at Morgantown are some instances where construction has been safeguarded by these combined analyses.

In 1969, McGraw left the Department, and Gordon Scott transferred to the Division of Materials.

In November 1969, an embankment failure occurred during the construction of I 64 in Louisville (between Grinstead Drive and Cherokee Park, US 60). On-site decisions were made to construct a berm; movement continued; we were able to synthesize the slide in the computer program and found that the berm first visualized did not provide sufficient counterbalance; no work was lost; the berm was merely enlarged; and the work proceeded without much delay.

I am sure you will be pleased to know that there has not been a single instance of embankment failure on any new construction where there has been due overview from the standpoint of embankment stability at the design stages. This overview dates from 1967. Indeed, the credit for this degree of success belongs to the soils engineers and geologists in the Division of Materials.

The purpose of this submission is to present a revised, more versatile computer program and to recommend its adoption -- supplanting the program now in use. I am using this device not only to obtain your assent to the recommendation but also to inform you briefly of a true success story.

Respectfully submitted,

Jas. H. Havens

Jas. H. Havens
Director of Research

JHH:dw
attachment
cc's: Research Committee

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Slope Stability Analysis: A Computerized Solution of Bishop's Simplified Method of Slices

5. Report Date
February 1973

6. Performing Organization Code

7. Author(s)
S. M. Yoder, Research Engineer Assistant
T. C. Hopkins, Research Engineer Senior

8. Performing Organization Report No.
358

9. Performing Organization Name and Address
Division of Research
Kentucky Department of Highways
533 South Limestone
Lexington, Kentucky 40508

10. Work Unit No.

11. Contract or Grant No.
KYHPR- 64-17

12. Sponsoring Agency Name and Address

13. Type of Report and Period Covered
Interim

14. Sponsoring Agency Code

15. Supplementary Notes
Prepared in cooperation with the US Department of Transportation, Federal Highway Administration

Study Title: Settlement of Bridge Approaches and Embankments

16. Abstract

A computer program based on Bishop's simplified method of slices (1954) and capable of analyzing the slope stability of a multilayered soil mass is described. The computer program was specifically developed for analyzing the slope stability of highway bridge approach embankments; however, it can be applied to a broad spectrum of practical slope configurations and bearing capacity problems. Details of the use, applications, and accuracy of the program are presented. Important features of the computer program include a grid type, search operation for locating the critical shear surface and a ledger printout of the forces acting on each individual slice. The latter feature was included so that results of the computer program could be compared to those obtained from manual computations. Pore pressures in the computer program are handled in a manner described by Bishop (1954). Additionally, for seepage cases, infinite slope conditions are assumed and used to simulate a flow net.

17. Key Words
Slope Stability, Computer Program, Slices, Circular Failure, Bishop

18. Distribution Statement

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of Pages

22. Price

Form DOT F 1700.7 (8-69)

Research Report
358

SLOPE STABILITY ANALYSIS: A COMPUTERIZED SOLUTION OF BISHOP'S SIMPLIFIED METHOD OF SLICES

Interim Report

KYHPR-64-17; HPR-1(8), Part II

by

S. M. Yoder
Research Engineer Assistant

and

T. C. Hopkins
Research Engineer Senior

Division of Research
DEPARTMENT OF HIGHWAYS
Commonwealth of Kentucky

in cooperation with the
U. S. Department of Transportation
Federal Highway Administration

February 1973

INTRODUCTION

Material presented in this report is the result of an effort to devise a computer program capable of analyzing the slope stability of a multilayered soil mass and describes procedures for applying this computer program to a broad spectrum of practical slope configurations. The program is based on the simplified Bishop method of slices, which assumes a circular slip surface. It can be used to search for coordinates of the center of the critical circle which has the least factor of safety and can also determine the factor of safety for a defined slip circle. The computer program, (Deen, Scott, and McGraw, 1966) originated in the Kentucky Department of Highways Division of Research as a slope stability program using the Fellinius method. The work reported herein makes use of the simplified Bishop method and makes the program more universally adaptable.

METHOD OF ANALYSIS

The method of analysis used in this computer program is the simplified Bishop method of slices (Bishop, 1954), and is based on a limiting equilibrium condition. Figure 1 shows a free body (or vertical slice of soil lying above an assumed circular slip surface); using known or assumed forces acting on the slice, the shearing resistance of the soil required for equilibrium is calculated. The ratio of the shear strength of the soil to the calculated shearing resistance required indicates the factor of safety for the slope. Forces and dimensions on Figure 1 are defined as follows:

E_n, E_{n+1}	are the resultants of the total horizontal forces on the slice,
X_n, X_{n+1}	are the vertical shear forces,
W	is the weight of the slice,
N	is the total normal force on the base,
R	is the radius of the slip circle,
S	is the shear force on the base,
U	is the boundary water force,
l	is the arc length of an assumed slip surface for the slice,
b	is the width of the slice,
θ	is the angle between S and the horizontal,
O	is the center of the circle and point of rotation, and
x	is the horizontal distance from the center of the slice to O.

Using the definition of the factor of safety and the Mohr-Coulomb failure criterion, the mobilized shear stress can be written in terms of the shear strength as

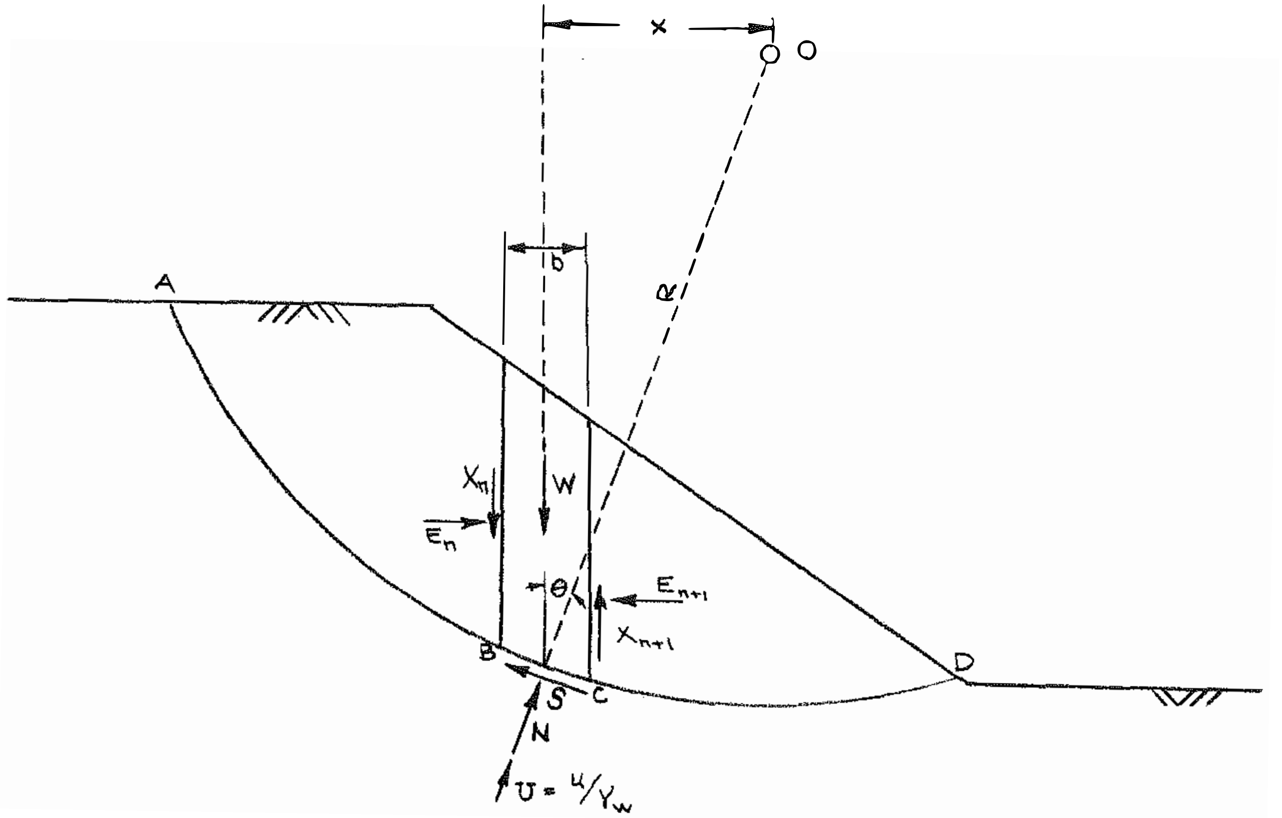


Figure 1. Forces on Slice in Bishop Method.

$$s = (c' + \sigma' \tan \phi')/F \qquad 1$$

where c' and ϕ' are effective strength parameters, σ' is the normal effective stress and F is the factor of safety.

Taking moments around 0 of the weight of the soil and the external forces acting on the slice on the circluar arc and assuming equilibrium conditions:

$$\Sigma Wx = \Sigma SR = \Sigma s l R. \qquad 2$$

Noting that $\sigma' = N/l - u$, it follows from Equations 1 and 2 that

$$F = R\ \Sigma[c'l + (N - ul) \tan \phi'] / \Sigma Wx, \qquad 3$$

where u = boundary pore water pressure. Bishop observed that more accurate solutions (especially for deep slip circles where an appreciable change in θ can occur) were obtained by solving for and resolving the normal forces vertically. Doing so, and letting $l = b \sec \theta$, the factor of safety becomes

$$F = R\ \Sigma[\ \{c'b + \tan \phi' (W + X_n - X_{n+1})\} \sec \theta / \{1 + (\tan \phi' \tan \theta)/F\}\ ] / \Sigma Wx. \qquad 4$$

Horizontal side forces do not appear in Equation 4 since forces were resolved vertically.

In addition to the circular failure assumption, Bishop concludes that $(X_n - X_{n+1})$ can be taken to be zero throughout the arc without significant error -- typically less than one percent. This conclusion was verified by Whitman and Bailey (1967). They solved many problems using this assumption and a statically accurate method (Morgenstern-Price) and found that the resulting error was seven percent or less. Usually, the error was two percent or less.

Noting that $x = R \sin \theta$, Equation 4 can be simplified to

$$F = \Sigma[\ \{c'b + (W - ub) \tan \phi'\} \sec \theta / \{1 + (\tan \theta \tan \phi')/F\}\ ] / \Sigma W \sin \theta. \qquad 5$$

One additional point of clarification is necessary to use Equation 5 for a broad range of cases; that is the weight (W) of the slice must be defined exactly. Referring to Figure 2, the driving moment of the soil mass above the circular arc BF is found to be the moment of the total weight of the soil,

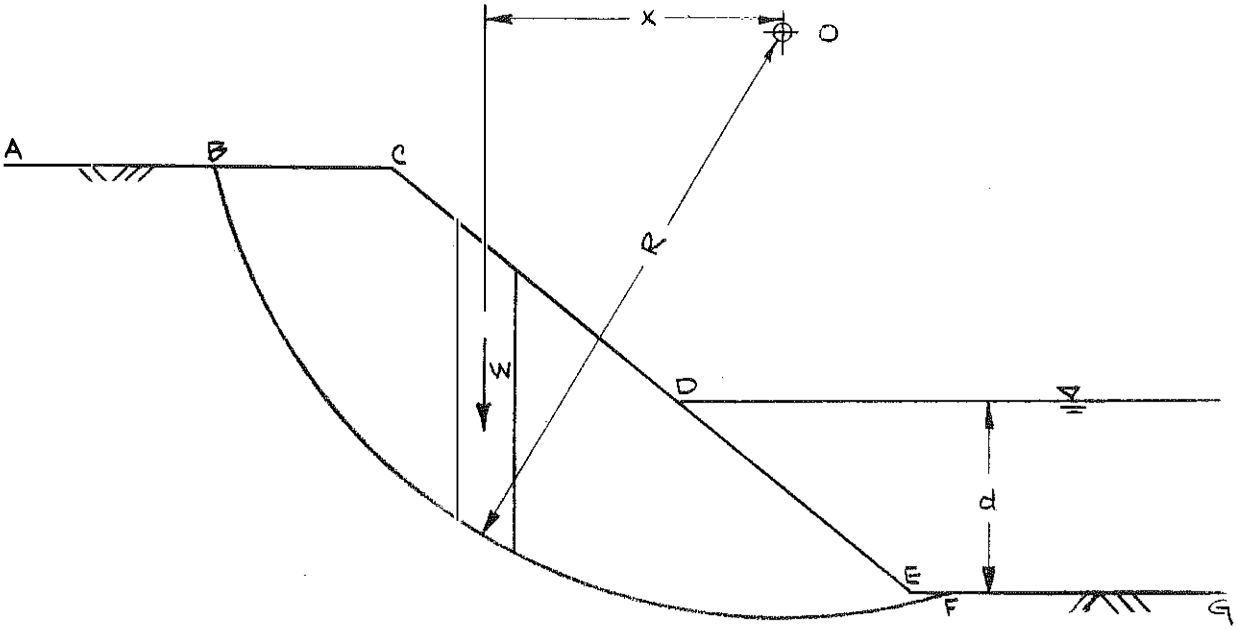


Figure 2. Partially Submerged Slope.

including pore water, about 0 minus the moment of the water pressure acting on the surface DEF about 0.

When computing the effective normal force, the pore pressure acting on the base of the slice is calculated as

$$u = u_s + z\gamma_w. \qquad 6$$

The term u_s is the excess pore pressure above the simple static pore pressure, $z\gamma_w$, and is due to seepage and(or) consolidation (Figure 3). Therefore, whenever u_s is zero, the pore pressure is defined only by the height of a static water column in the slice. Similarly, if there is no static water-table condition in the cross section, then $u = u_s$ since $z = 0$. Therefore, the simplified Bishop equation as used in the program becomes:

$$F = \Sigma \left\{[c'b + (W - ub) \tan \phi'] \sec \theta/[1 + (\tan \phi' \tan \theta)/F]\right\} / \Sigma (W - z\gamma_w) \sin \theta. \qquad 7$$

where W is now defined as the weight of the slice using the total unit weight of the soil.

Equation 7 gives the factor of safety for a particular circle. However, it must be realized that there are several limitations of this method: 1)static equilibrium is not satisfied, 2) strength of the soil is described by the Mohr-Coulomb equation, 3) the slip surface is circular, and 4) the factor of safety is uniform over the entire arc. Consequently, the user of the program must decide whether these limitations will appreciably influence results of the analysis. Whitman and Bailey (1967) pointed to another difficulty. Whenever the factor of safety is less than 1.0 and the pore pressures are large, the numerator in Equation 7 may become negative. This necessitates some sort of warning in the program to indicate the numerator is negative or the denominator is negative or small, i.e.

$$\cos \theta + (\tan \phi' \sin \theta)/F < 0.2. \qquad 8$$

In the program, the left side of Equation 8 is designated $M_i(\theta)$ or MTHETA; whenever it is less than 0.2, a warning is printed out. In the event of such a warning, the user must examine the result in detail and possibly use other methods to analyze the particular circle.

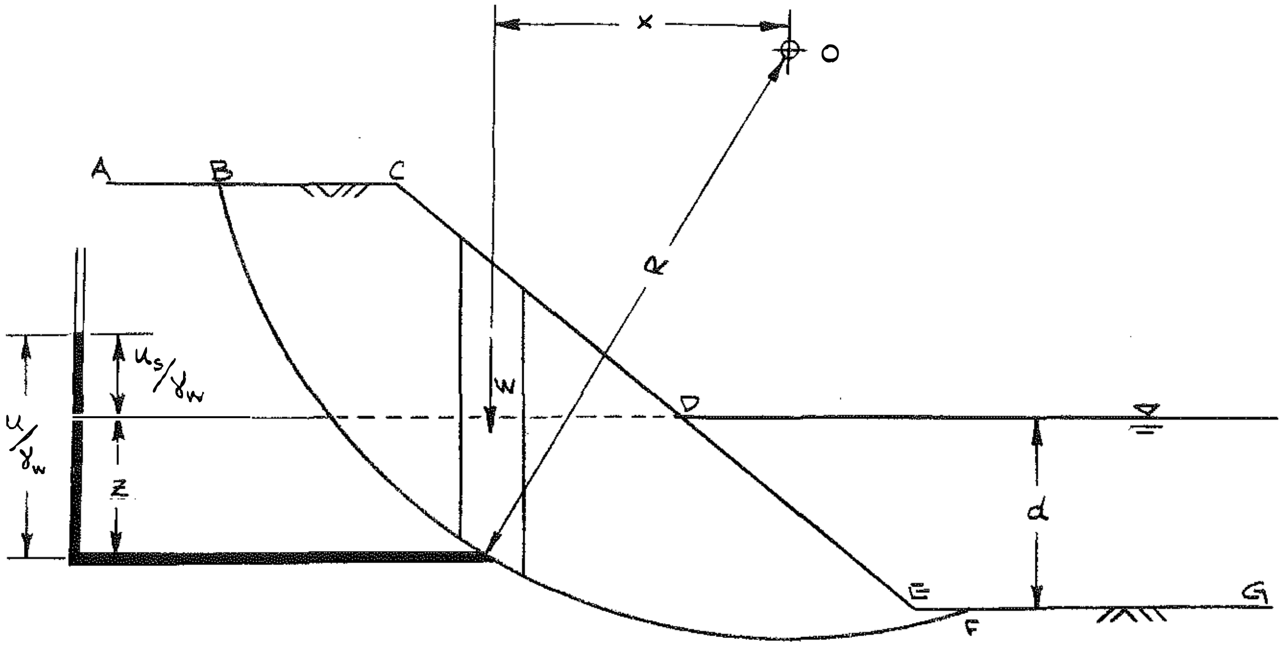


Figure 3. Calculation of Pore Pressure.

PORE PRESSURE CALCULATIONS

The program computes the pore water pressure on the slip circle as a piezometric head calculated from the phreatic surface. The manner in which this head is determined depends on the type of water table specified in the data deck. These water table conditions can either be a phreatic surface or an actual piezometric line.

Since the actual piezometric line can be determined for only one circle at a time, it is desired to use a simple approximation to estimate the piezometric line from the actual phreatic surface. The program computes a piezometric head for each slice by assuming that the water flows as if on an infinite slope (see Figure 4). Here the pore pressure at A is u and the vertical height to the phreatic surface is h. It follows that the piezometric head at A is

$$u = h\gamma_w \cos^2 i. \qquad 9$$

Since the program uses straight line approximations for all lines, including the water table, the angle i is computed for each water table line segment. This computation will also be made for a static water table since $i = 0$ and $\cos^2 0 = 1$. Therefore, the pore pressure is $u = h\gamma_w$.

This method, however, will produce errors when the phreatic surface changes slope rapidly. When the slope changes rapidly, the flow net is not made up of straight lines and the pore pressure at a point on the failure arc is not given exactly by Equation 9. Figure 5 illustrates this error for point A. In cross sections with a severe change of slope, such as when a drain is placed under a fill, the error was found to be less than one percent when compared to solutions using the actual-piezometric line for computation of pore pressures.

In the program, the piezometric line for a particular circle can be input directly. This allows the exact pore pressure to be used in calculations for each slice and a more accurate solution may be obtained. When this actual piezometric head or "effective water table" is used, the pore pressures are calculated as a vertical height of water rather than using Equation 9 as an approximation.

INPUT INSTRUCTIONS

The following is a guide for the entire data deck. It may be used with either a source or object deck. In the Appendices, a source deck listing, a detailed explanation of the program process, and several example problems with data input are shown.

There are several requirements with respect to the input of the cross-sectional geometry. The entire cross section must be approximated by straight line segments. This applies to the ground line, layer

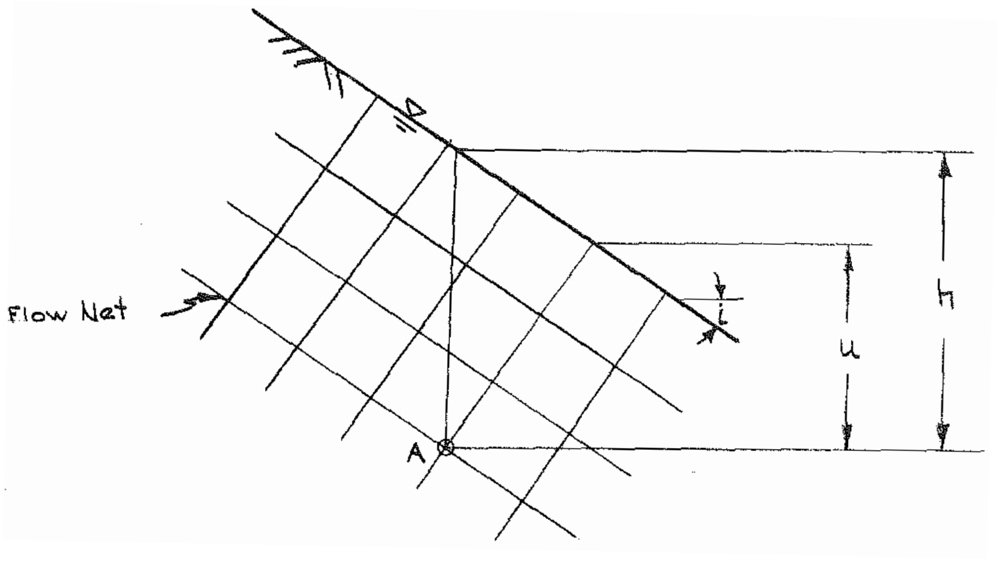


Figure 4. Infinitely Sloping Water Table.

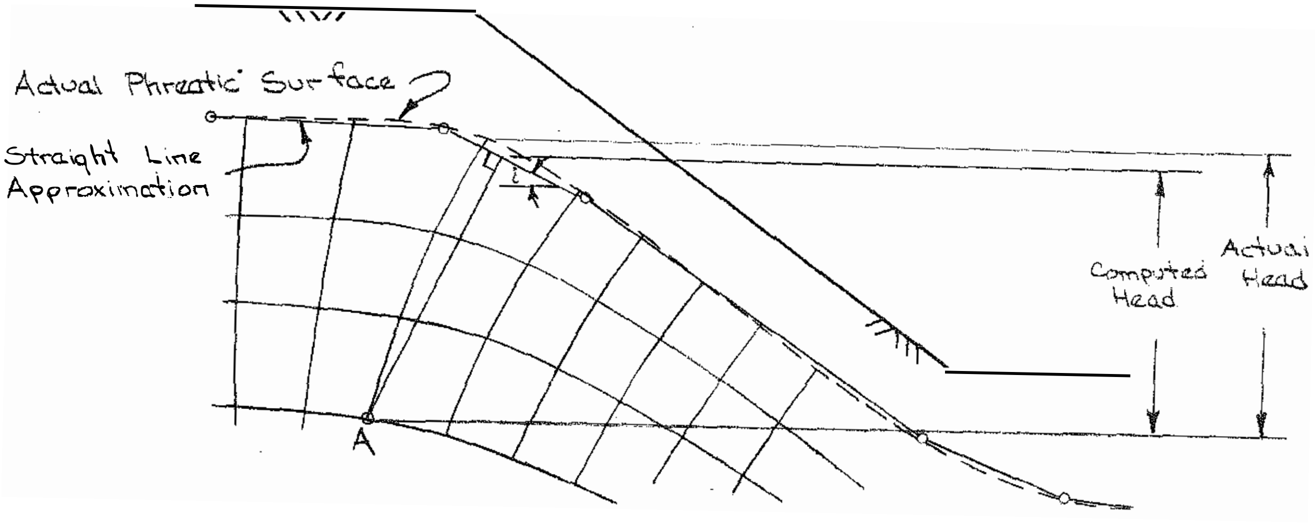


Figure 5. Error in Pore Pressure Calculation.

boundary lines, and the water table line. These line segments are defined by coordinate points X and Y (Figure 6). All lines must be continuous and run throughout the entire cross section. In addition, the cross section must have a general negative slope with respect to the customary X-Y coordinate system.

Although no prior knowledge of computer programming is required, a few basic points must be understood to properly apply these instructions:

1. Anytime a number is required to be integer, a decimal point must not be used. If a number is to be real, a decimal point must be used even if it is a whole "integer" value.
2. The term "justified right" appears many times in this outline. It simply means that when a number is punched on a data card, it has been alloted a certain number of spaces and the number must be positioned so as to leave no blank spaces to the right of the number in the alloted spaces. For example, if the number 4071 is to be punched in Columns 1 through 10, justified right, the digit 4 will have to be in Column 7 to allow the last digit, 1, to be in the last allocated column, 10.
3. Any capitalized term refers to the variable exactly as it is found in the program. A complete list of these variables can be found in the appendices.

Figure 7 and the following descriptions illustrate the manner in which all problems are to be submitted.

I/O Card

This card specifies the method of input, method of output, and number of problems to be solved.

I. Input: IN

Place in Column 4 the proper input code corresponding to the manner in which data will be entered into the computer.

II. Output: IOUT

Place in Column 8 the proper output code corresponding to the manner in which output is desired.

III. Number of Problems: NOP

Place the integer value of the number of problems to be solved in Columns 9 through 12, justified right.

DATA DECK

I. Heading Cards

These two cards provide all identification information found on the first page of the printout. These cards may use any alphanumeric character.

A. In Columns 1 to 24, place the identification of the problem.

B. In Columns 25 to 34, place the route designation.

C. In Columns 35 to 46, place the county name or abbreviation.

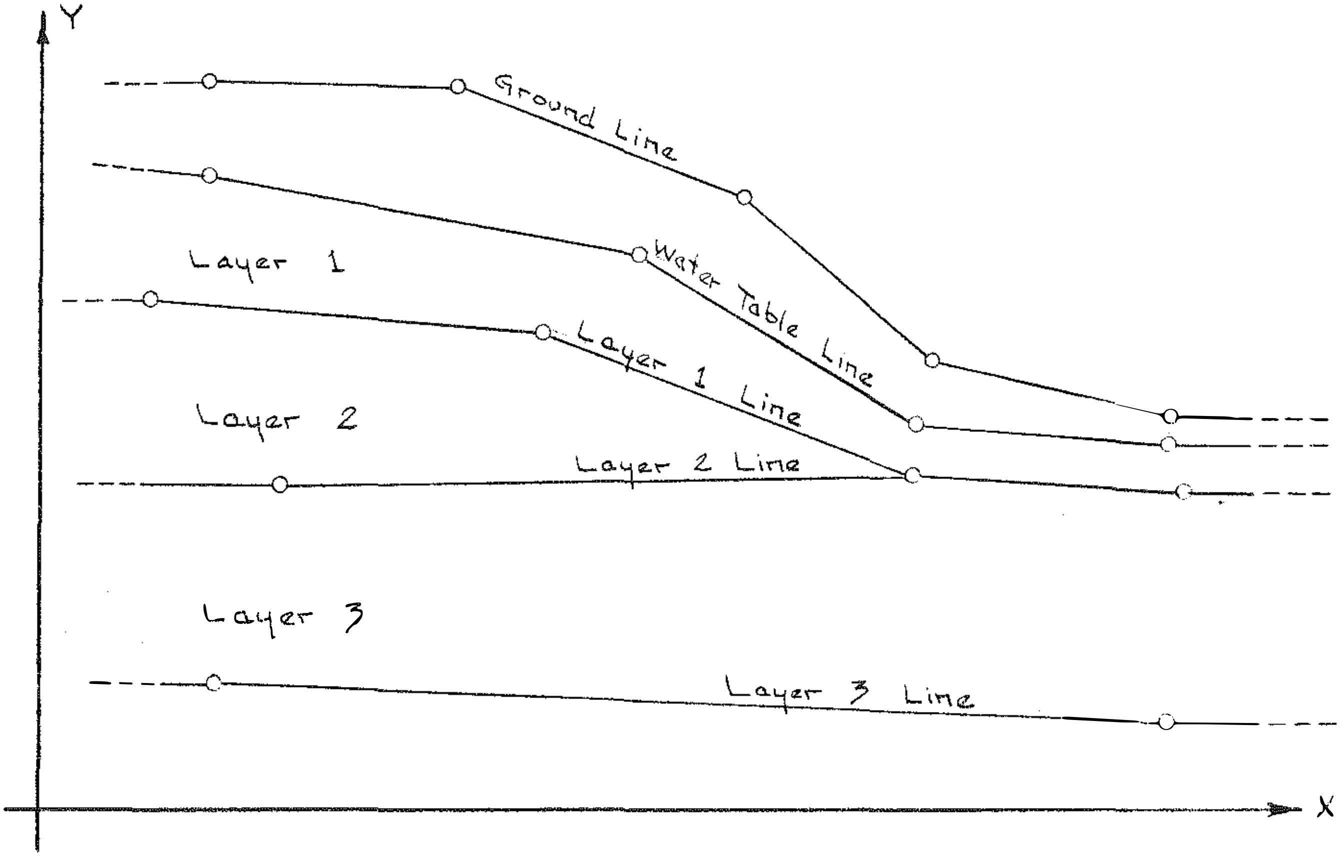


Figure 6. Cross-section Coordinate System.

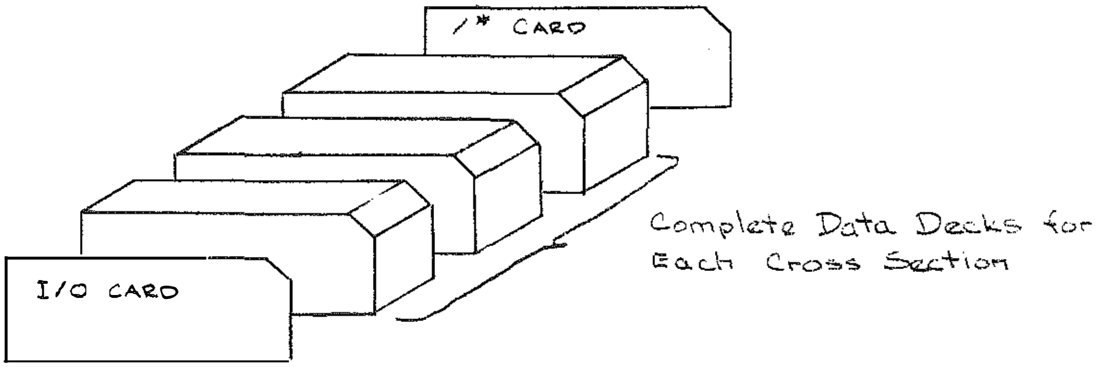


Figure 7. Punched Card Input.

D. In Columns 47 to 52, place the analysis number.

E. In Columns 53 to 66, place any project number.

F. In Columns 67 to 79, place the project designation.

G. If additional information is required to fully identify the problem, place any digit (1 through 9) in Column 80. This directs that another heading card be read; that card can contain any alpha-numeric description in all 80 columns. This information will be printed in the heading of the output.

H. In Columns 1 to 6 on the next card, encode the date, making sure to use two digits each for the month, day and year.

NOTE: If some of this information is not necessary or desired, it may be omitted without any operational difficulties. The output will simply leave blank spaces for any data not submitted. However, there must be two non-blank cards in this position to insure proper computation.

II. Gimmick Card

In this program, there are several routing sequences that require specification by the user. This card is used to specify these routings. The use of a 0.,1., or 2. in the proper columns is all that is necessary, but the user must be aware of their significance.

A. Gimmick 1: SOIL

Place a 0., 1., or 2. in Columns 4 and 5 according to the following:

1. A 0. implies normal printout, i.e. the factor of safety for each circle.
2. A 1. implies a grid of the lowest values of the factor of safety for each circle center. This is in addition to the normal printout. It is very useful in searching for the critical circle(s) because this option searches each grid point and indicates the minimum factor of safety for each point.
3. A 2. directs the computer to supply the values of the X and Y coordinates and minimum factor of safety for each grid point as output on punched cards. This is in addition to normal output on the printout. It is useful when a contouring program is available to plot the grid in X and Y and with the minimum factor of safety for each grid point as the Z value. The contouring program can accept these cards as input and plot the grid; contour lines show areas of low factors of safety as depressions.

B. Gimmick 2: SOIL1

Place a 0. or 1. in Columns 9 and 10 according to the following:

1. A 0. directs the computer to calculate the factor of safety for each required slip circle

that intersects the cross section.

2. A 1. directs the computer to calculate the factor of safety for each circle that lies above the bottom layer. This should be used whenever the bottom layer is much stiffer (such as a rock layer) than the overlying layers. The factor of safety for a slip circle passing through a rock layer will be extremely higher than those passing through the weaker layers and meaningless answers will result. However, when using a grid, it is good practice to use a value of 1. in most cases and provide a low bottom layer. This will eliminate calculations for unnecessarily deep circles.

B. Gimmick 3: POUT

Place a 0. or 1. in Columns 14 and 15 according to the following:

1. Use a 0. if detailed output is desired in addition to normal output. Detailed output consists of all quantities used to compute the factor of safety, tabulated for each slice. This option is very useful in checking the program against a hand calculation. (It is highly recommended to do at least one hand calculation for each cross section to verify the program and data deck. There are many cases where one wrong number in the data deck might cause large errors in the factor of safety. This keypunching error might very easily be overlooked. The user could detect this error by working through one hand calculation.)
2. Use a 1. if normal output is all that is necessary. Normal output consists of X and Y coordinates of the circle center, radius of the circle, factor of safety, area of failure (cross-sectional area above slip circle), and the X and Y coordinates of the intersection of the slip circle with the ground line.

D. Gimmick 4: EFFWT

Place a 0. or 1. in Columns 19 and 20 according to the following:

1. A 0. should be used only when an effective water table is used to allow for an excess pore pressure due to seepage or consolidation. In this case, pore pressures are calculated using a vertical distance from the circular surface to the effective water table line.
2. A 1. should be used whenever an effective water table is not used. In general, a 1. can be used in all cases without serious error in the factor of safety. However, if seepage is present, a more accurate analysis can be obtained by using a 0. and an effective water table.

A complete discussion of the manner in which pore pressures are calculated can be found in the discussion of Bishop's method of slices. Procedures for analyzing problems can be summarized with respect to the effective water table:

Case 1. With a static water table, either a 0. or 1. may be used.

Case 2. With a sloping water table, use a 1. to locate the critical circle. If only one circle is being analyzed, plot that circle on the cross section and sketch in the flow net. Then draw an effective water table which corresponds to the pore pressures along the critical circle. Use this effective water table in the program and use a 0. for Gimmick 4. This gives a more exact solution.

Case 3. Use a 0. whenever the piezometric surface is used.

III. General Information Card

A. Number of Slices: NSLICE

Place the integer value of number of slices desired for analysis in Columns 1 through 4, justified right. The maximum number of slices that may be used is 50.

B. Number of points defining water table: NOWT

Place the integer value of the number of coordinate points defining the water table in Columns 5 through 8, justified right. The maximum number of points that may be used is 50.

C. Number of Layers: NL

Place the integer value of the number of layers in Columns 9 through 12, justified right. The maximum number of layers that may be used is 20.

D. Number of points defining boundary layers: NOPL

Place the integer value of the number of points defining the boundary layer line in Columns 13 through 16, justified right. Each boundary layer line, therefore, must be defined by the same number of coordinate points, with a maximum of 50.

E. Number of points defining ground line: NO

Place the integer value of the number of points defining the ground line in Columns 17 through 20, justified right. The maximum number of points that may be used is 50.

F. Initial factor of safety: FS1

Since a solution for the factor of safety is not direct, an iteration process is used. This requires an initial value. Usually a number close to 1.0 will be used. If however, the user has some idea of what the factor of safety will be, the use of that number will save some computing time. Place the real value of the initial factor of safety in Columns 21 through 25.

IV. Grid Information

This program is capable of analyzing a large number of circles without repeatedly submitting the deck for each circle. From input data on this card, it is possible to set up a rectangular grid of circle center points and to analyze each point using circles from one specified radius length to another. (To work only one circle see note at end.) The rectangular grid is defined by the following card:

A. Beginning X and Y coordinate values: ISTART, JSTART (upper left corner of rectangular grid)

Place the integer values of the initial X coordinate and initial Y coordinate in Columns 1 through 10 and 11 through 20, respectively, justified right.

B. Ending X and Y coordinate values: IFIN, JFIN (lower right corner of rectangular grid)

Place the integer value of the final X coordinate and final Y coordinate in Columns 21 through 30 and 31 through 40, respectively, justified right.

C. Radius lengths: IRS, IRF

Place the integer value of the initial radius length in Columns 41 through 50 and of the final radius length in Columns 51 through 60, justified right. The initial radius must be short enough to produce circles from the closest grid point to the cross section. The final radius must be long enough to produce circles from the farthest grid point from the cross section.

D. Increments: IDEL1, IDEL2, IDEL3

Place the positive integer values of the X and Y coordinate increments in Columns 61 through 63 and Columns 64 through 66, respectively, justified right. Place the integer value of the radius length increment in Columns 67 through 69, justified right. The difference between all initial and final values must be a multiple of their respective increments. The program will not perform properly if these final values are not equal to the initial values plus some multiple of their increments.

NOTE: To analyze only one circle, the grid card must still be used. Make all final values equal to the initial values (which would be the coordinates of the circle center and the radius length). Make all increments equal to zero.

V. Minimum Radius Point: JJ

This is the point that all slip circles analyzed must enclose. It serves the purpose to eliminate those circles which do not intersect the cross section by locating this point somewhere close to the ground line (however, the point may be placed anywhere in the cross section). If the ground surface has a vertical line segment, care must be taken to insure the circle does not intersect this line segment. Select the "JJ point" just below the vertical segment and no operational difficulties will be encountered. Place the real value of the X coordinate of the "JJ point" in Columns 1 through 10 and the real value of the Y coordinate in Columns 11 through 20.

VI. Layer Properties: CO(M), PHI(M), WT(M)

Place the real value for the cohesion of a layer m kips per square foot in Columns 1 through

5, for the angle ϕ in degrees in Columns 6 through 10, and for the unit weight of the soil in kips per cubic foot in Columns 11 through 15. Make one card for each layer and put them in the same order as the boundary layer coordinate cards -- from top layer to bottom layer. If the problem is an earth fill retaining a lake, treat the water as the first layer with cohesion and angle of friction equal to zero and a unit weight equal to .0624 kips per cubic foot.

VII. Ground Line Coordinate Points: X(I), Y(I)

The ground surface must be approximated by straight line segments defined by X and Y coordinates. Place the real value of the X and Y coordinates in Columns 1 through 10 and Columns 11 to 20, respectively. Place each set of coordinates on a separate card and put the cards in order from the smallest to the largest X value. Since the critical circle must be completely within the defined ground line, it is good practice to extend the first and last X coordinates beyond the actual area to be analyzed.

VIII. Layer Boundary Line Coordinate Points: XLS(K,M), YLS(K,M)

Same as above except with respect to layer boundary lines. Put in layer lines from top of cross section to bottom.

IX. Water Table Coordinate Points: XWT(I), YWT(I)

Same as ground line coordinate points but with respect to the water table.

The /* Card

The last card of the data deck is the slash-asterisk card. Place a slash (/) in Column 1 and an asterisk (*) in Column 2.

OUTPUT OPTIONS

In INPUT INSTRUCTIONS, several references were made to normal output. This is the standard output and is printed for each circle regardless of other output options. It consists of X and Y coordinates of the circle center, the radius of the circle, the factor of safety, the area of failure, and the X and Y coordinates of the intersections of the slip circles and the ground line. Also, the water table condition is listed as either static or sloping with seepage. It also consists of all coordinates for all line segments.

All output options are controlled by two routing gimmick numbers 1 and 3. Gimmick 1 controls output of a grid system. There are three options available for a grid: 1) normal output for all circles, 2) a plot of the X coordinate and Y coordinate and the minimum factor of safety at each grid point on each card for use in a contouring program in addition to normal output. Gimmick 3 controls output for analysis of one circle. There are two options available: 1) normal printout for the circle and 2) detailed output containing all values used for computation of the Bishop factor of safety in addition to the normal printout.

SUGGESTIONS FOR USE AND LIMITATIONS OF PROGRAM

1. Use Gimmick 2 equal to 0. when analyzing only one circle. This eliminates an unnecessary bottom layer and the check to see if the circle intersects the bottom layer.

2. Use Gimmick 2 equal to 1. when using a grid to search out the critical circle. Putting in an extra bottom layer to stop the radius incrementation will usually save computing time by eliminating analysis of unnecessarily deep slip circles. Otherwise, the radius will increment until the final radius length is reached, which must be long enough to compute factors of safety for circle center points the farthest away from the slope.

3. Since all layer boundary lines must have the same number of coordinate points, there will probably be several layer lines which require additional coordinate points. It is best to place these additional points to the extreme right of the cross section. This will decrease computing time because of the manner in which the intersection of each layer line and center of slice is calculated.

4. There are many cases where one wrong number punched in the data deck will cause the program to run much longer than would normally be required and often with questionable results. If possible, a time limit should always be used. Figure 8 is a graph to aid in determining, approximately the time required to run a certain number of circles on an IBM 360 computer and an IBM 370 computer. The graph was devised from a cross section of seven layers and twelve layer line segments and, since each problem is unique, should be used only as a guide. One circle will take anywhere from about 3 to 10 seconds on the IBM 370 and 10 to 20 seconds on the IBM 360.

5. The closer the initial value of the factor of safety is to the final computed value, the less computing time is required. If the user can predict the factor of safety, time and money will be saved. Although this appears to be unrealistic, there are several instances where a prediction will be very close.

In using a grid system to analyze a cross section, a large grid with large X and Y coordinate increments is usually used in a preliminary analysis. Then a smaller grid is used to isolate a particular area of the larger grid. This smaller grid usually centers around one or more critical grid points, i.e. grid points with low factors of safety found from the initial run. On the second run, if the initial factor of safety (FS1) is given to be a number close to factors of safety found from the initial run, the number of iterations will be cut significantly. When performing an analysis on a cross section and changing some particular characteristic, such as soil parameters or an added berm, a little experience with this program and slope stability will allow the user to fairly accurately predict factors of safety for subsequent runs.

Although this seems insignificant, the number of iterations may be reduced 50 percent with a

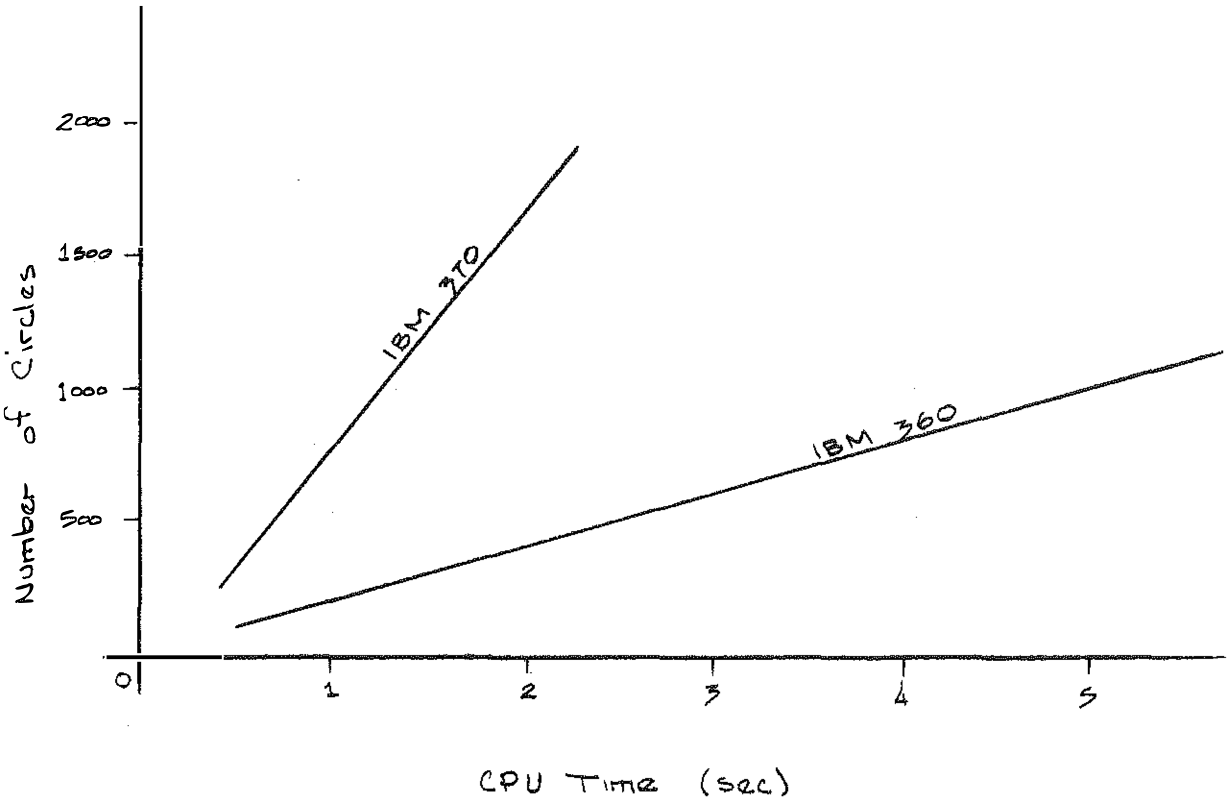


Figure 8. Approximate Computer Processing Time.

judicious choice of FS1. If the actual factor of safety is 2.000 and if FS1 is given to be 1.000 it will take from 5 to 10 iterations. If FS1 is given to be 1.800 the number of iterations is reduced to about 3 to 6.

6. It is not recommended to use Gimmick 3 equal to 0. when using the grid system. This would print detailed output for each slice and require much more time. The best time to use the detailed output option is when a critical circle has already been selected.

7. It is good practice to use X and Y coordinates as small as possible. Computer truncation introduces appreciable errors when values of the coordinates are greater than 2500. Usually, the entire cross section can be defined using coordinates from 0 to 1000, including the additional portions on both sides that must be defined when using a grid system.

8. The choice of number of slices can be used to diminish computing time. If the cross section is relatively simple, i.e. few layers and few coordinate points defining layer and ground lines, a choice of about 20 slices will be sufficient. Even with a more complex cross section, the difference between using 30 and the maximum, 50, results in errors of less than 5 percent. However, when running very few circles, a high number of slices will increase accuracy but not appreciably affect computing time.

9. It must be realized at all times that this method of analysis will only be appropriate for circular failure planes. If the cross section lends itself to a "sliding block" type failure, other methods should be used.

TROUBLESHOOTING GUIDE

There are several common errors that occur from time to time and will often terminate execution of the problem. They usually result in error messages of "Illegal decimal character," "Square root of negative argument," or "Divide by zero".

In case of an "Illegal decimal character", it nearly always indicates (assuming no keypunching error) the wrong number of data cards. The best check is to make sure the number of points on the ground line, water table, and layer boundary lines on the general information card match exactly the number of cards in the data deck that define each line. Also make sure each layer line has the same number of cards.

The other error messages come from a variety of causes. Most commonly, it is an improper coordinate due to a keypunching error or a card out of order. It also could be because of a poorly assigned minimum radius point. Make sure that the circle where the error occurred only intersects the ground line in two points. If the circle intersects the ground line as Figure 9 illustrates, errors result. In this case, change

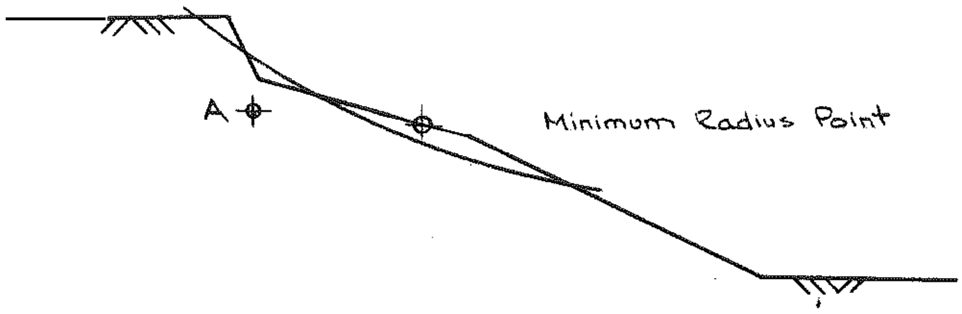


Figure 9. Choice of Minimum Radius Point.

the minimum radius to go through point A; this will eliminate from consideration circles that intersect the ground line at four points.

In the event these changes were not necessary or if the error is still present, this checklist is an aid to verify the data deck:

1. Check to see if proper I/O card is used.
2. Scan all coordinate point cards to make sure decimal points are present.
3. Check to see if the heading cards, Ginunick card, general information card, and grid information card are all in the deck.
4. Checking the general information card, simply count the number of cards in the ground line, each layer line, and water table line to make sure the numbers specified on the general information card match the number of counted cards.
5. Check all cards for proper spacing of data and decimal points where required.
6. Check all coordinate points against cross section plot.

If the grid system is being used and an error results after some circles are worked, the best procedure is to plot the circle on which the run terminated on the cross section. With this arc plotted to scale, the user might be able to quickly determine the reason for the error.

Another aid in checking a problem is to use a WATFOR or WATFIV compiler. Error messages are much easier to read and understand; in addition, such messages tell the user what line of the program was being executed at the time of the error.

CONCLUSIONS

This report results from an effort to find an effective and practical way to analyze a soil embankment with respect to slope stability. It describes a method of analysis which has been applied to the digital computer.

The simplified Bishop method and the program used for analysis do, however, have several limitations. These have been discussed and the user must be able to decide if they will appreciably affect the solution. To do this, some prior knowledge of soil mechanics and slope stability is essential. This report makes it possible to use the program without any knowledge of soil mechanics, but analysis of the solutions obtained must be justified realizing the limitations. The intent of the program is to aid the user in long, rigorous calculations, not to allow the user to blindly accept the results.

REFERENCES

1. Deen, R. C.; Scott, G. D.; and McGraw, W. W., *Stability Analysis of Earth Masses.* Kentucky Department of Highways, Division of Research. September 1966.
2. Bishop, A. W., *The Use of the Slip Circle in the Stability Analysis of Slopes.* **Geotechnique.** 1954.
3. Lambe, T. W. and Whitman, R. V., **Soil Mechanics.** John Wiley and Sons, New York. 1967.
4. Whitman, R. V. and Bailey, W. A., *Use of Computers for Slope Stability Analysis.* **Journal of the Soil Mechanics and Foundations Division,** ASCE, Vol 93, No. SM4, Proc. Paper 5327. July 1967.

APPENDIX A

PROGRAM SOURCE DECK

```
C                                                                       0010
C                                                                       0020
C      PROGRAM FOR SLOPE STABILITY ANALYSIS                             0030
C                                                                       0040
C                     USING                                             0050
C                                                                       0060
C       SIMPLIFIED BISHOP METHOD OF SLICES                              0070
C                                                                       0080
C                 DEVELOPED BY                                          0090
C                                                                       0100
C        KENTUCKY DEPARTMENT OF HIGHWAYS                                0110
C                                                                       0120
C            DIVISION OF RESEARCH                                       0130
C                                                                       0140
C         533 SOUTH LIMESTONE STREET                                    0150
C                                                                       0160
C         LEXINGTON, KENTUCKY 40508                                     0170
C                                                                       0180
C              PHONE 606-254-4475                                       0190
C                                                                       0200
      REAL MTHETA,NBAR,LOP                                              0210
      DOUBLE PRECISION ID1,ID2,ID3,ID4,IRT1,IRT2,ICO1,ICO2,IAN,IPNF1,   0220
     KIPNF2,IPNF3,IPNS1,IPNS2,IPNS3                                     0230
      DOUBLE PRECISION IDA,IDB,IDC,IDD,IDE,IDF,IDG,IDH,IDI,IDJ,IDK,IDL, 0240
     KIDM,IDN                                                           0250
      DIMENSION X(50),Y(50),XC(50),YC(50),XCOR(20),YCOR(20),FST(20,20), 0260
     KCOSI(50), XWT(50),YWT(50),YG(50),YT(50),YTT(20,50),W(1000),WT(50),0270
     KSLOPE(50),YINT(50),SINA(50),COSA(50),CO(20),TANPHI(20),LC(50),    0280
     KXLS(50,20),YLS(50,20),WW(50),WE(50),SLOP(50),YINTER(50),PHI(20),  0290
     KHWW(50)                                                           0300
      F=0.                                                              0310
C                                                                       0320
C     I.  READ NUMBER OF PROBLEMS                                       0330
C     NOTE    THE INPUT CODE MAY VARY FROM ONE COMPUTER TO ANOTHER.     0331
C             IF THE INPUT CODE IS DIFFERENT FROM THE ONE SUPPLIED IN THE 0332
C             PROGRAM, WHICH IS FOUND IN COLUMN 12 OF CARD NUMBER 0350, 0333
C             IT NEED BE CHANGED ON THIS CARD ONLY.                     0334
C                                                                       0340
      READ(1,1000)IN,IOUT,NOP                                           0350
 1000 FORMAT(3I4)                                                       0360
      DO750 INO=1,NOP                                                   0370
      L=0                                                               0380
C                                                                       0390
C     II.  READ ALL DATA FOR ONE PROBLEM                                0400
C                                                                       0410
      READ(IN,1010)ID1,ID2,ID3,ID4,IRT1,IRT2,ICO1,ICO2,IAN,IPNF1,IPNF2,I 0420
     KPNF3,IPNS1,IPNS2,IPNS3,IPADD                                      0430
 1010 FORMAT(2A6,2A6,2A5,2A6,A6,2A6,A2,2A6,A1,I1)                       0440
      IF(IPADD.EQ.0) GO TO 10                                           0450
      READ(IN,1020)IDA,IDB,IDC,IDD,IDE,IDF,IDG,IDH,IDI,IDJ,IDK,IDL,IDM,I 0460
     1DN                                                                0470
 1020 FORMAT(13A6,A2)                                                   0480
   10 READ(IN,1030) MONTH,KDAY,KYEAR                                    0490
 1030 FORMAT(3A2)                                                       0500
      READ(IN,1040) SOIL,SOIL1,POUT,EFFWT                               0510
 1040 FORMAT(4F5.0)                                                     0520
      READ(IN,1050)NSLICE,NOWT,NL,NOPL,NO,FS1                           0530
 1050 FORMAT(5I4,F5.3)                                                  0540
      WRITE(IOUT,1060)                                                  0550
 1060 FORMAT(1H1)                                                       0560
      WRITE(IOUT,1070)ID1,ID2,ID3,ID4,IRT1,IRT2,ICO1,ICO2,IPNF1,IPNF2,IP 0570
```

```
     1NF3,IPNS1,IPNS2,IPNS3,IAN
 1070 FORMAT(1H1,////,21X,2A6,2A6,//,21X,2A5,//,21X,2A6,6HCOUNTY,//,21X,
     111HPROJECT NO.,2A6,A2,/,32X,2A6,A2,//,21X,12HANALYSIS NO.,4X,A6)
      IF(IPADD.EQ.0) GO TO 20
      WRITE(IOUT,1080) IDA,IDB,IDC,IDD,IDE,IDF,IDG,IDH,IDI,IDJ,IDK,IDL,I
     1DM,IDN
 1080 FORMAT(' ',21X,13A6,A2)
   20 WRITE(IOUT,1090) MONTH,KDAY,KYEAR,NSLICE
 1090 FORMAT('0',20X,6HDATE  ,A2,1H/,A2,1H/,A2,//,21X,40HNUMBER OF SLICE
     1S USED IN THIS ANALYSIS =,I4)
      READ(IN,1100)ISTART,JSTART,IFIN,JFIN,IRS,IRF,IDEL1,IDEL2,IDEL3
   00 FORMAT(6I10,3I3)
      IF(IDEL1.EQ.0) IDEL1=1
      IF(IDEL2.EQ.0) IDEL2=1
      IF(IDEL3.EQ.0) IDEL3=1
      READ(IN,1110) XE,YE
 1110 FORMAT(2F10.3)
      READ(IN,1120)(CO(M),PHI(M),WT(M),M=1,NL)
 1120 FORMAT(3F5.0)
      READ(IN,1110)( X(I),Y(I) ,I=1,NO)
      READ(IN, 1110)(( XLS(K,M),YLS(K,M) ,K=1,NOPL),M=1,NL)
      FS=FS1
      READ(IN,1110)( XWT(II),YWT(II) ,II=1,NOWT)
C
C     III.  DETERMINATION OF STRAIGHT LINE EQUATIONS FOR ALL GROUND
C           LINE SEGMENTS
C
      N=NO-1
      DO50 I=1,N
      ABC=X(I+1)-X(I)
      IF(ABC.LE.0.) GO TO 30
      SLOPE(I)=(Y(I+1)-Y(I))/(X(I+1)-X(I))
      GO TO 40
   30 SLOPE(I)=99999.
   40 YINT(I)=Y(I)-SLOPE(I)*X(I)
   50 CONTINUE
C
C     IV.  INITIALIZING FACTORS OF SAFETY FOR ENTIRE GRID TO ZERO
C
      IF(SOIL.EQ.0.) GO TO 70
      DO 60 IX=1,20
      DO 60 IY=1,20
   60 FST(IX,IY)=0.
   70 CONTINUE
C
C     V.  SETTING UP LOOPING STRUCTURE FOR GRID
C
      IYO=JSTART+IDEL2
   80 IYO=IYO-IDEL2
      DO630 IXO=ISTART,IFIN,IDEL1
      JJJ=0
      IXX=((IXO-ISTART)/IDEL1)+1
      IYY=((JSTART-IYO)/IDEL2)+1
      DO610 IRO=IRS,IRF,IDEL3
C
C     VI.  INITIALIZING SLICE VARIABLES TO ZERO
C
      DO 90 J=1,NSLICE
      DO 90 M=1,NL
   90 YTT(M,J)=0.
      DO 100 J=1,NSLICE
```

```
      YG(J)=0.
      YT(J)=0.
      LC(J)=0.
      W(J)=0.
      WE(J)=0.
  100 WW(J)=0.
C
C     VII.  CALCULATION OF THE INTERSECTION OF GROUND LINE AND FAILURE
C                CIRCLE
C
      XO=IXO
      YO=IYO
      RO=IRO
      Z=SQRT((XO-XE)**2+(YO-YE)**2)
      IF(RO.LE.Z) GO TO 610
      XA=0.
      XB=0.
      YA=0.
      YB=0.
      DO210 I=1,N
      A=1.+SLOPE(I)**2
      B=SLOPE(I)*(YINT(I)-YO)-XO
      C=(YINT(I)-YO)**2+XO**2-RO**2
      TEST=B**2-A*C
      IF(TEST.GT.0.)GO TO 110
      GO TO 210
  110 XP=(-B+SQRT(TEST))/A
      XM=(-B-SQRT(TEST))/A
      IF(ABS(X(I)-X(I+1)).LT.3.) GO TO 150
      IF(XA.GT.0.) GO TO 130
      IF((XM.GE.(X(I)-1.)).AND.(XM.LE.(X(I+1)+1. ))) GO TO 120
      GO TO 130
  120 XA=XM
      YA=SLOPE(I)*XA+YINT(I)
  130 IF(XB.GT.0.) GO TO 220
      IF((XP.GE.(X(I)-1.)).AND.(XP.LE.(X(I+1)+1.))) GO TO 140
      GO TO 210
  140 XB=XP
      YB=SLOPE(I)*XB+YINT(I)
      GO TO 220
  150 YP=SLOPE(I)*XP+YINT(I)
      YM=SLOPE(I)*XM+YINT(I)
      IF(Y(I).LE.Y(I+1)) GO TO 160
      HOP=Y(I)
      LOP=Y(I+1)
      GO TO 170
  160 HOP=Y(I+1)
      LOP=Y(I)
  170 IF(XA.GT.0) GO TO 190
      IF((XM.GE.(X(I)-1.)).AND.(XM.LE.(X(I+1)+1.)).AND.(YM.GE.(LOP-1.)).
     1AND.(YM.LE.(HOP+1.))) GO TO 180
      GO TO 190
  180 XA=XM
      YA=YM
  190 IF(XB.GT.0) GO TO 220
      IF((XP.GE.(X(I)-1.)).AND.(XP.LE.(X(I+1)+1.)).AND.(YP.GE.(LOP-1.)).
     1AND.(YP.LE.(HOP+1.))) GO TO 200
      GO TO 210
  200 XB=XP
      YB=YP
  210 CONTINUE
```

```
C
C     VIII.  CALCULATION OF SLICE WIDTH
C
  220 SLICE=NSLICE
      BB=(XB-XA)/SLICE
C
C      IX.  CALCULATION OF COORDINATES OFINTERSECTION OF FAILURE CIRCLE
C           AND CENTER OF EACH SLICE
C
      XC(1)=XA+BB/2.
      DO 230 J=2,NSLICE
  230 XC(J)=XC(J-1)+BB
      DO 240 J=1,NSLICE
      YC(J)=YO-SQRT(RO**2-(XC(J)-XO)**2)
C
C     X.  CALCULATION OF THETA ANGLE FOR BISHOP'S EQUATION
C
      SINA(J)=(XO-XC(J))/RO
  240 COSA(J)=(YO-YC(J))/RO
C
C     XI.  CHECK TO SEE IF CIRCLE LIES IN BOTTOM LAYER
C
      IF(SOIL1.EQ.0.) GO TO 270
      NL1=NL-1
      NL2=NOPL-1
      DO 260 J=1,NSLICE
      DO 260 I=1,NL2
      IF(XC(J).GE.XLS(I,NL1).AND.XC(J).LE.XLS(I+1,NL1)) GO TO 250
      GO TO 260
  250 YR=((YLS(I+1,NL1)-YLS(I,NL1))/(XLS(I+1,NL1)-XLS(I,NL1)))*(XC(J)-XL
     1S(I,NL1))+YLS(I,NL1)
      IF(YC(J).LT.YR) GO TO 620
  260 CONTINUE
  270 DO300 J=1,NSLICE
      DO 280 I=1,N
      IF(XC(J).GE.X(I).AND.XC(J).LE.X(I+1))GO TO 290
  280 CONTINUE
C
C     XII.  DETERMINATION OF Y COORDINATE OF CENTER OF EACH SLICE AT
C           GROUND LINE
C
  290 YG(J)=SLOPE(I)*XC(J)+YINT(I)
  300 CONTINUE
C
C     XIII.  DETERMINATION OF Y COORDINATE OF CENTER OF EACH SLICE AT
C            WATER TABLE LINE
C
      NN=NOWT-1
      DO340 J=1,NSLICE
      DO310 II=1,NN
      IF(XC(J).GT.XWT(II).AND.XC(J).LE.XWT(II+1))GO TO 330
      IF(XC(J).GT.XWT(NOWT))GO TO 320
  310 CONTINUE
  320 YT(J)=YG(J)
      GO TO 340
  330 YT(J)=((YWT(II+1)-YWT(II))/(XWT(II+1)-XWT(II)))*(XC(J)-XWT(II))+
     1YWT(II)
  340 CONTINUE
C
C     XIV.  DETERMINATION OF Y COORDINATE OF CENTER OF EACH SLICE AT
C           ALL LAYER BOUNDARY LINES
```

```
C
      NLPL=NOPL-1
      DO 360 J=1,NSLICE
      DO 360 M=1,NL
      DO 360 K=1,NLPL
      IF(XC(J).GT.XLS(K,M).AND.XC(J).LE.XLS(K+1,M))GO TO 350
      GO TO 360
  350 YTT(M,J)=((YLS(K+1,M)-YLS(K,M))/(XLS(K+1,M)-XLS(K,M)))*(XC(J)-XLS
     1(K,M))+YLS(K,M)
  360 CONTINUE
      EFF=0.
      TOP=0.
      BOT=0.
      AA=0.
C
C     XV.  DETERMINATION OF STRAIGHT LINE EQUATIONS FOR ALL WATER
C          TABLE LINE SEGMENTS
C
      NNN=NOWT-1
      DO 370 I=1,NNN
      SLOP(I)=(YWT(I+1)-YWT(I))/(XWT(I+1)-XWT(I))
      YINTER(I)=YWT(I)-SLOP(I)*XWT(I)
      IF(YWT(I).NE.YWT(I+1)) GO TO 370
      SLOP(I)=0.
      YINTER(I)=YWT(I)
  370 CONTINUE
C
C     XVI.  CALCULATION OF THE COSINE OF THE ANGLE EACH WATER TABLE
C           LINE SEGMENT MAKES WITH THE HORIZONTAL (FOR EACH SLICE)
C
      DO 390 J=1,NSLICE
      DO 380 I=1,NNN
      IF(XC(J).GT.XWT(I+1)) GO TO 380
      COSI(J)=(XWT(I+1)-XWT(I))/SQRT((YWT(I+1)-YWT(I))**2+(XWT(I+1)-XWT(
     1I))**2)
      GO TO 390
  380 CONTINUE
  390 CONTINUE
C
C     XVII.  DETERMINATION OF TOTAL WEIGHT OF EACH SLICE
C
      DO 500 J=1,NSLICE
      DO 400 M=1,NL
      A1=0.
      A2=0.
      A3=0.
      A4=0.
      IF(YTT(M,J).NE.0.) GO TO 410
  400 CONTINUE
  410 MT=M
      SL=YG(J)-YC(J)
      DO 420 M=1,NL
      IF(YTT(M,J).EQ.0.) GO TO 420
      SL1=YG(J)-YTT(M,J)
      IF(SL1.GE.SL)GO TO 430
  420 CONTINUE
  430 IF(M.EQ.1)GO TO 460
      IF(YTT(M-1,J).EQ.0.) GO TO 460
      MB=M-1
      M=MT
      A1=BB*(YG(J)-YTT(M,J))+A1
```

```
      W(J)=W(J)+BB*(YG(J)-YTT(M,J))*WT(M)
      SL2=YTT(M,J)-YTT(M+1,J)
      SL3=YTT(M,J)-YC(J)
      IF(SL2.GE.SL3)GO TO 450
      MB=MB-1
      DO 440 M=MT,MB
      A2=BB*(YTT(M,J)-YTT(M+1,J))+A2
  440 W(J)=W(J)+BB*(YTT(M,J)-YTT(M+1,J))*WT(M+1)
      M=MB+1
  450 W(J)=W(J)+BB*(YTT(M,J)-YC(J))*WT(M+1)
      A3=BB*(YTT(M,J)-YC(J))+A3
      LC(J)=M+1
      GO TO 470
  460 W(J)=W(J)+BB*(YG(J)-YC(J))*WT(M)
      A4=BB*(YG(J)-YC(J))+A4
      LC(J)=M
  470 IF(YT(J).LT.YC(J))GO TO 480
      WW(J)=(YT(J)-YC(J))*.0624*BB
      IF(EFFWT.EQ.1.) WW(J)=((YT(J)-YC(J))*COSI(J)**2)*.0624*BB
C
C     XVIII.  DETERMINATION OF EFFECTIVE WEIGHT OF EACH SLICE
C
      WE(J)=W(J)-WW(J)
      IF(WE(J).LE.0.) WE(J)=0.
      GO TO 490
  480 WE(J)=W(J)
C
C     IXX.  DETERMINATION OF TOTAL AREA OF FAILURE
C
  490 AA=AA+A1+A2+A3+A4
C
C     XX.  DETERMINATION OF DRIVING WEIGHT OF EACH SLICE
C
      HWW(J)=0.
      IF(YWT(NOWT).NE.YWT(NOWT-1)) GO TO 500
      IF(YC(J).LT.YWT(NOWT).AND.WE(J).GE.0.) HWW(J)=(YWT(NOWT)-YC(J))*BB
     1*.0624
  500 CONTINUE
C
C     XXI.  CALCULATION OF FACTOR OF SAFETY
C
      MTH=0.
  510 DO 520 J=1,NSLICE
      EFF=WE(J)
      M=LC(J)
      ZZ=PHI(M)*3.14159/180.
      MTHETA=COSA(J)+(SINA(J)*TAN(ZZ))/FS
      IF(MTHETA.LT..2) MTH=1.
      TOP=TOP+(CO(M)*BB+EFF*TAN(ZZ))/MTHETA
  520 BOT=BOT+(W(J)-HWW(J))*SINA(J)
      F=TOP/BOT
      IF(ABS(F-FS).LT..005) GO TO 530
      BOT=0.
      TOP=0.
      MTH=0.
      FS=F
      IF(FS.GE.10.) GO TO 600
      GO TO 510
  530 FS=F
      IF(POUT.EQ.1.) GO TO 560
      TOP=0.
```

```
      BOT=0.
      WRITE(IOUT,1130) FS
 1130 FORMAT('1',10X,3HFS=,F6.3)
      WRITE(IOUT,1140) BB
 1140 FORMAT(' ',10X,21HWIDTH OF EACH SLICE =F8.4)
      WRITE(IOUT,1150)
 1150 FORMAT('0',2X,1HJ,12X,4HW(J),6X,5HWW(J),6X,5HWE(J),4X,6HMTHETA,6X,
     14HNBAR,5X,8HCO(M)*BB,4X,5HSHEAR,6X,5HSHEAR,6X,6HCOSINE,6X,4HSINE,/
     2,1X,5HSLICE,9X,6HWEIGHT,7X,1HU,38X,8HCOHESION,3X,8HSTRENGTH,4X,6HF
     3ORCE ,5X,5HTHETA,7X,5HTHETA,/,15X,6H(KIPS),5X,6H(KIPS),5X,6H(KIPS)
     4,14X,6H(KIPS),5X,6H(KIPS),5X,6H(KIPS),6X,6H(KIPS),///)
      MTH=0.
      DO 540 J=1,NSLICE
      EFF=WE(J)
      M=LC(J)
      ZZ=PHI(M)*3.14159/180.
      MTHETA=COSA(J)+(SINA(J)*TAN(ZZ))/FS
      IF(MTHETA.LT..2) MTH=1.
      NBAR=(WE(J)-(1/FS)*CO(M)*BB*SINA(J)/COSA(J))/MTHETA
      TOPS=(CO(M)*BB+EFF*TAN(ZZ))/MTHETA
      BOTS=(W(J)-HWW(J))*SINA(J)
      COHES=CO(M)*BB
      WRITE(IOUT,1160)J,W(J),WW(J),WE(J),MTHETA,NBAR,COHES,TOPS,BOTS,COS
     1A(J),SINA(J)
 1160 FORMAT(' ',2X,I2,7X,F9.3,2X,F9.3,2X,F9.3,3X,F6.4,2X,F9.3,2X,F9.3,2
     1X,F9.3,2X,F9.3,4X,F6.4,6X,F6.4)
      TOP=TOP+(CO(M)*BB+EFF*TAN(PHI(M)*3.14159/180.))/MTHETA
  540 BOT=BOT+(W(J)-HWW(J))*SINA(J)
      WRITE(IOUT,1170) TOP
 1170 FORMAT(' ',68X,4HSUM=,F12.3)
      WRITE(IOUT,1180) BOT
 1180 FORMAT(' ',79X,4HSUM=,F12.3)
      ARCLN=0.
      DO 550 J=1,NSLICE
  550 ARCLN=ARCLN+BB/COSA(J)
      AVESS=BOT/ARCLN
      WRITE(IOUT,1190) AVESS
 1190 FORMAT('0',22HAVERAGE SHEAR STRESS =,F9.3,5H(KSI))
  560 FS=F
C
C     XXII.  ALL DIFFERENT OPTIONS FOR FORMATTED OUTPUT
C
      L=L+1
      IF(L.GT.1) GO TO 580
      IF(YWT(1).NE.YWT(NOWT)) GO TO 570
      WRITE(IOUT,1200)
 1200 FORMAT(1H1,54X,24HSLOPE STABILITY ANALYSIS,//,63X,5HUSING,//,54X,2
     14HSIMPLIFIED BISHOP METHOD,///,54X,28HTHIS ANALYSIS WAS MADE USING
     2,/,54X,29HSTATIC WATER TABLE CONDITIONS,////,11X,57HCOORDINATES OF
     3 THE   RADIUS OF   FACTOR OF          AREA OF,13X,30HCOORDINATES OF
     4INTERSECTION OF,/,10X,58HCENTER OF THE CIRCLE THE CIRCLE   SAFETY
     5            FAILURE,13X,31HFAILURE CIRCLE WITH GROUND LINE,//,12X,27
     6HX(FEET)   Y(FEET)    (FEET),21X,9H(SQ.FEET),10X,9HX-INITIAL,3X,9H
     7Y-INITIAL,5X,7HX-FINAL,5X,7HY-FINAL,//)
      GO TO 580
  570 WRITE(IOUT,1210)
 1210 FORMAT(1H1,54X,24HSLOPE STABILITY ANALYSIS,//,63X,5HUSING,//,54X,2
     14HSIMPLIFIED BISHOP METHOD,///,54X,28HTHIS ANALYSIS WAS MADE USING
     2,/,47X,43HSLOPING WATER TABLE CONDITIONS WITH SEEPAGE,////,11X,57H
     3COORDINATES OF THE   RADIUS OF   FACTOR OF          AREA OF,13X,30HC
     4OORDINATES OF INTERSECTION OF,/,10X,58HCENTER OF THE CIRCLE THE CI
```

```
     5RCLE   SAFETY             FAILURE,13X,31HFAILURE CIRCLE WITH GROUND
     6LINE,//,12X,27HX(FEET)   Y(FEET)    (FEET),21X,9H(SQ.FEET),10X,9HX
     7-INITIAL,3X,9HY-INITIAL,5X,7HX-FINAL,5X,7HY-FINAL,//)
  580 CONTINUE
      WRITE(IOUT,1220)XO,YO,RO,FS,AA,XA,YA,XB,YB
 1220 FORMAT(10X,3F10.3,F11.3,5X,F12.2,10X,F10.3,2X,F10.3,2X,F10.3,2X,F1
     10.3)
      IF(MTH.EQ.1.) WRITE(IOUT,1230)
 1230 FORMAT(' ',55H**********WARNING   DEEP FAILURE   MTHETA LESS THAN
     10.2)
      IF(IXO.EQ.IFIN.AND.IYO.EQ.JFIN.AND.IRO.EQ.IRF)GO TO 730
      IF(SOIL.EQ.0.) GO TO 590
      JJJ=JJJ+1
      IF(JJJ.EQ.1) FST(IXX,IYY)=FS
      IF(FS.LE.FST(IXX,IYY)) FST(IXX,IYY)=FS
  590 FS=FS1
C
C      END  CALCULATIONS TO DETERMINE THE FACTOR OF SAFETY.
C     NOW INCREMENT RADIUS, OR MOVE TO THE NEXT GRID POINT AND MINUM RAD
C
  600 CONTINUE
      IF(IRO.EQ.IRF) GO TO 620
  610 CONTINUE
  620 WRITE(IOUT,1240)
 1240 FORMAT(/)
  630 CONTINUE
      IF(IYO.EQ.JFIN) GO TO 640
      GO TO 80
  640 IF(SOIL.EQ.1.) GO TO 680
      IF(SOIL.EQ.2.) GO TO 650
      GO TO 730
  650 WRITE(7,1250) ID1,ID2,ID3,ID4,IRT1,IRT2,ICO1,ICO2,IAN,IPNF1,IPNF2,
     KIPNF3,IPNS1,IPNS2,IPNS3
 1250 FORMAT(4A6,2A5,2A6,A6,2A6,A2,2A6,A2)
      IYO=JSTART+IDEL2
  660 IYO=IYO-IDEL2
      IYY=((JSTART-IYO)/IDEL2)+1
      DO 670 IXO=ISTART,IFIN,IDEL1
      IXX=((IXO-ISTART)/IDEL1)+1
      IF(FST(IXX,IYY).EQ.0.) GO TO 670
      XO=IXO
      YO=IYO
      WRITE(7,1260) XO,YO,FST(IXX,IYY)
 1260 FORMAT(3F10.3)
  670 CONTINUE
      IF(IYO.EQ.JFIN) GO TO 730
      GO TO 660
  680 IXO=ISTART-IDEL1
      IYO=JSTART+IDEL2
      DO 690 I=1,20
      XCOR(I)=0.
  690 YCOR(I)=0.
      DO 700 I=1,20
      IF(IXO.EQ.IFIN.AND.IYO.EQ.JFIN) GO TO 710
      IXO=IXO+IDEL1
      IYO=IYO-IDEL2
      XO=IXO
      YO=IYO
      XCOR(I)=XO
  700 YCOR(I)=YO
  710 WRITE(IOUT,1270)
```

```
1270 FORMAT('1',50X,34HFAILURE CIRCLE CENTER GRID SYSTEM,//61X,12HNOT T
    KO SCALE,/,60X,15HIGNORE ALL 0.'S,///)
     WRITE(IOUT,1280)(XCOR(I),I=1,20)
1280 FORMAT(' ',10X,20F6.1)
     DO 720 I=1,20
     WRITE(IOUT,1290) YCOR(I),(FST(J,I),J=1,20)
1290 FORMAT(' ',/,1X,F6.1,3X,20F6.3)
 720 CONTINUE
 730 WRITE(IOUT,1300)
1300 FORMAT(1H1,10X,34HCOORDINATES OF ORIGINAL GROUNDLINE,//,16X,7HX(FE
    1ET),15X,7HY(FEET),//)
     WRITE(IOUT,1310)(X(I),Y(I),I=1,NO)
1310 FORMAT( 9X,F15.3,7X,F15.3)
     WRITE(IOUT,1320)
1320 FORMAT(1H1,15X,25HCOORDINATES OF WATERTABLE,//,16X,7HX(FEET),15X,7
    1HY(FEET),//)
     WRITE(IOUT,1310)(XWT(II),YWT(II),II=1,NOWT)
     DO 740 M=1,NL
     WRITE(IOUT,1330)M,CO(M),PHI(M),WT(M)
1330 FORMAT(1H1,10X,24HLAYER NUMBER            ,I5,///,11X,20HCOHESION(
    1KIPS/SQFT)    ,F15.5,/,11X,13HPHI (DEGREES),7X,F15.5,/,11X,22HUNIT
    2WEIGHT(KIPS/CUFT),F13.5)
     WRITE(IOUT,1340)
1340 FORMAT(/////,11X,34HCOORDINATES OF LAYER BOUNDARY LINE,//,13X,7HX(
    1FEET),15X,7HY(FEET),//)
 740 WRITE(IOUT,1350)(XLS(K,M),YLS(K,M),K=1,NOPL)
1350 FORMAT(6X,F15.3,7X,F15.3)
 750 CONTINUE
     CALL EXIT
     END
```

APPENDIX B

DETAILED EXPLANATION OF PROGRAM PROCESS

DETAILED EXPLANATION OF PROGRAM PROCESS

I. Read number of problems to be worked and set up looping structure to go through entire program as many times as there are problems.

II. Read all data for one problem.

III. Determine straight-line equations for all ground-line segments.

A. To determine the slope, the difference between the Y coordinates of two adjacent ground-line points is divided by the difference between the X coordinates of the same two points. This value for the slope is stored in the variable SLOPE(I), where the subscript corresponds to the number of the ground-line segment counted from left to right. In case of an overhanging ledge, the line segment is given a slope of 99999.

B. To determine the y-intercept, the slope of the line segment as determined above and the X and Y coordinate of the left-most ground-line point for the particular line segment is used in the general equation of a straight line:

$$b = y - mx$$

where x and y are the coordinates and b is the y-intercept. This number is stored in the variable YINT(I).

IV. Initialize factors of safety for entire grid to zero.

If normal printout, i.e. only grid coordinates, radius length, factor of safety, area of failure and coordinates of intersections of slip surface and ground line is all that is required, this step will be bypassed. This happens when the routing gimmick SOIL is equal to 0. A 20 x 20 array is established to store values for the least factor of safety at each grid point. This step simply stores a 0.0 in all 400 locations of this array in the form of the variable FST(IX,IY) where IX and IY go from one to twenty.

V. Set up looping structure for grid.

The looping structure for the grid is done so that the radius, X coordinate, and Y coordinate begin at their initial values. The order of operations in these nested loops is as follows:

A. The radius begins at its initial value (IRS) and goes to its final value (IRF) by specified increments (IDEL3). The program attempts to calculate a factor of safety for each radius length.

B. The X coordinate begins at its initial value (ISTART) and goes to its final value (IFIN) by specified increments (IDEL1). For each X coordinate value, all radius lengths are used.

C. The Y coordinate begins at its initial value (ISTART) and goes to its final value (IFIN) by specified increments (IDEL2). For each Y coordinate value, all X coordinates are used.

VI. Initialize slice variables to zero.

Subscripted variables which depend on the slice and which need to be initialized to zero are the following (the variable subscript J is used throughout the program to designate the slice):

YTT(M,J) -- Y coordinate of the intersection of the center of the Jth slice and the Mth layer (nested loops where J goes from 1 to NSLICE and M goes from 1 to NL are used for this operation).

YG(J) -- Y coordinate of the intersection of the ground line and center of the Jth slice.

YT(J) -- Y coordinate of the intersection of the water table and center of the Jth slice.

LC(J) -- Layer counter for the Jth slice.

W(J) -- Total weight of the Jth slice.

WE(J) -- Effective weight of the Jth slice.

WW(J) -- Weight of water in the Jth slice.

VII. Calculate the intersections of ground line and slip circle.

This is done by treating each ground-line segment as a line of infinite length. The program uses one ground-line segment at a time and solves the equation describing this line and the equation of the particular circle used for the analysis. It then checks to see if either of these two intersection points lies within the bounds of the ground-line segment given by the two ground-line points. If not, the next ground-line segment is used. The above procedure is repeated until intersections of the ground line and slip circle are found.

To determine the intersection of a line and a circle, simultaneous solution of two equations is required:

$y = mx + b$ Equation of straight line B1

$(x - h)^2 + (y - k)^2 = r^2$ Equation of circle B2

Substituting for y in Equation B2 yields

$$(x - h)^2 + (mx + b - k)^2 = r^2.$$

Carrying out the squaring operations and regrouping terms:

$$(x^2 - 2xh + h^2) + (m^2x^2 + 2bmx - 2kmx + b^2 - 2bk + k^2) = r^2$$
$$x^2 + m^2x^2 + 2bmx - 2kmx - 2xh + b^2 - 2bk + k^2 + h^2 = r^2$$
$$x^2(1 + m^2) + m\left\{2[m(b-k) - h]\right\} + [h^2 + (b-k)^2 - r^2] = 0$$

Using the quadratic equation, values of x can be determined as

$$x = -[m(b-k) - h] \pm \sqrt{[m(b-k) - h]^2 - (1 + m^2)\,[h^2 + (b - k)^2 - r^2]}$$
$$/(1 + m^2). \qquad \text{B3}$$

The program uses a looping structure to solve this equation and check all ground-line segments. Using simplified notation,

$A = (1 + m^2)$,
$B = m(b-k) - h$,
$C = (b - k)^2 + h^2 - r^2$, and
$\text{TEST} = B^2 - AC$,

the program finds two solutions for x as

$$XP = (-B + \sqrt{TEST})/A.$$

and

$$XM = (-B - \sqrt{TEST})/A.$$

(If TEST is negative, the program ignores this line segment and increments to the next.) The program then checks XP and XM to determine whether or not either one lies on the ground-line segment.

In the general case, a simple check to see if XP or XM lie between an adjacent pair of X coordinate values of two ground-line points would be sufficient. Figure 10 shows how

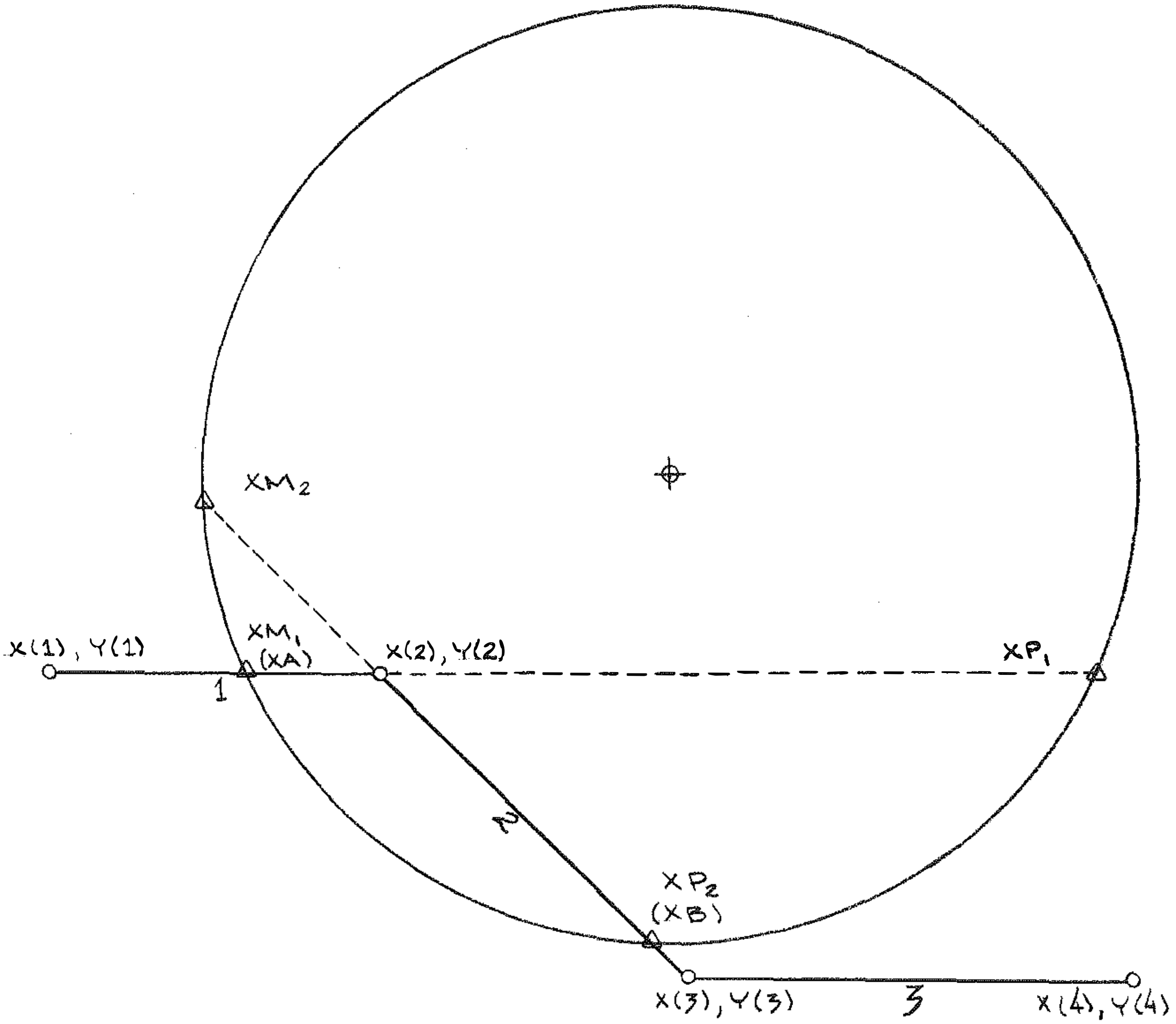


Figure 10. Intersection of Slip Circle and Ground Line.

the intersections of the slip circle and each ground-line segment are used to calculate the intersection of the circle and actual ground-line. In this case, XM_1 and XP_1 are solutions to Equation B3. Checking XM_1, it is found that it lies between X(1) and X(2) and therefore is one intersection point of the ground line and slip circle. It now takes on the variable name XA. Checking XP_1, it is found that it is greater than X(2); therefore, the next ground line segment is tried. By similar use of Equation B3, XP_2 is found as the other intersection point and is renamed XB. Then by substituting XA and XB into their respecting straight-line equations, values for the Y coordinates, YA and YB, can be determined.

However, there are two difficulties and a more sophisticated checking procedure is required.

Case 1. Truncation errors in the computer.

Truncation errors become significant when the circle passes very close to a ground-line coordinate point as in Figure 11. It is possible for the computer to never find an intersection point. When working through the looping structure, XB might never be assigned a value. The program will have no problem with line *l*, but line m and line n might halt calculations. If the circle lies very close to coordinate point n (within a foot), it is possible that XP might be calculated to be greater than X(n) when solving Equation B3 for line m and less than X(n) when solving Equation B3 for line n. Figure 12 illustrates this case. This problem was solved by putting a one foot extra boundary on the coordinate points; i.e. if the circle passes within one foot of the coordinate point, the solution is accepted. A one-foot maximum error will not appreciably increase an error in calculations.

Case 2. Vertical or nearly vertical ground-line segments.

Without a more comprehensive check, any vertical line segment will produce errors. For example, consider a bridge abutment. From Figure 13, it is observed that solving Equation B3 for line m gives XP_m less than X(n) plus the one foot extra boundary. This would imply that the slip circle stops right under the vertical line segment. To overcome this problem, an additional condition is implemented when the difference between the X coordinate values of two adjacent ground-line points is less than three feet (arbitrarily selected). The program then uses the value of XP and XM in the straight-line equation of that particular line segment and calculates the respective values of the Y coordinates, YP and YM. It then checks XP, YP and(or) XM, YM against the X and Y ground-line coordinates (within the one-foot extra boundary).

VIII. Calculate slice width.

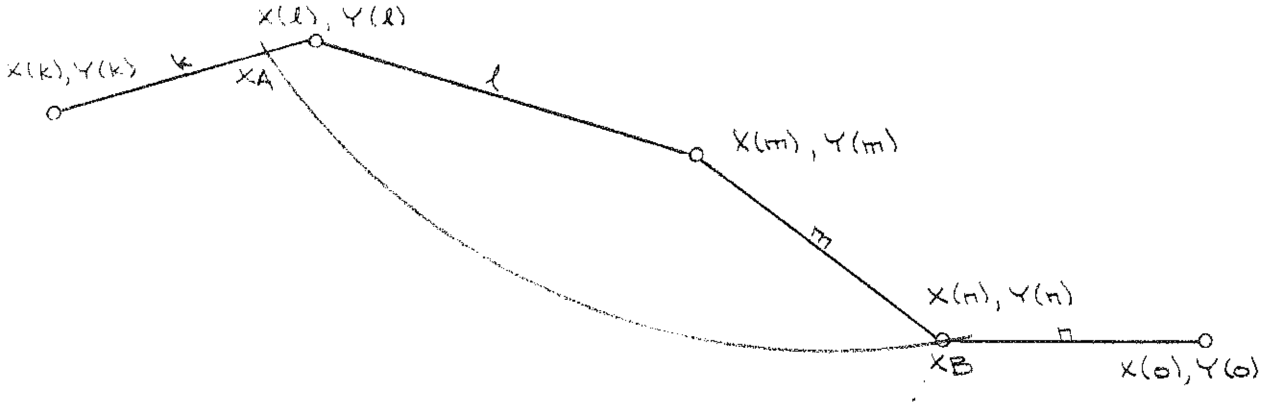
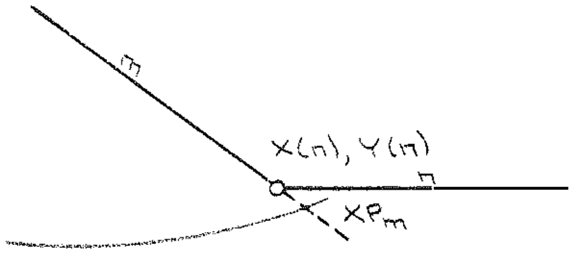
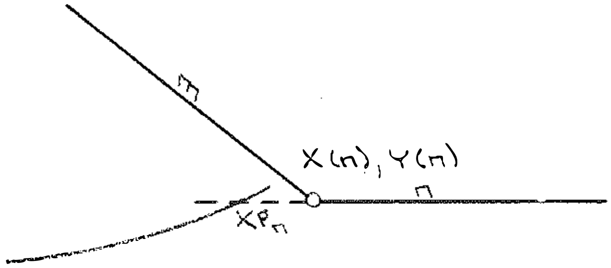


Figure 11. Source of Possible Truncation Error in Computer.



Calculation of XP for Line m



Calculation of XP for Line n

Figure 12. Exaggerated Case of Truncation Error.

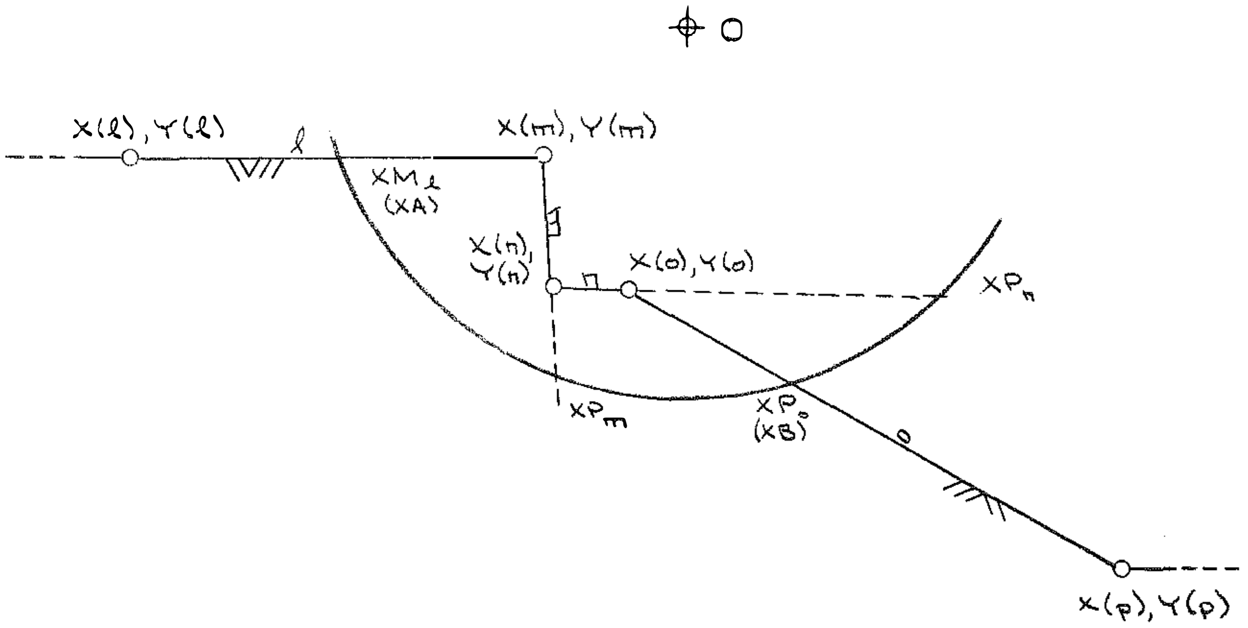


Figure 13. Vertical Line Segment Error.

The width of each slice (BB) is found by determining the horizontal distance between the two points where the circle intersects the ground line and dividing by the number of slices:

NSLICE = SLICE

BB = (XB - XA)/SLICE.

NSLICE becomes SLICE to make the number of slices a real number.

IX. Calculate coordinates of intersection of slip circle and center of each slice (see Figure 14).

A. X coordinate

The X coordinate of the center of the first slice, XC(1), is one-half the slice width plus the value for XA. All others are determined by adding the width of slice BB to the previous XC(J). A looping structure is used to perform this operation.

B. Y coordinate

Another looping structure is used to find the Y coordinate, YC(J). Since the X coordinate is now known, the equation for the slip circle will yield the Y coordinate:

$$(x - h)^2 + (y - k)^2 = r^2$$

$$(y - k) = \pm \sqrt{r^2 - (x - h)^2}$$

$$y = k \pm \sqrt{r^2 - (x - h)^2}$$

NOTE: Since the only interest is in the least value for y, the equation becomes

$$y = k - \sqrt{r^2 - (x - n)^2}$$

where (h, k) = coordinates of circle center,

y = YC(J),

x = XC(J), and

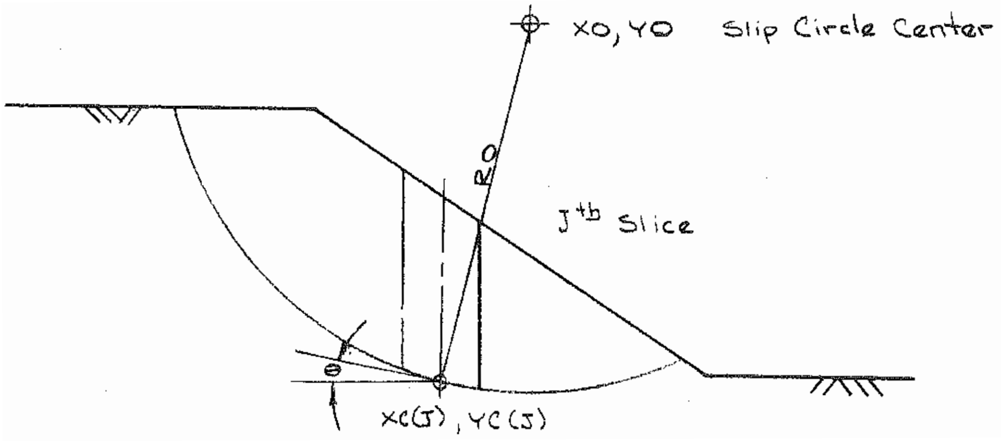
r = radius of circle.

X. Calculate theta angle for Bishop's equation.

The theta angle (θ) is the angle between the tangent to the slip circle at the center of each slice and the horizontal. Figure 15 illustrates the computation.

XI. Check to see if circle lies in bottom layer.

This check is made if Ginunick 2 (SOIL1) equals 1. Basically, this is done by determining



$$\mathrm{SINA(J)} = \sin\theta_J = [\mathrm{XO} - \mathrm{XC(J)}]\,/\,\mathrm{RO}$$

$$\mathrm{COSA(J)} = \cos\theta_J = [\mathrm{YO} - \mathrm{YC(J)}]\,/\,\mathrm{RO}$$

Figure 14. Calculation of XC(J) and YC(J).

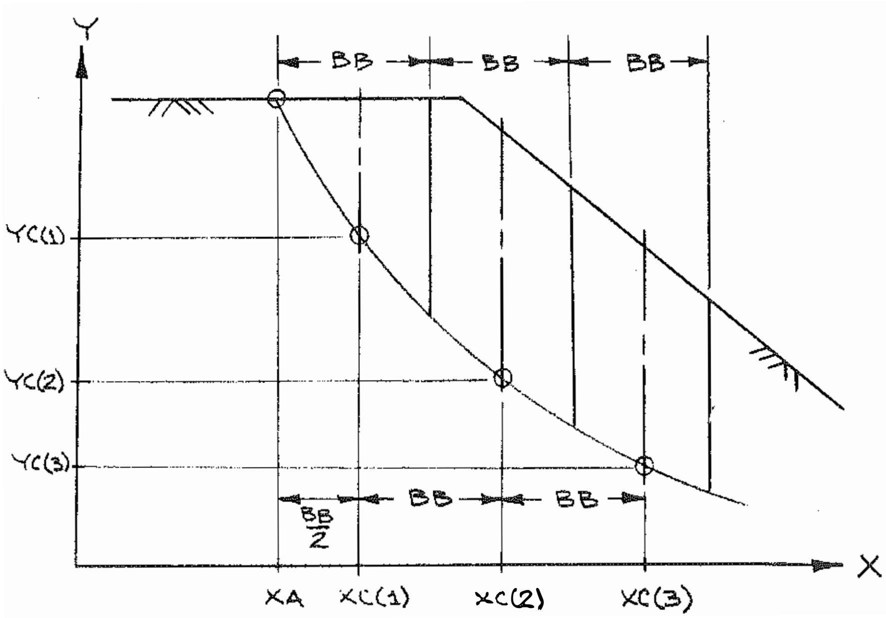


Figure 15. Computation of the Theta Angle.

the Y coordinate of the intersection of the center of each slice and the top of the bottom layer. This is then compared with YC(J). If this Y coordinate (YR) is greater than YC(J), the circle lies in the bottom layer and the rest of the program is bypassed and a new grid point is tried. Figure 16 shows how YR is calculated. XLS(K, M) and YLS(K, M) are the X and Y coordinates of the Kth point on the Mth layer line. NL1 is the layer line directly over the bottom layer.

XII. Determine Y coordinate of center of each slice at ground line.

This procedure is done in two steps; first to find out which ground-line segment intersects each center of slice XC(J) and second to use the straight-line equation and XC(J) to find the Y coordinate of the center of slice at the ground line, YG(J). Nested loops are used for this two-step operation. The inner loop finds the ground-line segment that intersects the center of the slice by comparing adjacent ground-line X coordinates to determine if a particular XC(J) lies between them. Once this is found, the outer loop determines the YG(J) for all slices as follows:

$$YG(J) = SLOPE(I) \quad XC(J) + YINT(I)$$

where I denotes the number of the ground-line segment which was found in the inner loop and J denotes the slice.

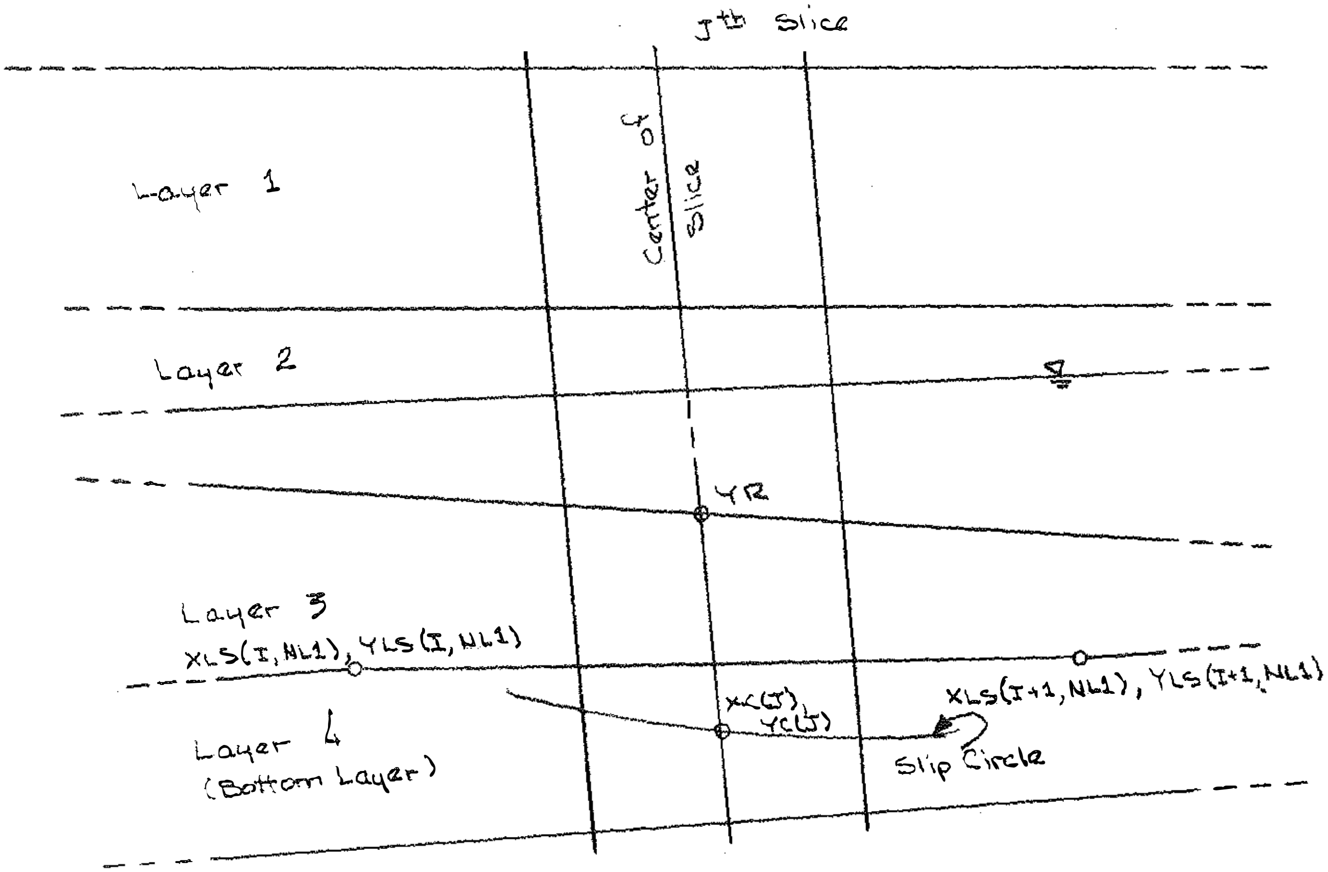
XIII. Determine Y coordinate of center of each slice at water-table line.

To find this coordinate, YT(J), a procedure similar to the above method is used; that is, two nested loops, one to find which water-table line segment intersects the center of the slice and the other to solve its straight-line equation for XC(J). An additional operation of putting YT(J) equal to YG(J) when any XC(J) is greater than the X coordinate of the last water-table line segment is used.

XIV. Determine Y coordinate of center of each slice at all layer boundary lines.

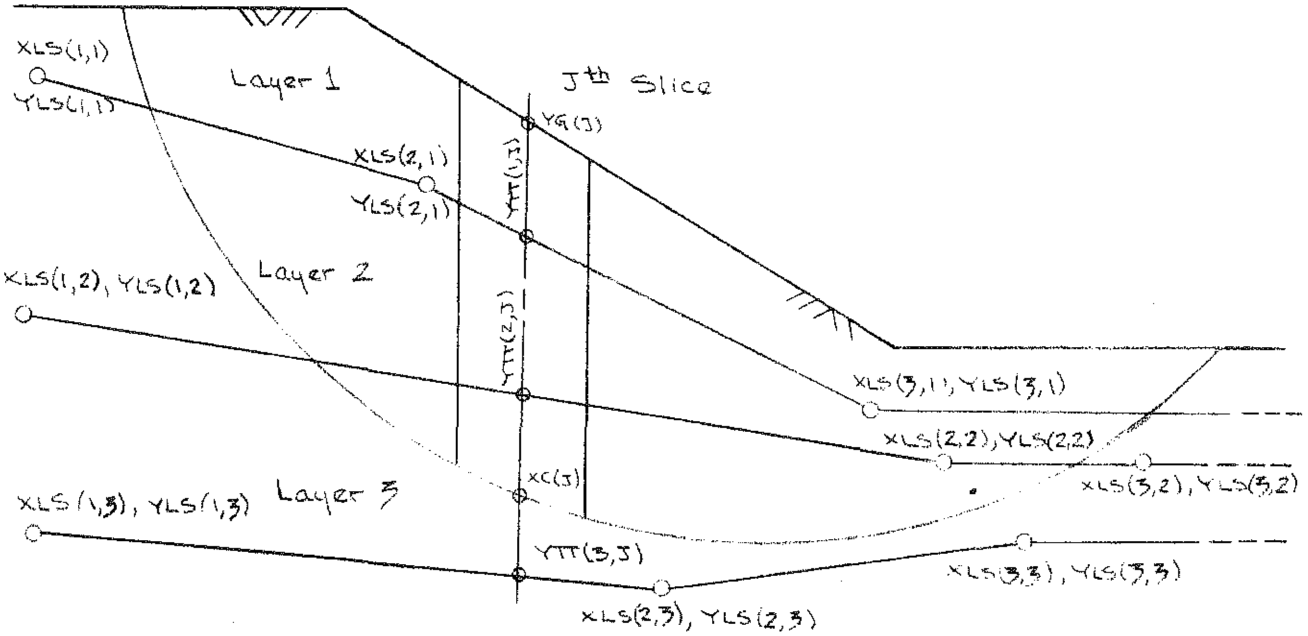
This procedure uses three nested loops. The inner loop finds the layer boundary line segment intersecting the center of the slice. The middle loop determines which layer line is being used and the outer loop determines for which slice calculations are being made. The calculated value for the coordinate is stored in the variable YTT(M,J) where M is the number of the boundary layer line and J is the number of the slice. Figure 17 shows this procedure and defines the variables for a three-layer problem.

NOTES:



$$YR = \frac{YLS(I+1,NL1) - YLS(I,NL1)}{XLS(I+1,NL1) - XLS(I,NL1)} \times [XC(J) - XLS(I,NL1)] + YLS(I,NL1)$$

Figure 16. **Bottom Layer Check.**



$$YTT(M,J) \quad \frac{YLS(K+1,M) - YLS(K,M)}{XLS(K+1,M) - XLS(K,M)} \times \left[XC(J) - XLS(K,M)\right] + YLS(K,M)$$

Figure 17. Computation of YTT(M, J).

1. This calculation is made after determining which ground-line segment intersects the centerline of the slice.
2. The intersections of the centerline of the slice with all layer boundary lines are computed even though the YTT(M,J)'s which lie below the slip circle may not be used in further computations.

XV. Determine straight-line equations for all water-table line segments.

The same procedure as III is used here except there can be no vertical line segments or overledges in the water-table line. The slope is stored in the variable SLOP(I) and the y-intercept is stored in the variable YINTER(I) where I is the number of the water-table line segment.

XVI. Calculate the cosine of the angle each water-table line segment makes with the horizontal (for each slice).

Two nested loops are used for this operation. The inner loop determines the water-table line segment used and the outer loop determines the slice for which calculations of COSI(J) are made. A slightly simpler determination of the water-table line segment is used here. If XC(J) is greater than XWT(I+1), where I comes from the inner loop, no calculations are made and I is incremented by 1 until XC(J) is less than or equal to XWT(I+1). At this time, COSI(J) is calculated and the outer loop is incremented by 1 to start the next slice.

XVII. Determine total weight of each slice.

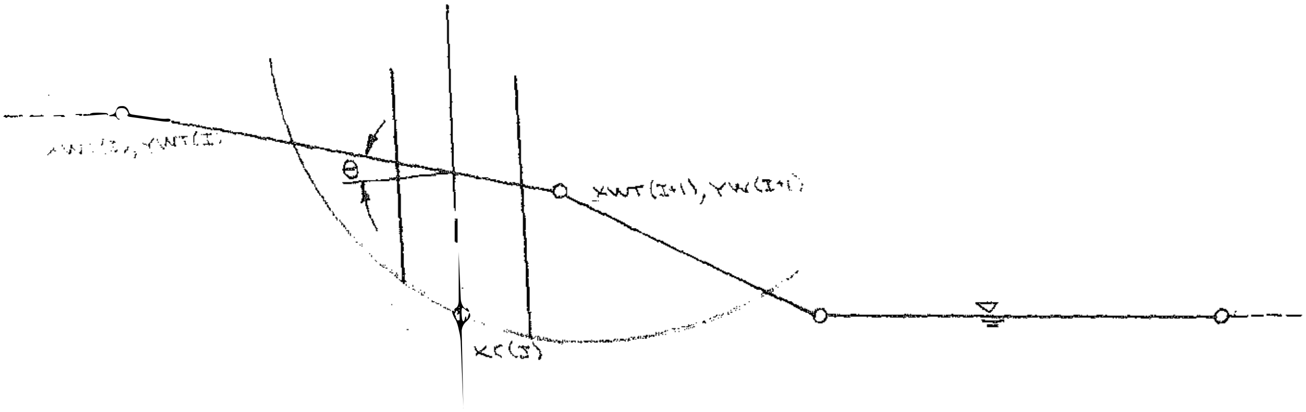
In simplified terms, this part of the program calculates the height of soil in each layer in each slice and multiplies by the unit weight of the soil in that layer, WT(M), and sums them to obtain a total weight of slice, W(J). To understand this operation, a typical cross section for one slice is shown in Figure 19. An enlargement of the slice is shown in Figure 20a. If YC(J) would have been in Layer 1, the program would have calculated W(J) as

$$W(J) = W(J) + BB\,[YG(J) - YC(J)]\,WT(M)$$

where M = 1.

$$A4 = BB[YG(J) - YC(J)] + A4$$

MB is now defined to be the layer line directly above YC(J) and M is now defined to be the top layer. Since YC(J) lies below Layer 1, the weight and cross-sectional area of the soil in Layer 1 in Slice J is now computed as in Figure 20b:



$$\cos \theta_j = \mathrm{COSA}(J) = \frac{\mathrm{XWT}(I+1) - \mathrm{XWT}(I)}{\sqrt{[\mathrm{YWT}(I+1) - \mathrm{YWT}(I)]^2 + [\mathrm{XWT}(I+1) - \mathrm{XWT}(I)]^2}}$$

Figure 18. Computation of COSI(J).

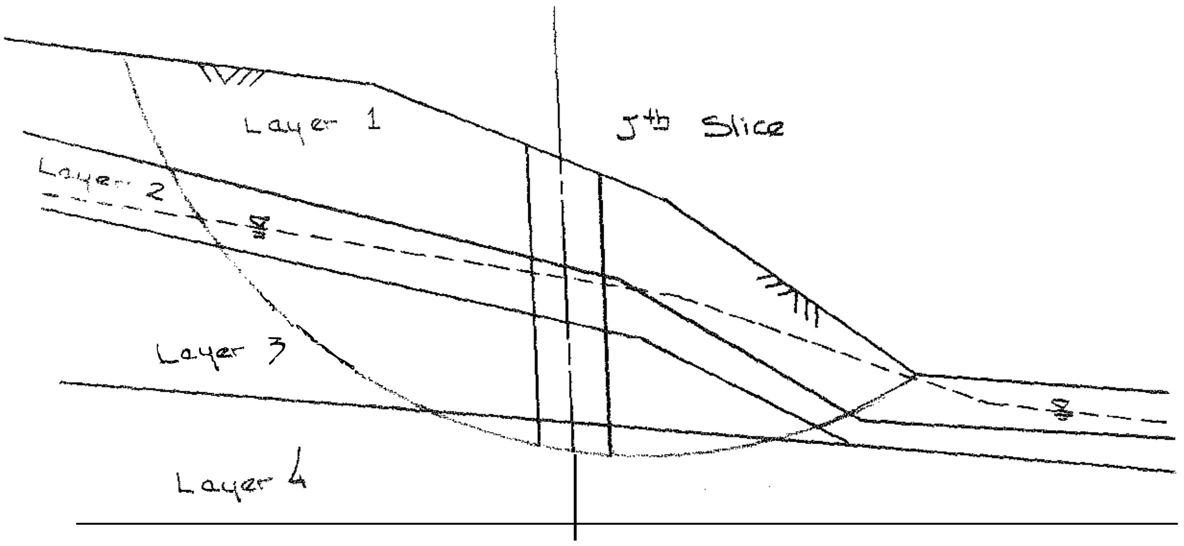
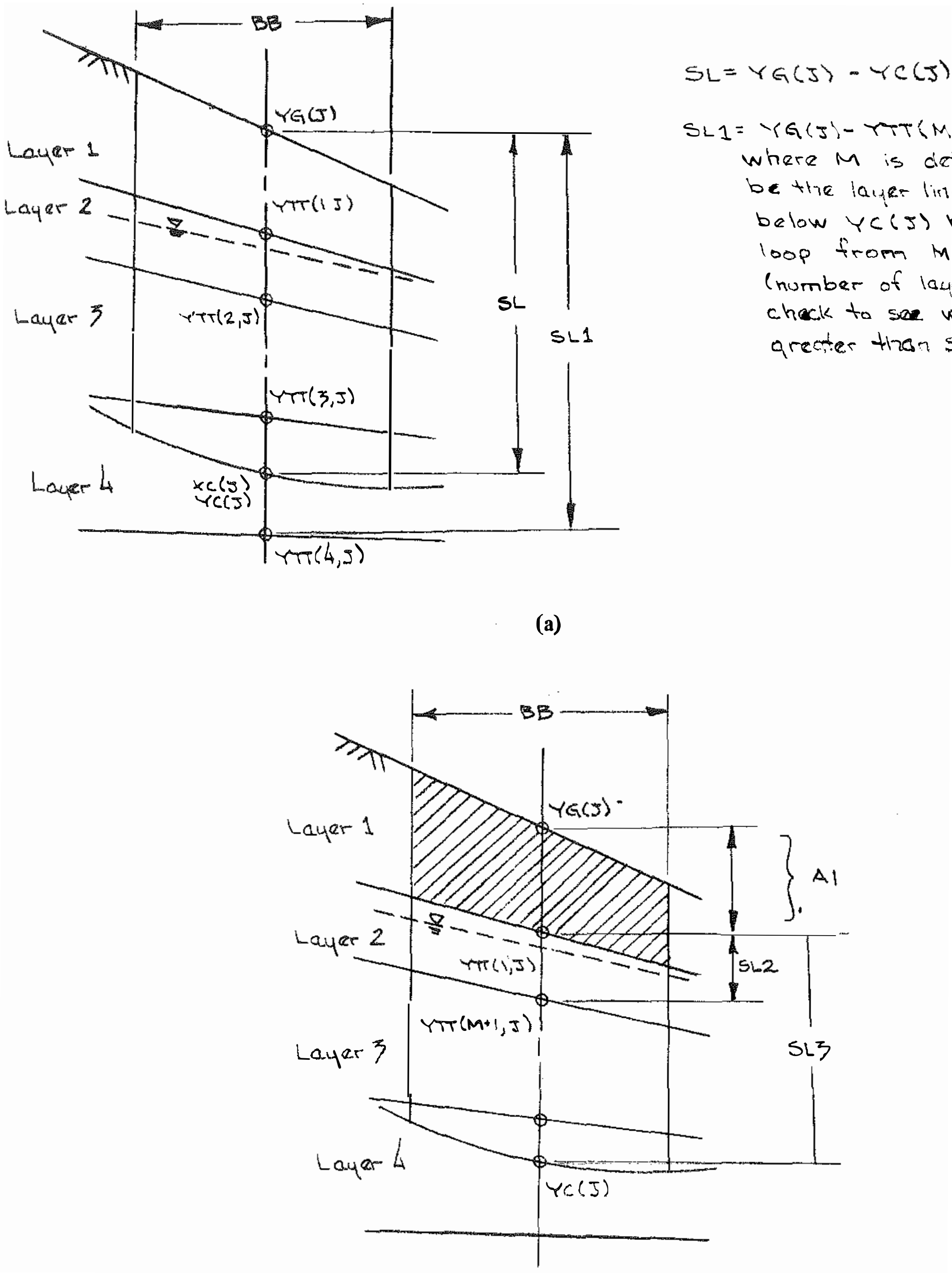


Figure 19. Typical Cross Section.



SL= YG(J) - YC(J)

SL1= YG(J)- YTT(M,J)
where M is determined to be the layer line directly below YC(J) by using a loop from M = 1 to M= NL (number of layers) and check to see when SL1 is greater than SL.

Figure 20. Computation of Total Weight of a Slice.

Z1 = BB[YG(J) - YTT(M, J)] + A1

WJ = W(J) + BB[YG(J) - YTT(M, J)]WT(M)

NOTE: The additional W(J) on the right side of the equation is used to obtain a total weight of the slice:

SL2 = [YTT(M, J) - YTT(M+1,J)]

SL3 = [YTT(M, J) - YG(J)]

If SL2 would have been greater than or equal to SL3, the program would have calculated the remaining weight and area in the slice as follows:

A3 = BB[YTT(M, J) - YC(J)] + A3

W(J) = W(J) + BB [YTT(M, J) - YC(J)] WT(M+1).

Since SL2 is less than SL3 (which means YC(J) lies in the layer directly below A1), a loop is set up to calculate weights and areas of layers between MT(top layer) and MB (layer directly above YC(J)) shown in Figure 20c:

A2 = BB[YTT)M, J) - YTT(M+1, J)] + A2

W(J) = W(J) + BB [YTT(M, J) - YTT(M+1, J)] WT(M+1)

The loop allows a sum of all "middle" layers to be grouped into A2. A3, the remaining portion of the slice, as shown above, is easily calculated:

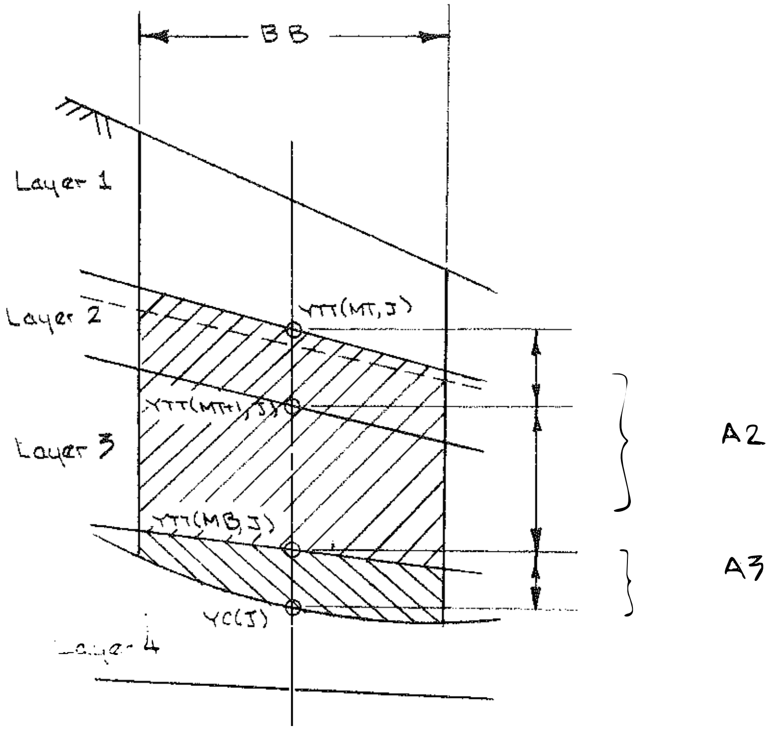
A3 = BB[YTT(M, J) - YC(J)] + A3

W(J) = W(J) + BB [YTT(M,J) - YC(J)]

where M = MB.

XVIII. Determine effective weight of each slice.

The effective weight of the slice depends on water-table conditions. A complete understanding of these conditions can be obtained from a study of the method of analysis of the Bishop method of slices.



(c)

Figure 20 (Con't). Computation of Total Weight of a Slice.

If the water table lies below YC(J); i.e. YT(J) is less than or equal to YC(J), the effective weight of the slice, WE(J), equals the total weight of the slice, W(J).

If YT(J) is greater than YC(J), the weight of water in the slice, WW(J), is calculated in one of two ways, depending on the routing gimmick EFFWT.

A. When EFFWT equals 1.

$$WW(J) = [YT(J) - YC(J)]\,[COSI(J)]^2\,(0.0623999)(BB)$$

This equation assumes an infinitely sloping water table and finds the height from YC(J) to a point perpendicular to the water table line segment. This is also used when hydrostatic water-table conditions exist, since for a horizontal line $[COSI(J)]^2 = 0$.

B. When EFFWT is not equal to 1.

$$WW(J) = [YT(J) - YC(J)]\,(0.0623999)(BB)$$

This is used when an effective water table is supplied as input. An effective water table is a line representing the piezometric height of the pore pressures along the slip surface. The effective weight is now calculated:

$$WE(J) = W(J) - WW(J)$$

XIX. Determine total area of failure.

The area of failure for each slice is the sum of A1, A2, A3, A4 and the total area of failure for the slope is

$$AA = AA + A1 + A2 + A3 + A4$$

where the AA on the right side of the equation is the area of failure for all previous slices.

XX. Determine the driving weight of each slice.

Again, the driving weight depends on water-table conditions. A complete understanding of driving weight (or shear stress) can be obtained from a study of Bishop's method of slices.

As discussed previously, the driving weight of the slice is the total weight of the slice less the weight of water obtained from the horizontal projection of a hydrostatic water table on the "toe" side of the cross section. This implies conditions shown in Figure 21. Therefore, the driving weight is

W(J) - HWW(J)

where HWW(J) = $z\gamma_w$. For cases where the water table is sloping throughout the cross section or when there is a hydrostatic section of water and YC(J) is greater than YWT(NOWT), z = 0. Here the driving weight is just W(J).

XXI. Calculate factor of safety.

The equation used by this program is the simplified Bishop equation:

$$FS = \Sigma\left\{[c'_i b_i + (W_i - u_i b_i)\tan\phi'_i]/M_i(\theta)\right\}/\Sigma(W_i - z\gamma_w)\sin\theta_i$$

where c' = CO(M)

M = layer where slip occurs for the slice = LC(J)

W_i = W(J)

$u_i b_i$ = WW(J)

$\tan\phi'_i$ = TANPHI(M)

$M_i(\theta)$ = $\cos\theta_i + (\sin\theta_i \tan\theta_i)/FS$ = MTHETA = COSA(J) + [SINA(J) TANPHI(M)]/FS

$\cos\theta_i$ = COSA(J)

$\sin\theta_i$ = SINA(J)

This procedure is done using a loop and incrementing J from 1 to NSLICE. Since FS cannot be solved for directly, an iteration process is used. The first value of FS is given in the input data and a new FS is calculated by summing the numerator and denominator for each slice and then dividing:

TOP = TOP + [CO(M) BB + EFF TANPHI(M)]/MTHETA

BOT = BOT + [W(J) - HWW(J)] SINA(J)

where EFF = WE(J).

FS = TOP/BOT.

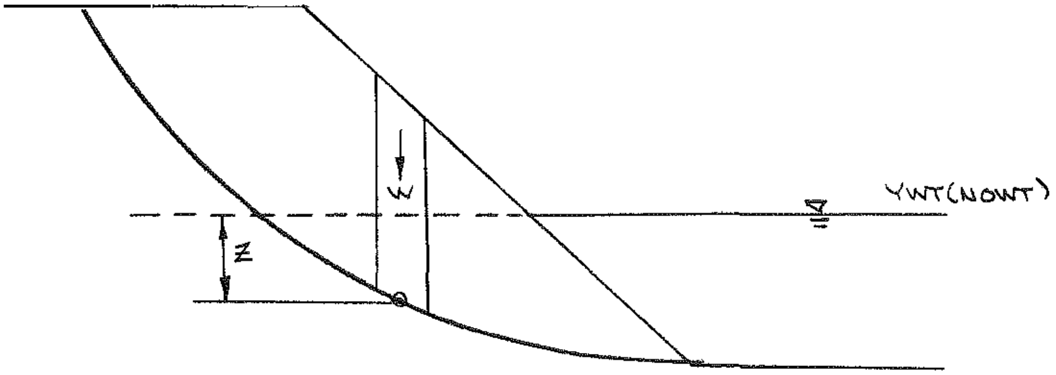


Figure 21. Computation of Driving Weight of a Slice.

If the calculated FS is within 0.005 of the assumed FS, calculations are terminated, If not, the calculated FS becomes the assumed value and the operation is repeated until the values converge. During these calculations, MTHETA is checked to determine if it is less than 0.2. If it is, the variable MTH, which was set equal to zero before each iteration, becomes equal to 1. This is done so that at the time of printout, a warning can be typed indicating MTH < 0.2. Also, due to the width of the typed output, the value of the factor of safety must be less than ten. If the final value for the factor of safety is greater than or equal to ten, the program increments to the next radius.

If a ledger type printout is required (POUT not equal to 1.), the final value of the factor of safety is used to repeat computations and allow for printout of all important quantities for each slice. These quantities are: NSLICE, W(J), WW(J), WE(J), MTHETA, NBAR, COHES, TOPS, BOTS, COSA(J), and SINA(J). Additional computations are necessary to determine NBAR, COHES, TOPS and BOTS:

NBAR = WE(J) - [CO(M)   BB   SINA(J)/COSA(J)]/FS   MTHETA
COHES = CO(M)   BB
TOPS = CO(M)   BB + [EFF   TANPHI(M)]/MTHETA
BOTS = W(J)   SINA(J)

At the bottom of the page, the value for the average shear stress is printed. It is computed as

AVESS = BOT/ARCLN

where ARCLN is an approximation of the summation of BB/COSA(J) for each slice.

XXII. The remainder of the program deals with various output options. It is, for the most part, the different format statements used to generate each type of printout.

APPENDIX C

VARIABLE LIST

VARIABLE LIST

VARIABLE	DEFINITION	FIELD (IF REQUIRED)
A	Value used in determining intersection of ground line and slip circle	
AA	Total area of failure	
ABC	Change in horizontal distance from one point on ground line to the next	
ARCLIN	Approximate arc length	
AVESS	Average shear stress	
A1, 2, 3, 4	Values used to determine total area of failure	
B	Value used in determining intersection of ground line and slip circle	
BB	Width of each slice	
BOT	Sum of terms in denominators (for all slices) in Bishop's equation	
BOTS	Same as BOT, but for each slice only	
C	Value used in determining intersection of ground line and slip circle	
CO(M)	Cohesion for each layer in kips per square foot	F5.0
COHES	Cohesion of soil in failure times width of slice	
COSA(J)	Cosine of the angle between the tangent to the slip circle at each slice and the horizontal	
COSI(a)	Cosine of the angle each water table line segment makes with the horizontal	
EFF	Effective weight of each slice	
EFFWT	Routing gimmick	F5.0

F	Factor of safety	
FST(IXX,IYY)	Least factor of safety for each grid point	
FSI	Initial factor of safety used in iteration process	F5.2
HOP	Highest ground-line point on each ground-line segment, used when calculating intersection of slip circle and ground line	
HWW(J)	Weight of water below hydrostatic tailwater	
IAN	Analysis number	A6
ICO1, 2	Name of county	2A6
IDEL1	X-coordinate increment for grid system	I3
IDEL2	Y-coordinate increment for grid system	I3
ID1, 2, 3, 4	Identification of problem	2A6, 2A6
IFIN	Final X coordinate of grid system	I10
IN	Method of input code	I4
IOUT	Method of output code	I4
IPNF1, 2, 3	Project number	2A6, A2
IPNS1, 2, 3	Project designation	2A6, A2
IRF	Final radius length for grid system	I10
IRS	Initial radius length for grid system	I10
ISTART	Initial X coordinate for grid system	I10
IRT1, 2	Route designation	2A5
IX, IY	Variable subscripts to initialize all factors of safety for grid system to zero	
IXO	X coordinate of grid point at time of calculations	
IXX	Number of the X coordinate of grid system, i.e. the first X coordinate or column of grid is 1, etc.	
IYO	Y coordinate of grid point at time of calculations	
IYY	Number of the Y coordinate of grid system,	

	i.e. the first Y coordinate or row of grid is 1, etc.	
J	Variable subscript used to designate slices	
JFIN	Final Y coordinate of grid system	I10
JJJ	Counter for searching operation to determine lowest factor of safety for each grid point	
JSTART	Initial Y coordinate of grid system	I10
KDAY	Two-digit number corresponding to present day	A2
KYEAR	Two-digit number corresponding to present year	A2
L	Counter for printout control	
LC(M)	Counter for layers to determine soil parameters at slip surface	
LOP	Lowest ground line point on each ground line segment, used when calculating intersection of critical circle and ground line	
M	Variable subscript used to designate layers	
MB	Counter for layers	
MONTH	Two-digit number corresponding to present month	A2
MT	Counter for layers	
MTH	Variable to monitor $M_i(\theta)$	
MTHETA	$M_i(\theta)$ from Bishop's equation	
NBAR	N from general circular failure equation	
NL	Number of soil and rock layers	I4
NL1	Number of the bottom layer	
NL2	Number of boundary-layer line segments	
NLPL	Number of boundary-layer line segments	
NN	Number of water-table line segments	
NNN	Number of water-table line segments	
NO	Number of coordinate points on ground line	I4

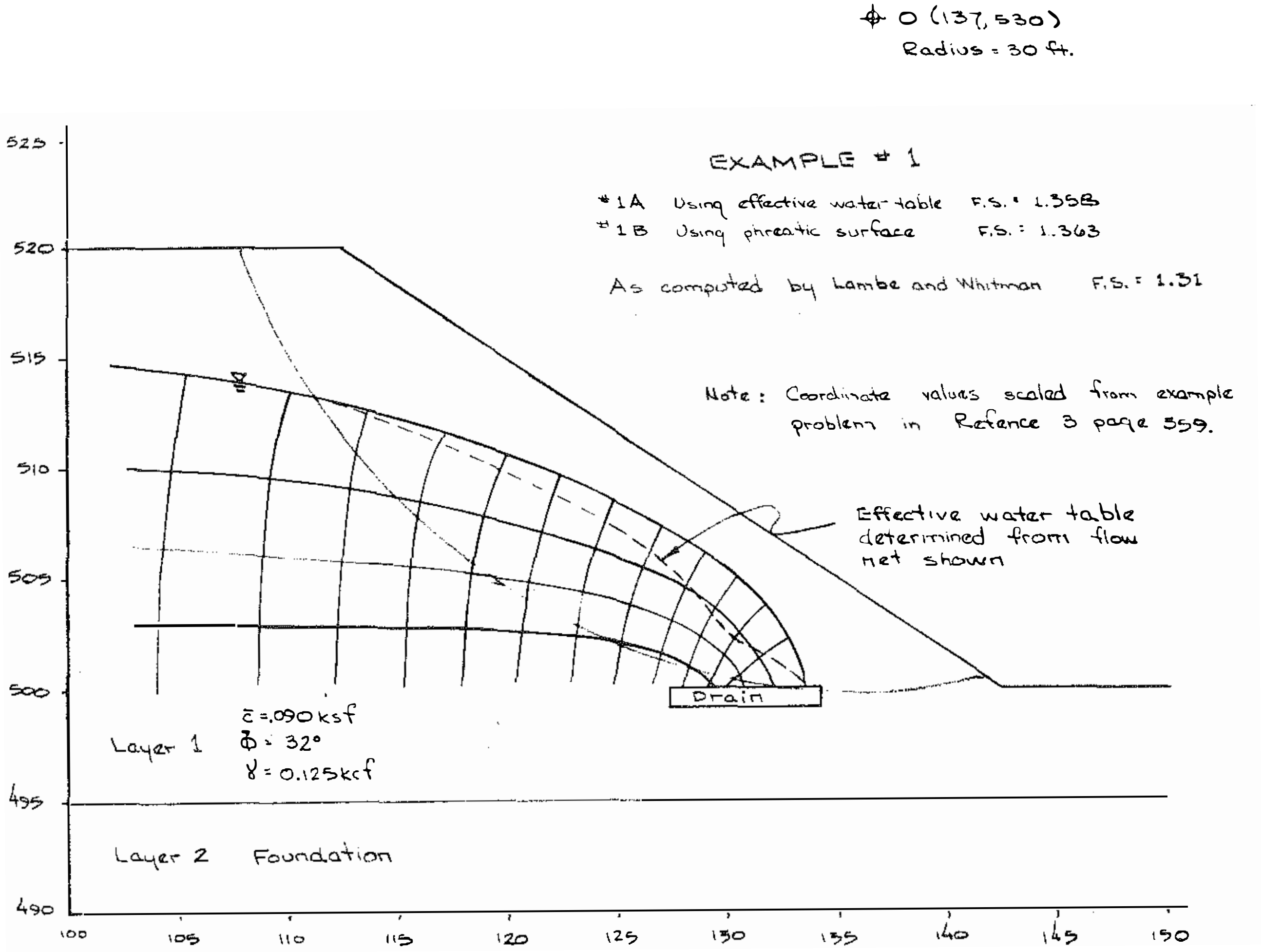
NOP	Number of data decks (problems) submitted	I4
NOPL	Number of coordinate points on layer boundary lines	I4
NOWT	Number of coordinate points on water-table line	I4
NSLICE	Number of slices to be used in analysis	I4
POUT	Routing gimmick	F5.0
RO	Real value of radius used in calculations	
SINA(J)	Sine of the angle the tangent to the slip circle at each slice makes with the horizontal	
SL	Vertical distance from ground line to slip circle at each slice	
SL1	Vertical distance from ground line to layer boundaries at each slice	
SL2	Vertical distance from slip circle to layer boundary directly below slip circle at each slice	
SL3	Vertical distance from slip circle to layer boundary directly above slip circle at each slice	
SLOP(I)	Slope of each water-table line segment	
SLOPE(I)	Slope of each ground-line segment	
SOIL	Routing gimmick	F5.0
SOIL1	Routing gimmick	F5.0
TANPHI(M)	Tangent modulus phi for each layer (tan ϕ)	F5.0
TEST	Value used in determining intersection of slip circle and ground line	
TOP	Sum of terms in numerator for all slices) in Bishop's equation	
TOPS	Same as TOP, but for each slice only	
W(J)	Weight (either buoyant or total) of each slice	

WE(J)	Effective weight of each slice	
WT(J)	Unit weight of soil in each layer in kips per cubic foot	F5.0
WW(J)	Weight of water in each slice	
X(I)	X coordinate of ground-line segment	F10.3
XA	X coordinate of intersection of ground line and critical circle on uphill side	
XB	X coordinate of intersection of ground line and critical circle on downhill side	
XC(J)	X coordinate of center of each slice on critical circle	
XCOR(I)	X coordinate of grid point	
XE	X coordinate of minimum radius point	F10.3
XLS(K,M)	X coordinate of layer-boundary line segment	F10.3
XO	Real value of X coordinate of grid point at time of calculations	
XP, XM	X coordinates of intersection of critical circle and each ground-line segment	
XWT(I)	X coordinate of water-table line segment	F10.3
Y(I)	Y coordinate of ground-line segment	F10.3
YA	Y coordinate of intersection of ground line and slip circle on uphill side	
YB	Y coordinate of intersection of ground line and slip circle on downhill side	
YC(J)	Y coordinate of center of each slice on slip circle	
YCOR(I)	Y coordinate of grid point	
YE	Y coordinate of minimum radius point	F10.3
YG(J)	Y coordinate of center of each slice on ground line	
YINT(I)	Y intercept in straight-line equation for each ground-line segment	

YINTER(I)	Y intercept in straight-line equation for each water-table line segment	
YLS(K,M)	Y coordinate of layer-boundary line segment	F10.3
YO	Real value of Y coordinate of grid point at time of calculations	
YP, YM	Y coordinates of intersection of failure circle and each ground-line segment	
YR	Value used to check if circle intersects bottom layer	
YT(J)	Y coordinate of center of each slice on water table	
YTT(M,J)	Y coordinate of center of each slice on each layer boundary line	
YWT(I)	Y coordinate of ground-line segment	F10.3
Z	True distance from grid point to minimum radius point	

APPENDIX D

EXAMPLE PROBLEMS



EXAMPLE # 1

#1A Using effective water table F.S. = 1.358
#1B Using phreatic surface F.S. = 1.363

As computed by Lambe and Whitman F.S. = 1.31

Note: Coordinate values scaled from example problem in Refence 3 page 559.

CARD COLUMN NUMBER

```
1         11        21        31        41        51        61        71       80
EXAMPLE #1A                  EFFWT=0.                                          1
FROM LAMBE AND WHITMAN  SOIL MECHANICS  TEXT REFERENCE #3   PAGE 359
020673
   0.    0.   0.    0.
  20   16   2    2    4   1.0
          137       530       137       530        30        30  1  1  1
112.5     520.
 .09  32. .125 ]
9999..0000.150 ] LAYER PROPERTIES
  0.      520. ]
112.5     520. ]
142.5     500. ] GROUND LINE
200.      500. ]
  0.      495. ]
200.      495. ] LAYER 1
  0.      480. ]
200.      480. ] LAYER 2
  0.      515.    ]
107.5     514.    ]
112.5     512.7   ]
115.0     511.75  ]
117.5     510.75  ]
120.0     510.    ]
122.0     509.0   ]
123.75    508.0   ] WATER TABLE
125.75    507.0   ]
127.0     505.75  ]
128.5     504.5   ]
130.0     503.25  ]
131.25    501.25  ]
132.75    500.5   ]
133.      400.    ]
200.      400.    ]
```

INPUT DATA DECK FOR EXAMPLE #1A

CARD COLUMN NUMBER

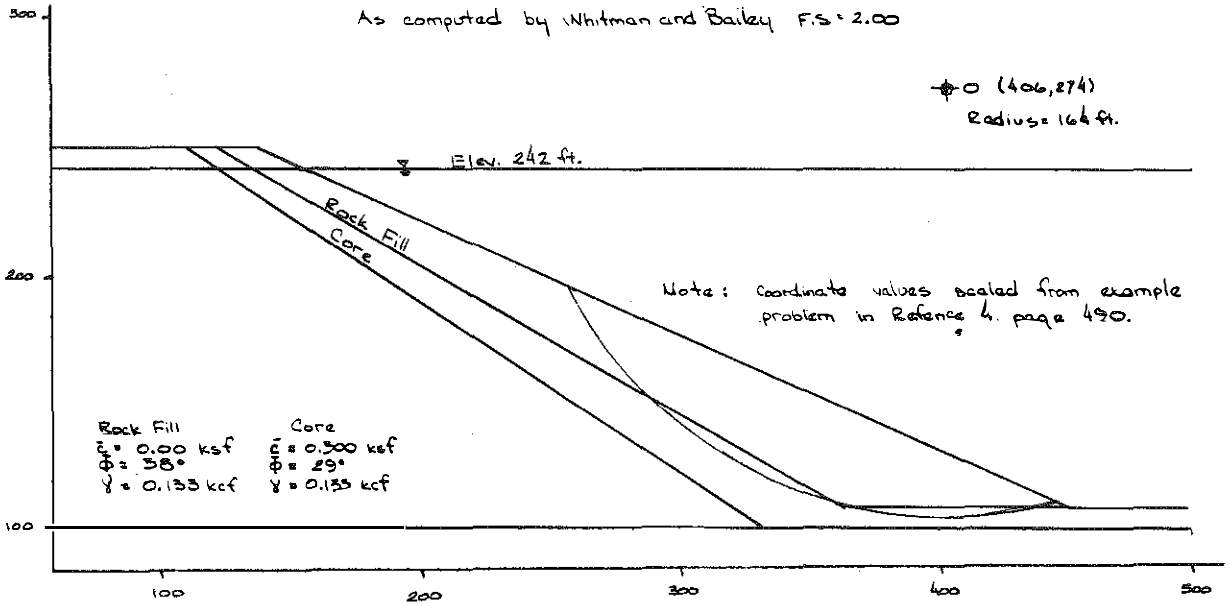
```
1         11        21        31        41        51        61        71       80
EXAMPLE #1B              EFFWT=1.                                              1
FROM LAMBE AND WHITMAN  SOIL MECHANICS  TEXT REFERENCE #3    PAGE 359
020673
   0.   0.   0.   1.
  20  16   2   2   4  1.0
       137       530       137       530        30        30  1  1  1
112.5     520.
 .09  32. .125
9999..0000.150
  0.      520.
112.5     520.
142.5     500.
200.      500.
  0.      495.
200.      495.
  0.      480.
200.      480.
  0.      515.
108.      514.
115.      512.
120.      510.
123.      509.
125.      508.
127.      507.
130.      505.
131.      504.
132.      503.
133.      501.
133.5     500.
134.      450.
180.      450.
190.      450.
200.      450.
```

INPUT DATA DECK FOR EXAMPLE #1B

EXAMPLE #2

F.S. = 2.073

As computed by Whitman and Bailey F.S = 2.00



Note: Coordinate values scaled from example problem in Reference 4. page 490.

Rock Fill	Core
$\bar{c}$ = 0.00 ksf	$\bar{c}$ = 0.300 ksf
$\bar{\phi}$ = 38°	$\bar{\phi}$ = 29°
γ = 0.133 kcf	γ = 0.135 kcf

CARD COLUMN NUMBER

```
1         11        21        31        41        51        61        71       80
EXAMPLE #2                                                                     1
ASCE SOILS AND FOUNDATIONS JOURNAL  REFERENCE #4 PAGE 490  WHITMAN AND BAILEY
020673
   0.    0.    1.    1.
  50   2   5   4   4 1.0
       406       274       406       274       164       164 1  1  1
400.      200.
   0..0   .0624
   0.38.  .133
 .30029.  .133
   0.30.  .133
9999..0   .150
  0.      250.
140.      250.
158.      242.
900.      242.
  0.      250.
140.      250.
452.      112.
900.      112.
  0.      250.
123.      250.
364.      112.
900.      112.
  0.      250.
110.      250.
334.      105.
900.      105.
  0.      105.
200.      105.
800.      105.
900.      105.
  0.       50.
900.       50.
910.       50.
920.       50.
  0.      242.
900.      242.
```

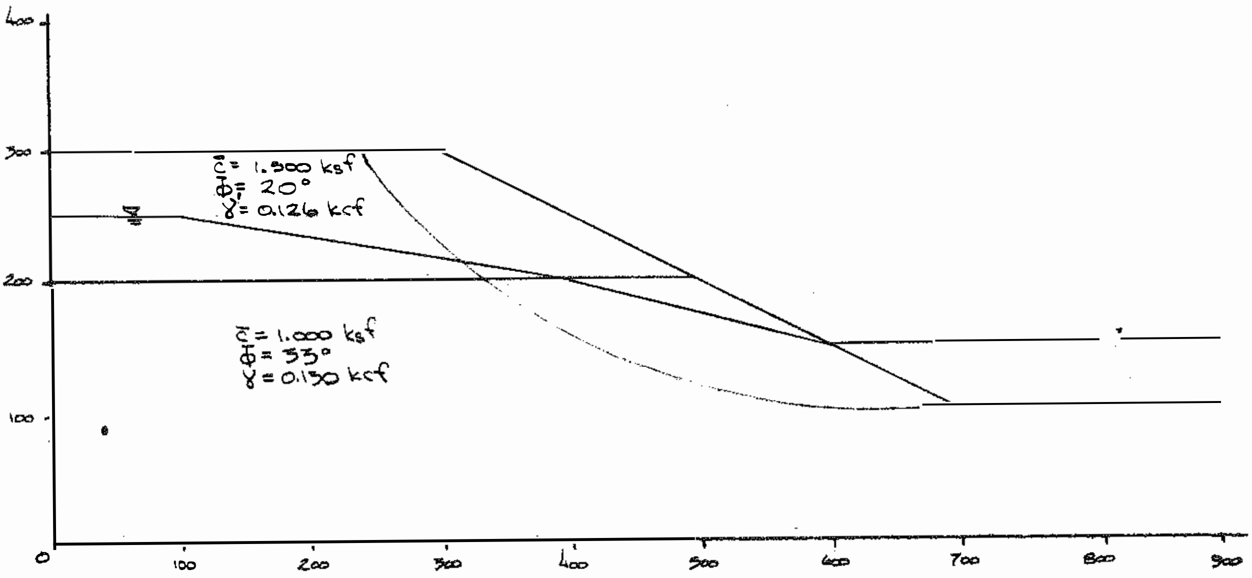
INPUT DATA DECK FOR EXAMPLE #2

EXAMPLE #3

F.S. = 1.580

⊕ O (636, 600)
Radius = 500 ft.

As computed by Whitman and Bailey F.S. = 1.569



CARD COLUMN NUMBER

```
1         11        21        31        41        51        61        71      80
EXAMPLE #3                                                                     1
ASCE SOILS AND FOUNDATIONS JOURNAL REFERENCE #4  PAGE 494  WHITMAN AND BAILEY
020673
   0.   0.   1.   1.
   50    5  3  4   5 1.0
       636       600       636       600       500       500  1  1  1
500.      200.
   0..0   .0624
  1.5 20. .126
  1.0 33. .130
  0.      300.
300.      300.
500.      200.
600.      150.
900.      150.
  0.      300.
300.      300.
700.      100.
900.      100.
  0.      200.
500.      200.
700.      100.
900.      100.
  0.       20.
800.       20.
850.       20.
900.       20.
  0.      250.
100.      250.
400.      200.
600.      150.
900.      150.
```

INPUT DATA DECK FOR EXAMPLE #3