Feasibility and Cost of Using a Computer to Prepare Landslide Susceptibility Maps of the San Francisco Bay Region, California

GEOLOGICAL SURVEY BULLETIN 1443

Newman and others—LANDSLIDE SUSCEPTIBILITY MAPS, SAN FRANCISCO BAY REGION—Geological Survey Bulletin 1443

Feasibility and Cost of Using a Computer to Prepare Landslide Susceptibility Maps of the San Francisco Bay Region, California

By EVELYN B. NEWMAN, ARTHUR R. PARADIS,
and EARL E. BRABB

GEOLOGICAL SURVEY BULLETIN 1443

UNITED STATES GOVERNMENT PRINTING OFFICE, WASHINGTON: 1978

UNITED STATES DEPARTMENT OF THE INTERIOR

CECIL D. ANDRUS, *Secretary*

GEOLOGICAL SURVEY

W.A. Radlinski, *Acting Director*

Library of Congress Cataloging in Publication Data

Newman, Evelyn B.
Feasibility and cost of using a computer to prepare landslide susceptibility maps of the San Francisco Bay Region, California.
(Geological Survey Bulletin 1443)
Bibliography: p. 10
Supt. of Docs. no.: I 19.3:1443
1. Landslides–Maps–Computer programs. 2. Landslides–Maps–Costs. 3. Landslides–California–San Francisco Bay region–Maps. I. Paradis, Arthur R., joint author. II. Brabb, Earl E., 1929- joint author. III. Title: Feasibility and cost of using a computer to prepare landslide susceptibility maps... IV. Series: United States. Geological Survey. Bulletin 1443.
QE75.B9 no. 1443 [QE599.A2] 557.3'08s [551.3'5'3] 77-608264

For sale by the Superintendent of Documents, U. S. Government Printing Office
Washington, D. C. 20402

Stock Number 024-001-03067-5

CONTENTS

ILLUSTRATIONS

TABLES

FEASABILITY AND COST OF USING A COMPUTER TO PREPARE LANDSLIDE SUSCEPTIBILITY MAPS OF THE SAN FRANCISCO BAY REGION, CALIFORNIA

By Evelyn B. Newman, Arthur R. Paradis, and Earl E. Brabb

ABSTRACT

Geologic, landslide, and slope maps of an area near San Francisco were digitized and converted to grid cell form for easy manipulation by computer. Landslide susceptibility categories were assigned to grid cells on the basis of the percentage of the rock units that had failed by landsliding in each slope category. The resulting landslide susceptibility map units were assigned color codes by computer, and color film was exposed by an image recorder. A comparison of the manual and computer-generated versions of the area shows the feasibility of compiling landslide susceptibility maps by computer in approximately the same time and at less cost than the manual version, with the benefits of less human labor and error and the availability of grid cell data for future mapping.

INTRODUCTION

Landslides were at least a $25,000,000 problem in the San Francisco Bay region during the rainy season of 1968–69 (Taylor and Brabb, 1972). The U.S. Geological Survey, in cooperation with the U.S. Department of Housing and Urban Development, is studying these landslides and other geologic problems as part of a pilot project to test the usefulness of environmental resource data in improving urban planning and decisionmaking. One of the early products of this project is a landslide susceptibility map of San Mateo County at 1:62,500 scale (Brabb and others, 1972). The map provides an easily read analysis of selected geologic factors related to landsliding in the county. Unfortunately, the cost and difficulty of preparing the map were greater than expected; therefore, simpler, less expensive, and smaller scale (1:125,000) maps are being prepared for the rest of the nine-county bay region.

The need for large-scale (1:62,500 and larger) landslide susceptibility maps has been firmly established in the San Francisco Bay region. The San Mateo County map has been used by the county to establish the density of development, to require geologic investigations before development is approved, and to prepare seismic safety, open-space, and conservation elements of the county general plan. A similar map at 1:24,000 scale is being prepared for the city of San Jose. Several other counties and cities have expressed the desire for these maps if the cost is reasonable, which generally means a few tens of thousands of dollars.

There is, in addition to the need for landslide susceptibility maps, a need for more understanding of the relative importance of all the factors related to the landslide process. Brabb, Pampeyan, and Bonilla (1972) selected the areal extent of landsliding in each geologic unit and the original slope as the most critical factors in San Mateo County, but other factors such as the orientation of bedding relative to the slope, nearness to faults, rainfall, and vegetation should also be investigated. They were not investigated for the San Mateo County map because no simple correlations could be established by visual inspection, and because of the great difficulty in analyzing several subtle factors simultaneously in a large area.

The purpose of this study was to determine if computer techniques could be used to make a landslide susceptibility map of a selected test area of approximately 15 square miles in San Mateo County at a reasonable cost, and to estimate the cost of preparing similar maps for representative counties in the San Francisco Bay region. The location of the test area is shown on figure 1. The investigation will be used to establish the methodology and eventually the cost for more sophisticated regional analyses of several factors related to the formation of landslides.

Evelyn Newman wrote most of the report and selected the computer methods used in the analysis. Arthur Paradis converted map information into numerical form (digitized) and wrote most of the computer programs. Earl Brabb wrote part of the report and was responsible for determining the scope and objectives of the investigation.

PREPARATION OF A LANDSLIDE SUSCEPTIBILITY MAP

MANUAL METHOD

The original landslide susceptibility map of San Mateo County (Brabb and others, 1972) was prepared from an analysis of a geologic map (Brabb and Pampeyan, 1972a), a landslide map (Brabb and Pampeyan, 1972b), and an experimental slope map, all at 1:62,500 scale. The maps were originally used in the following manner:

1. The area of outcrop within San Mateo County was determined for each of the geologic formations (rock units) and some subunits on the geologic map using a grid overlay with a resolution of 0.01 mi^2 (6.4 acres or 0.0259 km^2) at the map scale.
2. The landslide inventory map was superimposed on the geologic

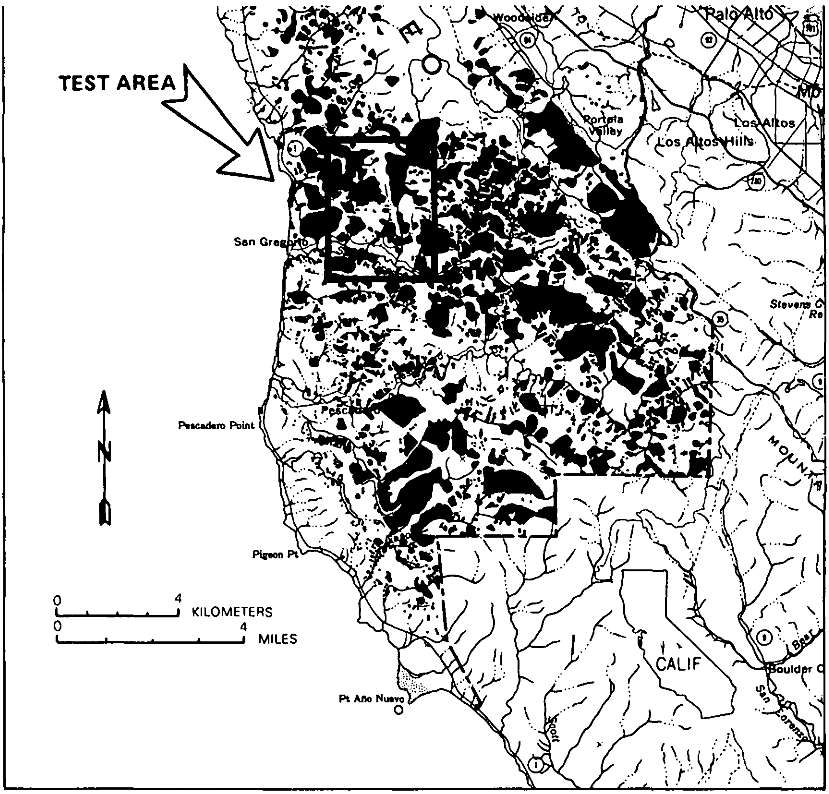

FIGURE 1.—Index map showing landslides in San Mateo County, Calif., and location of test area.

map in order to identify the rock units in which slope failures had occurred. The areas that had failed in each unit were measured using the grid.

3. The rock units were then listed in order of percentage of their outcrop areas that have failed by landsliding (see table 1).

4. The highest class of susceptibility (7) was assigned to the landslide deposits.

5. Other class limits were established at selected intervals on the list, and a class number from 1 to 6 was assigned to the map units. That number represents the relative susceptibility to landslide failure of any particular geologic unit.

6. The slope map was then superimposed on the combined geologic map and landslide inventory and systematically examined to determine the slope intervals with maximum landslide frequency for each map unit. In every landslide locality, an attempt was made to determine the original slope before the landslide moved. Those slope inter-

TABLE 1.—*Landslide susceptibility class number for rock units in test area in San Mateo County*

[From Brabb and others, 1972]

Proportion of surface of rock unit that has failed by landsliding (percent)	Rock unit on geologic map by Brabb and Pampeyan (1972a), in order of increasing proportion of surface having failed by landsliding	Map symbol	Approximate area of rock unit in county (mi^2)	Approximate area that has failed in county (mi^2)	Relative susceptibility number	Susceptibility numbers in each slope interval: 0–5	5–15	15–30	30–50	50–70	>70
0–1	None in test area				1						
2–8	Slope wash and ravine fill	Qt	4.51	0.18	2	1	1	1	2	2	2
9–25	None in test area				3						
	Purisima Formation, undivided	Tp	23.06	7.81		1	2	3	4	4	4
26–42	Mindego Basalt and other volcanic rocks	Tmb	10.80	4.01	4	1	1	2	3	3	3
	Butano Sandstone along Butano Ridge	Tb	20.18	7.60		1	1	2	3	4	4
	San Gregorio Sandstone member of Purisima Formation[1]	Tpsg	2.41	1.06		1	1	1	4	5	5
	Tunitas Sandstone member of Purisima Formation[1]	Tptu	2.76	1.24		1	1	3	5	5	5
43–53	Tahana Member of Purisima Formation[1]	Tpt	33.46	16.08	5	1	2	3	5	5	5
	Pomponio Member of Purisima Formation[1]	Tpp	11.97	5.76		1	1	2	5	5	5
54–70	Lobitos Mudstone member of Purisima Formation[1]	Tpl	3.71	2.57	6	1	2	2	6	6	6
71–99	None in test area										
100	Landslide deposits	Qls	83.88	83.88	7	7	7	7	7	7	7

[1]Cummings, Touring, and Brabb, 1962.

vals having the greatest number of landsides were then labeled with the highest class number. Slope intervals showing significantly fewer landslides were labeled with lower class numbers. Thus, a geologic unit having a maximum susceptibility of 3 would be labeled with that number on steep slopes, and with 2 or 1 on more gentle slopes with significantly fewer slides.

About 6 man-months of very tedious and meticulous labor were required to prepare the landslide susceptibility map of the entire San Mateo County area. The estimated cost was about $30,000.

COMPUTER METHOD

The same geologic, landslide, and slope maps for the test area were analyzed using a computer, but the procedures varied from the manual method.

GENERATING GRID-CELL MAPS

We experimented with two grid-cell sizes, 500 feet (152.4 m) on a side and 250 feet (76.2 m) on a side at map scale. The larger grid was designed to correlate with the 500-foot reliability figure mentioned by Brabb, Pampeyan, and Bonilla (1972), but the maps produced were too generalized and had such a blocky appearance in relation to the original maps prepared manually that we chose to use the smaller grid (see fig. 2). The 250-foot (1.4 acre) cell size ensured that the smallest landslide and geologic units and most of the detailed slope units would be mapped; however, it may be beyond the limits of accuracy of the original data. Further experimentation with grid-cell sizes between 250 and 500 feet is warranted but was not possible during the present investigation. The 1.4-acre cell size produced 6,734 grid cells in the test area, which was within the computer processing limitation of 10,000 cells. (That limit has been increased to 40,000.)

To be used by the computer, map data must be in numerical (digital) form. The unit boundaries from the geologic map, landslide map, and slope map were first translated into x, y coordinate locations using a CALMA digitizer and its related processing programs. The computer program CELSET (program A, p. 12) converted the coordinate data into grid-cell data, assigning the appropriate values of geology, landslide, and slope to the center of each grid cell. The grid-cell maps were plotted and checked against the originals (pl. 1).

ASSIGNING PRESLIDE SLOPES

We attempted to program the computer to assign preslide slopes to the landslide units, because manually derived preslide slopes were

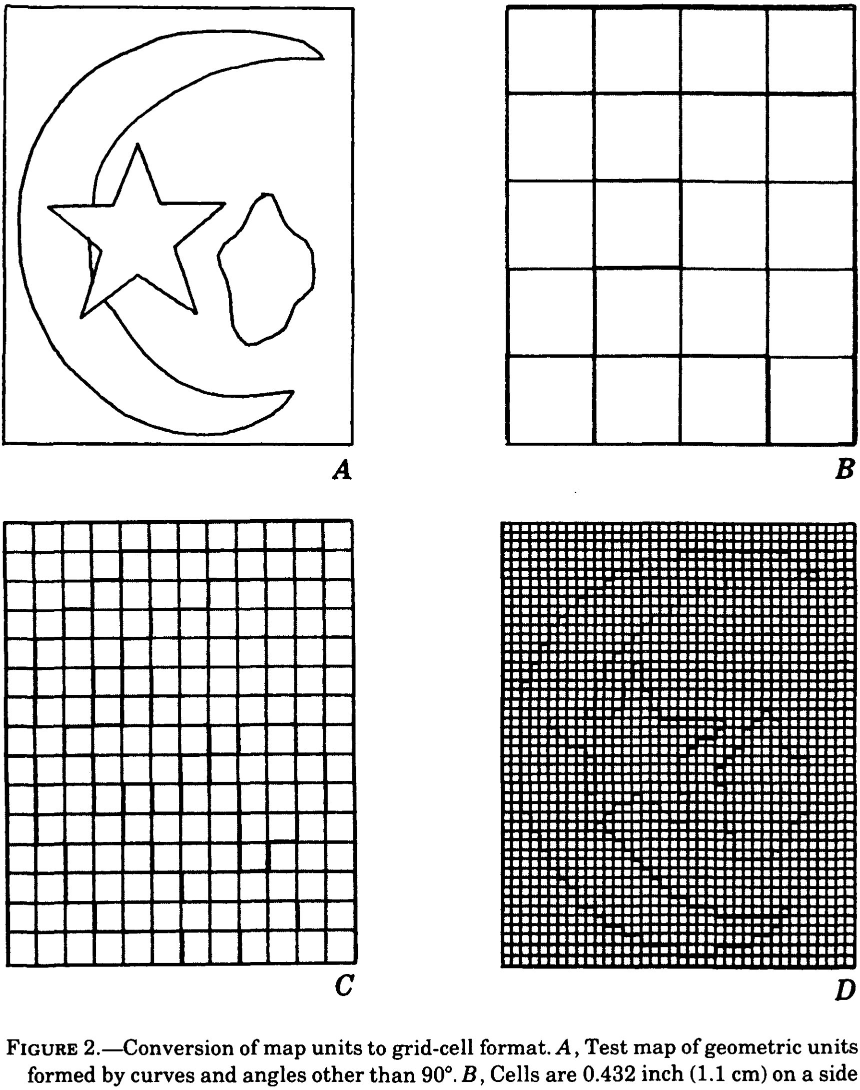

FIGURE 2.—Conversion of map units to grid-cell format. *A*, Test map of geometric units formed by curves and angles other than 90°. *B*, Cells are 0.432 inch (1.1 cm) on a side and illustrate the shortcomings of the method when using too large a cell size. The irregular unit was not assigned a cell because no cell center was within its boundary, even though it covers 1.28 cells. The top point of the crescent was assigned a cell even though it occupies only 0.083 of the cell. The star became a square. *C*, Cells are 0.144 inch (0.366 cm) on a side and show greater detail than *B* but are blocky in appearance. *D*, Cells are 0.048 inch (0.122 cm) on a side, which corresponds to 250 feet at 1:62,500 scale.

used by Brabb, Pampeyan, and Bonilla (1972) to adjust the landslide susceptibility numbers. In general, landslides form steeper slopes in the headwall and toe areas and gentler slopes in the middle as compared to the original surface. The programs tried to determine preslide slope, but the results were unsatisfactory. We have tentatively decided that the geologist can do this part of the operation mre effectively than the computer. Accordingly, we used the matrix developed by Brabb, Pampeyan, and Bonilla (1972) in table 1 to adjust the landslide susceptibility numbers for each slope interval.

ASSIGNING LANDSLIDE SUSCEPTIBILITY NUMBERS

A computer program was written and tested for listing the geologic units with the proportion of their outcrop areas that had failed by landsliding in each slope interval. This subroutine CHARTX (program B, p. 14) (1) totaled the number of grid cells of each geologic unit in each slope interval, (2) totaled the number of grid cells of each geologic unit in each slope interval that had failed by landsliding, (3) computed the percentage of failure, (4) assigned the landslide susceptibility number on the basis of the percent-failure category, and (5) printed a chart of the above totals, percents, and landslide susceptibility categories assigned.

A map produced from these data would be representative of the test area but not of the county as a whole. To compare the manually produced map and the one produced by a computer, we gave the computer the landslide susceptibility numbers for the entire county used by Brabb, Pampeyan, and Bonilla (1972) as input data in the form of a 9×6 matrix. The computer program COMPOS (program B) then (1) located the geologic unit and slope category for each grid cell, (2) assigned each cell the landslide susceptibility number for that geologic unit and slope interval as found in the matrix, and (3) punched cards containing the landslide susceptibility numbers. The resulting grid-cell landslide susceptibility map is shown on plate 1 along with the manually prepared map.

PLOTTING THE DATA

In order to check the digitized input data, the landslide, geologic, and slope grid-cell maps were plotted. Program PLTCEL (program C, p. 19) (1) derived regions of common value by eliminating the boundaries between adjacent cells of equal value, (2) labeled the regions, and (3) punched output cards for use on plotter. The plotting program is versatile in that any map may be plotted within the boundary dimensions of the plotter (29 inches (73.66 cm) in one direction and 110 feet (33.528 m) in the other). We plotted an unlabeled 1:62,500 map for comparison with the manually compiled maps (pl. 1), and a larger, labeled map for readability and ease of checking map labels (fig. 3).

COLOR OUTPUT

The landslide susceptibility data were converted into three magnetic tape files for use on an image recorder. This unit reads the magnetic tape and exposes color film with blue, green, and red filters. Each tape file defines the amount of light needed to pass through the corresponding filter so that a unique color results for each code (pl. 1). The tape can be used with several film types: polaroid for quick-look, color negative, or color positive. Each tape file may also be exposed on separate pieces of color film in order to give the blue, green, and red separates needed in some printing processes. The program CAT2DICO that converted the grid-cell format into the image recorder format was developed by Robert E. Slye of Ames Research Center and is currently in use there.

COST ANALYSIS

The cost of producing the computerized landslide susceptibility map with a 250-foot grid is $500 to $800 for a test area of approximately 15 mi² (39 km²). The figures include digitizer, computer, labor, and overhead expenses.

TABLE 2.—*Time and cost of producing computerized map*

County	Land area (mi²)	Land area (km²)	Minimum cost	Time (months)	Maximum cost	Time (months)
Alameda	733	1,898	$ 24,500	6	$ 39,200	10
Contra Costa	734	1,901	24,500	6	39,200	10
Marin	520	1,347	17,500	4	28,000	6
Napa	758	1,963	25,500	6	40,800	10
San Mateo	454	1,176	15,500	4	24,800	6
Santa Clara	1,302	3,372	43,500	10	69,600	15
Santa Cruz	439	1,137	15,000	4	24,000	6
Solano	827	2,142	27,500	6	44,000	10
Sonoma	1,579	4,090	52,500	11	84,000	18
Total			$246,000	4.75 yr	$393,600	7.5 yr

The estimated amount and times, shown in table 2, apply to nine counties in the San Francisco Bay region and are not meant to be universally valid. They indicate a reasonable range of expected values and assume availability of adequate geologic, landslide, and slope maps.

The wide range of estimated cost is due to uncertainties in the cost of digitizing slope maps and to uncertainties in predicting problems over large areas. The first county to use the computer system should expect the cost to be relatively high on the scale. As experience is gained, the cost and time should be reduced, except in counties where differing rock types require additional factor analysis.

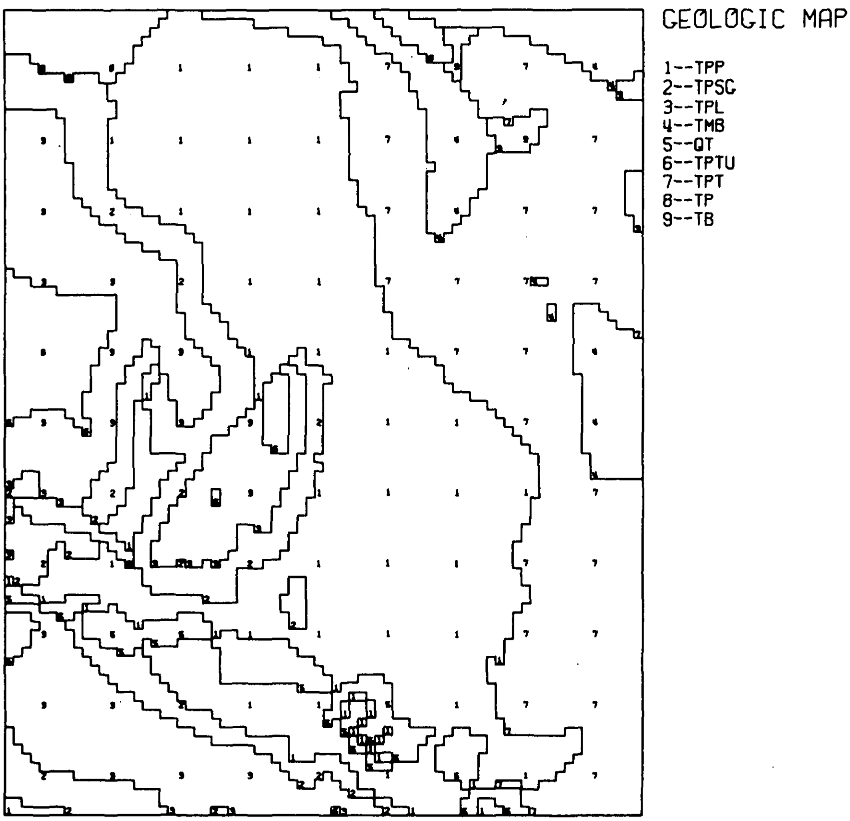

FIGURE 3.—Labeled geologic map of test area drawn by plotter.

CONCLUSIONS

The ability to produce a computerized landslide susceptibility map has been demonstrated. Computerized maps can be generated in approximately the same time as and at less cost than a comparable map compiled manually, and with the following benefits: computer compilation (1) frees the scientist from the drudgery of meticulous labor, allowing more time to concentrate on investigating other factors that relate to landslide susceptibility, (2) eliminates errors in human calculation, assuming the input data and programs have been thoroughly checked before being processed, and (3) creates a data bank for use in future mapping in the same area when additional factors related to landsliding are discovered.

On the other hand, the computer process is not automatic. Considerable judgment is still required from the geologist and the programmer in preparing the map. More testing of larger areas and testing of different geologic terranes are required before this method can be considered reliable, but the results so far are highly encouraging.

REFERENCES CITED

Brabb, E. E., and Pampeyan, E.H., compilers, 1972a, Preliminary geologic map of San Mateo County, California: U.S. Geol. Survey Misc. Field Studies Map MF–328, scale 1:62,500.

———1972b, Preliminary map of landslide deposits in San Mateo County, Caifornia: U.S. Geol. Survey Misc. Field Studies Map MF–344, scale 1:62,500.

Brabb, E. E., Pampeyan, E. H., and Bonilla, M. G., 1972, Landslide susceptibility in San Mateo County, California: U.S. Geol. Survey Misc. Field Studies Map MF–360, scale 1:62,500.

Cummings, J. C., Touring, C. M., and Brabb, E. E., 1962, Geology of the northern Santa Cruz Mountains, California: California Div. Mines and Geology Bull. 181, p. 179–220.

Taylor, F. A., and Brabb, E. E., 1972, Maps showing distribution and cost by counties of structurally damaging landslides in the San Francisco Bay region, California, winter of 1968–69: U.S. Geol. Survey Misc. Field Studies Map MF–327, scale 1:500,000 and 1:1,000,000.

COMPUTER PROGRAMS

PROGRAM A. MAP CONVERSION TO GRID-CELL ROUTINES

Program CELSET and its two related subroutines MASKP and PLYTST are designed to accept (*x*, *y*) coordinate data and related numeric codes that represent map unit boundaries. The output is a matrix of cells to which the codes have been assigned. The program listings below contain many comments to help the reader understand the conversion method.

The programs listed here were created for use on the CDC 7600 computer at the Lawrence Radiation Laboratory in Berkeley, Calif. Some of the code would have to be modified before use on other computer systems. We have tried to indicate where this would be necessary.

```
      PROGRAM CELSET(INPUT,OUTPUT,PUNCH)
      DIMENSION IZ(100,100),POLY(200),EXTRMA(4)
C---------INPUT CELL MAP GEOMETRY-----------
C         MX,MY -- NUMBER OF COLUMNS AND ROWS
C         XMIN,YMIN--COORDINATE OF CELL CENTER IN LOWER LEFT CORNER
C         DELX,DELY--SPACING BETWEEN CELL CENTERS
      READ 1,MX,MY,XMIN,DELX,YMIN,DELY
    1 FORMAT (2I5,4F10.3)
      PRINT 2, MX,MY,XMIN,DELX,YMIN,DELY
    2 FORMAT (1H ,4HMX =,I5,4HMY =,I5,6HXMIN =,F10.3,6HDELX =,F10.3,
     16HYMIN =,F10.3,6HDELY =,F10.3)
C----------INITIALIZE CELL GRID
      DO 10 I=1,MX
      DO 5  J=1,MY
      IZ(I,J) =0
    5 CONTINUE
   10 CONTINUE
C---------INPUT NEXT MAP UNIT BOUNDARY AND CODE
C         NXTRV IS A SITE SPECIFIC ROUTINE USING A LOCAL DIGITIZER FORMAT.
C         POLY(1)=X1
C         POLY(2)=Y1
C         POLY(3)=X2
C         POLY(4)=Y2
C         ETC.
C         NPTS IS NUMBER OF WORDS IN POLY
C         ICODE IS THE MAP UNIT CODE THAT WILL BE GIVEN TO EACH GRID CELL
C         WITHIN THE UNIT BOUNDARY
   20 CALL NXTRV(POLY,NPTS,ICODE)
C--- NO MORE BOUNDARIES WHEN ICODE=0
      IF(ICODE.EQ.0) GO TO 200
C----------CONDITION BOUNDARY FOR PROCESSING
      CALL MASKP(NPTS,POLY)
C---------FIND BOUNDARY COORDINATE MAX  AND MIN
      EXTRMA(1)=POLY(1)
      EXTRMA(2)=POLY(1)
      EXTRMA(3)=POLY(2)
      EXTRMA(4)=POLY(2)
      DO 30 I=2,NPTS
      IF(POLY(I).GT.EXTRMA(1))EXTRMA(1)=POLY(I)
      IF(POLY(I).LT.EXTRMA(2))EXTRMA(2)=POLY(I)
      IF(POLY(I+1).GT.EXTRMA(3))EXTRMA(3)=POLY(I+1)
      IF(POLY(I+1).LT.EXTRMA(4))EXTRMA(4)=POLY(I+1)
   30 CONTINUE
C--------DETERMINE WHICH CELL CENTERS LIE WITHIN THE BOUNDARY AND SET
C         THEM TO ICODE
      DO 100 I=1,MX
      DO 90 J=1,MY
      XTEST = XMIN+FLOAT(I-1)*DELX
      YTEST=YMIN+FLOAT(J-1)*DELY
      CALL PLYTST(IFLAG,NPTS,EXTRMA,POLY,XTEST,YTEST)
      IF(IFLAG.EQ.1) IZ(I,J)=ICODE
   90 CONTINUE
  100 CONTINUE
```

```
C---------PROCESS NEXT BOUNDARY
      GO TO 20
C---------PUNCH GRID CELL CODES
  200 DO 300 J=1,MY
      PUNCH 301,(IZ(I,J),I=1,MX)
  301 FORMAT(40I2)
  300 CONTINUE
      STOP




      END
      SUBROUTINE MASKP(NPTS,POLY)
C---     PURPOSE--
C          CONDITIONS THE REAL VALUES CONTAINED IN THE ARRAY POLY
C          FOR LATER PROCESSING
C          COMPUTER DEPENDENT CODE
      DIMENSION POLY(1)
      DO 10 I = 1,NPTS,2
      POLY(I) = AND(POLY(I),77777777777777777774B)
   10 CONTINUE
      RETURN
      END
      SUBROUTINE PLYTST(IFLAG,NPTS,EXTRMA,POLY,XTEST,YTEST)
C---       PURPOSE--
C               TO DETERMINE IF THE MID-POINT OF A CELL IS INSIDE OR
C               OUTSIDE OF A GIVEN MAP UNIT BOUNDARY.
C          METHOD--
C               IN GENERAL IT CAN BE DETERMINED IF A POINT LIES INSIDE
C               OR OUTSIDE A CLOSED BOUNDARY IF A RAY FROM THE POINT IN
C               QUESTION ENCOUNTERS AN ODD OR EVEN NUMBER OF INTERSECTIONS.
C               THE SUBROUTINE FIRST CHECKS IF THE TEST POINT LIES OUT OF
C               THE RANGE OF THE DATA ITSELF.  IF IT DOES, THEN THE TEST POINT
C               OBVIOUSLY CANNOT BE INSIDE THE BOUNDARY.  IF THE TEST POINT
C               IS FOUND TO LIE WITHIN THE RANGE OF THE DATA, MORE TESTS ARE MADE.
C               THE BOUNDARY POINTS ARE TAKEN IN PAIRS.  THE TEST POINT IS
C               COMPARED TO THE TWO BOUNDARY POINTS TO SEE IF ITS ABSCISSA IS
C               CONTAINED IN THE DOMAIN OF THE BOUNDARY TEST SEGMENT.  IF IT IS
C               OUTSIDE THE DOMAIN, NO FURTHER TESTS ARE MADE AND THE NEXT
C               BOUNDARY POINT IS USED FOR THE NEW LINE SEGMENT.  OTHERWISE THE
C               INTERSECTION OF A VERTICAL RAY AND THE LINE IS CALCULATED.
C               THE Y VALUE OF THIS POINT IS COMPARED WITH THAT OF THE TEST POINT.
C               IF THE Y VALUE OF THE INTERSECTION IS GREATER THAN THAT OF THE
C               TEST POINT THEN A COUNTER IS INCREMENTED TO INDICATE THAT
C               A VALID INTERSECTION HAS BEEN MADE.  OTHERWISE THE INTERSECTION
C               IS NOT COUNTED.  AFTER ALL LINE SEGMENTS HAVE BEEN COMPARED
C               AGAINST THE TEST POINT, INCLUDING  THE LINE SEGMENT FROM THE N-TH
C               POINT TO THE FIRST, THE COUNT IS EVALUATED.  IF THE NUMBER OF
C               INTERSECTIONS IS ODD THE POINT IS INSIDE THE BOUNDARY AND IFLAG
C               IS ASSIGNED THE VALUE OF 1.  IF THE NUMBER OF INTERSECTIONS IS
C               EVEN, THE POINT LIES OUTSIDE THE BOUNDARY AND IFLAG=0.
C---       INPUT PARAMETERS--
C               NPS--  THE NUMBER OF DATA POINTS THAT DEFINE THE BOUNDARY
C               EXTRMA--  A LINEAR FOUR ELEMENT ARRAY THAT CONTAINS THE
C                         MAXIMA AND MINIMA OF THE X AND Y DATA THAT DEFINES
C                         THE BOUNDATY.  EXTRMA(1)=XMAX, EXTRMA(2)=XMIN
C                         EXTRMA(3)=YMAX, EXTRMA(4)=YMIN.
C               POLY-- THE ARRAY WHICH HOLDS THE ACTUAL X AND Y VALUES OF THE
C                      DATA POINTS.  NOTE THAT THE DIMENSION IS ONLY 1, INDICATING
C                      A VARIABLE NUMBER OF ELEMENTS WILL BE USED EACH TIME THE
C                      SUBROUTINE IS USED.
C               XTEST-- THE X COORDINATE OF THE TEST POINT.
C               YTEST-- THE Y COORDINATE OF THE TEST POINT.
C---       OUTPUT PARAMETER--
C               IFLAG-- IFLAG IS RETURNED TO INDICATE IF THE POINT IS INSIDE OR
C                       OUTSIDE THE BOUNDARY.  A VALUE OF 1 INDICATES THAT THE
C                       POINT IS INSIDE, 0 INDICATES THAT THE POINT IS OUTSIDE
      DIMENSION EXTRMA(4),POLY(1)
      IFLAG = 0
      INSECT = 0
      XP =XTEST
      YP = YTEST
```

```
      MAX = NPTS + 1
C---      CHECK TO SEE IF POINT IS WITHIN DATA BOX SPECIFIED BY EXTRMA.
      IF(XP.LT.EXTRMA(2) .OR. XP.GT.EXTRMA(1)) RETURN
      IF(YP.LT.EXTRMA(4) .OR. YP.GT.EXTRMA(3)) RETURN
      XP = OR(XP,3B)
      YP = OR(YP,3B)
C---      SET UP LOOP TO COMPARE TEST POINT AGAINST BOUNDARY SEGMENTS, BUT FIRST
C---      SAVE INFORMATION IN ARRAY TO BE WRITTEN OVER.
      XSAVE = POLY(MAX)
      YSAVE = POLY(MAX + 1)
      POLY(MAX) = POLY(1)
      POLY(MAX+1) = POLY(2)
      DO 20  I = 3,MAX,2
      X1 = POLY(I-2)
      Y1 = POLY(I-1)
      X2 = POLY(I)
      Y2 = POLY(I+1)
      IF(X1 .LT. X2) GO TO 22
      X1 = POLY(I)
      Y1 = POLY(I+1)
      X2 = POLY(I-2)
      Y2 = POLY(I-1)
C---      CHECK TO SEE IF TEST POINT FALLS WITHIN THE DOMAIN
C         OF THE SEGMENT.  IF NOT, GO TO THE SEGMENT
   22 IF (XP.LT.X1 .OR. XP.GT.X2) GO TO 20
C---      CALCULATE THE Y COORDINATE OF THE INTERSECTION
      YSECT = (Y2 - Y1)/(X2 - X1)*(XP - X1) + Y1
C---      COMPARE THE Y COORDINATE OF THE INTERSECTION WITH THE COORDINATE OF
C         THE TEST POINT.  IF THE TEST POINT IS ABOVE THE INTERSECTION, GO TO
C         THE NEXT SEGMENT.  IF EQUAL, RETURN.  IF BELOW, INCREMENT THE COUNTER
C         INSECT.
      IF(YP - YSECT) 26,24,20
   24 IFLAG = 1
      GO TO 99
   26 INSECT = INSECT + 1
   20 CONTINUE
      IFLAG = AND(INSECT,1B)
   99 POLY(MAX) = XSAVE
      POLY(MAX + 1) = YSAVE
      RETURN
      END
```

PROGRAM B. LANDSLIDE SUSCEPTIBILITY CATEGORY ASSIGNMENT ROUTINES

Program COMPOS uses slope, landslide, and geologic grid-cell data as input. It combines the data to produce landslide susceptibility categories as output in punched card form. The DATA statement contains the matrix data mentioned in the text and would be deleted or at least modified for future jobs.

Subroutine CHARTX produces a tabular output of the assignments to landslide susceptibility categories and area totals for each geologic type and slope category. The table at the end of CHARTX is an example. The data were used only to test the program. We checked the totals using a planimeter, and in each case the category assignment agreed with the computer.

```
C     PROGRAM COMPOS(INPUT,OUTPUT,TAPE1)
C
C--------COMPOS VERSION 1.0---------------------------------------------
C
C---     PURPOSE --
C               COMPOS DOES THE FOLLOWING --
C                1. INPUTS A PARAMETER CARD,SLOPE CLASS DECK, LANDSLIDE
C                   CLASS DECK, GEOLOGICAL UNIT DECK AND A TITLE CARD
C                2. COMPUTES THE LANDSLIDE SUSCEPTIBILITY CLASSES
C                3. PUNCHES A DECK OF THE LANDSLIDE SUSCEPTIBILITY CLASSES
```

```
C               4. PRINTS TWO SUMMARY TABLES (SQ. MILES AND SQ. KILOMETERS) ONLY
C                  IN CASES WITHOUT MATRIX INPUT.
C---      INPUT DECK SETUP
C              1.  CONTROL CARDS
C              2.  PROGRAM
C              3.  PARAMETER CARD
C                    PARM    RMXCOL    RMYROW    SIDLEN
C                    (A4,6X,4F10.2)
C                         RMXCOL -- NUMBER OF COLUMNS IN CELL GRID
C                         RMYROW -- NUMBER OF ROWS IN CELL GRID
C                         SIDLEN -- LENGTH (IN FEET -- I.E. 250) OF A
C                                CELL SIDE
C              4.  SLOPE CLASS DECK
C              5.  LANDSLIDE CLASS DECK
C              6.  GEOLOGICAL UNITS DECK
C              7.  STOP      (A4 FORMAT)
C              8.  TITLE  (7A10 FORMAT)
C
C-----------------------------------------------------------------------
      INTEGER ISC(100,100),IL(100,100),IG(100,100),ITYPE(9,6)
      INTEGER KDATA(3,9,6)
      DATA ITYPE /11*1,2,3*1,2,2,1,2,1,2,2,1,3*3,2,5,4,6,3,2,5,5,4,3,
     1 5,5,6,3,2,5,5,4,4,5,5,6,4,2,5,5,4,4/
      REAL RTOT(9,6),RFAIL(9,6)
      INTEGER ITITL(7)
C---      INPUT PARAMETER CARD
      READ 3,IPARM,RMXCOL,RMYROW,SIDLEN
    3 FORMAT(A4,6X,3F10.2)
      PRINT 4,IPARM,RMXCOL,RMYROW,SIDLEN
    4 FORMAT(1H ,A4,6X,6HRMXCOL,F10.2,2X,6HRMYROW,F10.2,2X,6HSIDLEN,
     1 F10.2)
      IMAX = RMXCOL
      JMAX = RMYROW
C---      MM IS THE FIRST DIMESNION OF ISC,IL AND IG
      MM = 100
C---      INITIALIZE TOTALS ARRAYS
      DO 10 I = 1,9
      DO 5 J = 1,6
      RTOT(I,J) = 0.
      RFAIL(I,J) = 0.
    5 CONTINUE
   10 CONTINUE
C         INPUT SLOPE CLASSES
      DO 15 J = 1,JMAX
      READ 1,(ISC(I,J),I=1,IMAX)
    1 FORMAT(74I1)
   15 CONTINUE
C         INPUT LANDSLIDE CLASSES
      DO 20 J = 1,JMAX
      READ 1,(IL(I,J),I=1,IMAX)
   20 CONTINUE
C         INPUT GEOLOGICAL UNITS
      DO 30 J = 1,JMAX
      READ 1,(IG(I,J),I=1,IMAX)
   30 CONTINUE
C         READ CHECK STOP
      READ 2,ISTOP
    2 FORMAT(A4)
      IF(ISTOP.NE.4HSTOP) STOP 1111
C---      INPUT TITLE CARD
      READ 31,ITITL
   31 FORMAT(7A10)
C         COMPUTES TOTAL CELLS HAVING SAME GEOLOGY AND SLOPE, AND SUBTOTALS
C           THOSE CELLS THAT HAVE FAILED BY LANDSLIDING
C         INSERT A GO TO 280 CARD HERE WHEN THE ITYPE MATRIX IS USED
      DO 100 I = 1,IMAX
      DO 90 J = 1,JMAX
      IX = IG(I,J)
      JY = ISC(I,J)
      RTOT(IX,JY) = RTOT(IX,JY)+1.
      IF(IL(I,J).EQ.1) RFAIL(IX,JY) = RFAIL(IX,JY)+1.
   90 CONTINUE
  100 CONTINUE
```

```
  100 CONTINUE
C         COMPUTE LANDSLIDE SUSCEPTIBILITY CATEGORIES FOR CELLS
      DO 200 I = 1,9
      DO 190 J = 1,6
      IF(RTOT(I,J).EQ.0.0) GO TO 190
      PRCNT = RFAIL(I,J)/RTOT(I,J)
      IF(PRCNT.GT.0.01) GO TO 110
      ITYPE(I,J) = 1
      GO TO 190
  110 IF(PRCNT.GT.0.08) GO TO 120
      ITYPE(I,J) = 2
      GO TO 190
  120 IF(PRCNT.GT.0.25) GO TO 130
      ITYPE(I,J) = 3
      GO TO 190
  130 IF(PRCNT.GT.0.42) GO TO 140
      ITYPE(I,J) = 4
      GO TO 190
  140 IF(PRCNT.GT.0.53) GO TO 150
      ITYPE(I,J) = 5
      GO TO 190
  150 IF(PRCNT.GT.0.70) GO TO 160
      ITYPE(I,J) = 6
      GO TO 190
  160 ITYPE(I,J) = 7
  190 CONTINUE
  200 CONTINUE
  280 DO 300 I = 1,IMAX
      DO 290 J = 1,JMAX
      IX = IG(I,J)
      IY = ISC(I,J)
      ISC(I,J) = ITYPE(IX,IY)
      IF(IL(I,J).EQ.1) ISC(I,J) = 7
  290 CONTINUE
  300 CONTINUE
C         PUNCH FINAL RESULTS
      DO 400 J = 1,JMAX
      WRITE (1,1) (ISC(I,J),I=1,IMAX)
  400 CONTINUE
C         INSERT A GO TO 99  CARD HERE WHEN THE ITYPE MATRIX IS USED
      DO 500 J = 1,6
      DO 405 I=1,9
      KDATA(3,I,J) = ITYPE(I,J)
  405 CONTINUE
      DO 410 I=1,9
      KDATA(1,I,J) = RTOT(I,J)
  410 CONTINUE
      DO 420 I=1,9
      KDATA(2,I,J) = RFAIL(I,J)
  420 CONTINUE
  500 CONTINUE
      CALL CHARTX(KDATA,SIDLEN,1,ITITL)
      CALL CHARTX(KDATA,SIDLEN,2,ITITL)
   99 CONTINUE
      STOP
      END
```

```
      SUBROUTINE CHARTX(KDATA,SIDLEN,KLOFLG,T)
C
C--------CHARTX VERSION 1.1 APRIL 1 1975 ----------------------------
C
C---      PURPOSE --
C              CHARTX PRINTS THE SUMMARY TABLE
C
C---      INPUT PARAMETERS --
C              KDATA -- 3 DIM ARRAY CONTAINING TOTAL AREA,AREA FAIL,
C                   AND TYPE FOR EACH GEOLOGICAL TYPE AND SLOPE CLASS
C              SIDLEN -- LENGTH (IN FEET -- I.E. 250) OF A CELL SIDE
C              KLOFLG -- SQ. MILES OR SQ. KILOMETERS FLAG
C                   =1 PRINT TABLE IN SQ. MILES
C                   =2 PRINT TABLE IN SQ. KILOMETERS
C              T -- TABLE TITLE (7A10 FORMAT)
C
C---      OUTPUT PARAMETERS --
C              A TABLE IS PRINTED.  NOTE THAT THE DATA AND FORMAT STATEMENTS
C              ARE COMPUTER DEPENDENT
C
```

```
C--------------------------------------------------------------------------
      INTEGER IAREA(2)
      DATA IAREA/10HMILES     ,10HKILOMETERS/
      INTEGER T(7),S(7),R(9),ROW(4),ROW2(4),KDATA(3,9,6)
      REAL RDATA(3,9,7)
      DATA S/10H       0-5,10H      6-15,10H     16-30,10H     31-50,
     110H     51-70,10H       70+,10H     TOTAL/
      DATA R/5H  TPP,5H TPSG,5H  TPL,5H  TMB,5H   QT,5H TPTU,5H  TPT,5H
     1  TP,5H   TB/
      DATA ROW/8H      TO,8H    FAIL,8H  PERCEN,8H  SUSCEP/
      DATA ROW2/8HTAL AREA,8HURE AREA,8HT FAILED,8HTIBILITY/
      FACT = (SIDLEN/5280.)**2
      SQKILO = 1.60935**2
      IF(KLOFLG.EQ.2) FACT = FACT*SQKILO
      DO 120 K = 1,6
      DO 110 J = 1,9
      DO 100 I = 1,2
      RDATA(I,J,K) = FLOAT(KDATA(I,J,K))*FACT
  100 CONTINUE
  110 CONTINUE
  120 CONTINUE
      DO 210 K = 1,6
      DO 200 J = 1,9
      IF(RDATA(1,J,K).EQ.0.0) GO TO 190
      RDATA(3,J,K) = RDATA(2,J,K)/RDATA(1,J,K)*100.
      GO TO 200
  190 RDATA(3,J,K) = 0.0
  200 CONTINUE
  210 CONTINUE
      DO 240 J = 1,9
      RDATA(1,J,7) = 0.0
      RDATA(2,J,7) = 0.0
      DO 230 K = 1,6
      RDATA(2,J,7) = RDATA(2,J,7) + RDATA(2,J,K)
      RDATA(1,J,7) = RDATA(1,J,7)+RDATA(1,J,K)
  230 CONTINUE
      RDATA(3,J,7) = RDATA(2,J,7)/RDATA(1,J,7)*100.
  240 CONTINUE
      PRINT 55,T,IAREA(KLOFLG)
   55 FORMAT(1H1,7A10,15HAREA IN SQUARE ,A10)
       PRINT 7
    7 FORMAT(1X,13HGEOLOGIC UNIT,40X,25HSLOPE INTERVALS (PERCENT),/)
      PRINT 27
   27 FORMAT(1H )
      PRINT 8,S
    8 FORMAT(24X,7(A10,3X),//)
      DO 30 I = 1,9
      DO 20 J = 1,3
      IJK =5H
C      PRINTS GEOLOGIC UNIT NAME IN SECOND ROW
      IF(J.EQ.2) IJK = R(I)
      PRINT 15,IJK,ROW(J),ROW2(J),(RDATA(J,I,K),K=1,7)
   15 FORMAT(A5,1X,2(A8),5X,7(F8.4,5X))
   20 CONTINUE
      PRINT 16,IJK,ROW(4),ROW2(4),(KDATA(3,I,K),K=1,6)
   16 FORMAT(A5,1X,2(A8),5X,7(I4,9X))
      PRINT 26
   26 FORMAT(1H )
  30  CONTINUE
      PRINT 40
   40 FORMAT(1H1)
      RETURN
      END
```

LA HONDA QUADRANGLE TEST - AREA IN SQUARE KILOMETERS

GEOLOGIC UNIT		0-5	6-15	16-30	31-50	51-70	70+	COMPUTER TOTAL	PLANIMETER TOTAL
		SLOPE INTERVALS (PERCENT)							
TPP	TOTAL AREA	.0290	2.5258	3.4607	4.2039	2.0207	1.2252	13.4652	13.914
	FAILURE AREA	0.0000	1.0510	1.8581	1.9974	.6039	.1800	5.6904	5.784
	PERCENT FAILED	0.0000	41.6092	53.6913	47.5138	29.8851	14.6919	42.2596	41.57
	SUSCEPTIBILITY	1	4	6	5	4	3	IV	IV
TPSG	TOTAL AREA	0.0000	.3948	.9987	1.8465	.7897	.2613	4.2910	4.621
	FAILURE AREA	0.0000	.2090	.4936	.9348	.2729	.0929	2.0032	2.10
	PERCENT FAILED	0.0000	52.9412	49.4186	50.6289	34.5588	35.5556	46.6847	45.44
	SUSCEPTIBILITY	0	5	5	5	4	4	V	V
TPL	TOTAL AREA	0.0000	1.2368	1.8348	2.0207	.6794	.1916	5.9633	6.172
	FAILURE AREA	0.0000	.8419	1.1090	1.0626	.3890	.1335	3.5361	3.686
	PERCENT FAILED	0.0000	68.0751	60.4430	52.5862	57.2650	69.6970	59.2989	59.72
	SUSCEPTIBILITY	0	6	6	5	6	6	VI	VI
TMB	TOTAL AREA	0.0000	.1045	.4355	.9813	.3774	.3716	2.2703	2.448
	FAILURE AREA	0.0000	.0523	.2439	.3310	.1684	.0697	.8652	1.041
	PERCENT FAILED	0.0000	50.0000	56.0000	33.7278	44.6154	18.7500	38.1074	42.52
	SUSCEPTIBILITY	0	5	6	4	5	3	IV	IV
QT	TOTAL AREA	.1161	1.1845	.0987	.0348	.0058	0.0000	1.4400	1.471
	FAILURE AREA	0.0000	.0232	.0058	0.0000	0.0000	0.0000	.0290	.0725
	PERCENT FAILED	0.0000	1.9608	5.8824	0.0000	0.0000	0.0000	2.0161	4.93
	SUSCEPTIBILITY	1	2	2	1	1	0	II	II
TPTU	TOTAL AREA	0.0000	.2961	.3774	.2729	.1568	.0174	1.1207	1.207
	FAILURE AREA	0.0000	.0755	.1510	.1103	.0465	.0174	.4006	.495
	PERCENT FAILED	0.0000	25.4902	40.0000	40.4255	29.6296	100.0000	35.7513	40.81
	SUSCEPTIBILITY	0	4	4	4	4	7	IV	IV
TPT	TOTAL AREA	0.0000	1.4284	2.8800	3.4955	1.0336	.4181	9.2555	9.247
	FAILURE AREA	0.0000	.9581	1.7071	1.9103	.5284	.2090	5.3129	5.512
	PERCENT FAILED	0.0000	67.0732	59.2742	54.6512	51.1236	50.0000	57.4028	59.61
	SUSCEPTIBILITY	0	6	6	6	5	5	VI	VI
TP	TOTAL AREA	0.0000	.2845	.2381	.3658	.0406	.0523	.9813	.875
	FAILURE AREA	0.0000	.1974	.2032	.2381	.0348	.0523	.7258	.707
	PERCENT FAILED	0.0000	69.3878	85.3659	65.0794	85.7143	100.0000	73.9645	80.80
	SUSCEPTIBILITY	0	6	7	6	7	7	L	L
TB	TOTAL AREA	0.0000	.0755	.0987	.1045	.0348	0.0000	.3136	.267
	FAILURE AREA	0.0000	.0581	.0813	.0929	.0174	0.0000	.2497	.266
	PERCENT FAILED	0.0000	76.9231	82.3529	88.8889	50.0000	0.0000	79.6296	99.62
	SUSCEPTIBILITY	0	7	7	7	5	0	L	L

PROGRAM C. PLOTTING ROUTINES

Program PLTCEL and its related subroutines CELPLT, CFOLLW, CHKCEL, and DGBITR are utilized in conjunction with local installation Calcomp plotting routines (SYMBOL and PLOT) to produce a labeled plot like the one shown in figure 3. Comments within the listings describe the method by which grid cell data are converted to common regions and plotted.

```
      PROGRAM PLTCEL(INPUT,OUTPUT,TAPE99)
C---------PLTCEL VERSION 1.0-----------------------------------------------
C
C---      PURPOSE --
C              PLTCEL DRAWS A CELL MAP WITH AN OPTIONAL IDENTIFICATION
C              KEY DRAWN TO THE UPPER RIGHT OF THE MAP
C
C---      DECK SETUP --
C              1. CONTROL CARDS
C              2. PROGRAM
C              3. PARAMETER CARD
C                 PARM    RMXCOL    RMYROW  DEL   CHRHGT
C                 (A4,6X,4F10.2) FORMAT
C              4. CELL INPUT DECK
C                 ROW 1 TO ROW MY READ IN 80A1 FORMAT
C              5. STOP
C                 (A4 FORMAT)
C              6. UP TO 30 TITLE CARDS
C                 (5A10 FORMAT)
C              7. STOP
C                 (A4 FORMAT)
C
C---      REVISIONS --
C
C-------------------------------------------------------------------------
      INTEGER IZ(150,150)
      REAL PARM(6)
      REAL S(12)
      INTEGER ITITL(6)
      CALL PLOTS(DUM,DUM)
      READ 4,IPARM,RMXCOL,RMYCOL,DEL,CHRHGT
    4 FORMAT(A4,6X,4F10.2)
      PRINT 5,IPARM,RMXCOL,RMYCOL,DEL,CHRHGT
    5 FORMAT(1H ,A4,6X,6HRMXCOL,F10.2,2X,6HRMYCOL,F10.2,2X,6H   DEL,
     1 F10.2,2X,6HCHRHGT,F10.2)
      MX = RMXCOL
      MY = RMYCOL
C---      INPUT ARRAY
      DO 10 J = 1,MY
      READ 1,(IZ(I,J),I=1,MX)
    1 FORMAT(80A1)
   10 CONTINUE
      READ 2,ISTOP
    2 FORMAT(A4)
      IF(ISTOP.NE.4HSTOP) STOP 1111
C---      PLOT CELL MAP
   55 CONTINUE
      PARM(1) = DEL/2.
      PARM(2) = DEL/2.
      PARM(3) = DEL
      PARM(4) = DEL
      PARM(5) = CHRHGT
      PARM(6) = 1.
      CALL CELPLT(IZ,MX,MY,150,PARM)
      PRINT 777
  777 FORMAT(1H ,16HCELL MAP PLOTTED)
C---      PLOT TITLES
      CHRHGT = .21
      XPOS = PARM(1)+FLOAT(MX)*DEL+.25
      YPOS = PARM(2)+FLOAT(MY)*DEL-CHRHGT
```

```
      DO 200 KK = 1,30
      READ 77,(ITITL(I),I=1,5)
   77 FORMAT(5A10)
      ITITL(6) = 0
      IF(ITITL(1).EQ.4HSTOP) GO TO 300
      CALL SYMBOL(XPOS,YPOS,CHRHGT,ITITL,0.,50)
         PRINT 778,ITITL
  778 FORMAT(1H ,6A10)
      YPOS = YPOS-CHRHGT*1.5
      CHRHGT = .14
  200 CONTINUE
  300 CALL PLOT(0.,0.,999)
      STOP
      END
      SUBROUTINE CELPLT(IZ,MX,MY,MM,PARM)
C
C---------CELPLT VERSION 1.0---------------------------------------------
C
C---     PURPOSE --
C             CELPLT PLOTS AND LABELS A RECTANGULAR MAP CONSISTING OF
C             CELLS  AND THEIR VALUES
C
C             THE CELLS ARE REPRESENTED AS A TWO-DIMENSIONAL ARRAY
C             IZ(-,-), EACH OF WHOSE ELEMENTS CONTAINS A VALUE.
C             THE VALUES ARE ALPHA-NUMERIC DATA THAT ARE ALSO USED ON
C             THE MAP AS CHARACTER STRINGS TO IDENTIFY THE CELLS.
C
C             CELLS OF EQUAL VALUE ARE ARRANGED INTO REGIONS OF EQUAL
C             VALUE BY DRAWING BOUNDARIES BETWEEN CELLS OF UNEQUAL VALUE
C             THE REGIONS CONSIST OF ONE OR MORE CELLS AND ARE LABELLED
C             WITH ALL OR PART OF THE CHARACTER STRING CARRIED AS THE
C             APPROPRIATE VALUE.
C             THE LABELS ARE PLACED IN LOCAL LOWER LEFT CORNERS OF EACH
C             REGION AND AT STANDARD POSITIONS OVER THE ENTIRE MAP.
C             A CORNER LABEL IS OMITTED IF IT IS ADJACENT TO A STANDARD
C             POSITION LABEL IN THE SAME REGION.
C
C---     OPERATION--
C             CELPLT FIRST DRAWS THE BOUNDARIES TO MAKE REGIONS AND THEN
C             LABELS THE REGIONS.
C             THE BOUNDARIES ARE DRAWN BY FINDING A BOUNDARY SEGMENT AND
C             THEN CALLING SUBROUTINE CFOLLW.
C                CFOLLW DRAWS THAT SEGMENT IF IT HAS NOT ALREADY BEEN
C             DRAWN AND THEN CONTINUES TO DRAW THE BOUNDARY OF WHICH THE
C             SEGMENT WAS A PART.  IF A BOUNDARY SEGMENT IS FOUND AND
C             CFOLLW FINDS THAT IT HAS ALREADY BEEN DRAWN AS THE
C             CONTINUATION OF A PREVIOUSLY FOUND SEGMENT, CFOLLW RETURNS
C             CONTROL TO CELPLT.
C             COMPLETENESS IS GUARANTEED BY RUNNING THROUGH ALL THE
C             CELLS  TWICE, FIRST CHECKING TO THE RIGHT OF EACH CELL FOR
C             A BOUNDARY AND THE SECOND TIME CHECKING ABOVE EACH CELL.
C
C---     INPUT PARAMETERS --
C             IZ -- TWO-DIMENSIONAL ARRAY OF VALUES IN ALPHA-NUMERIC
C                  FORM--A MATRIX OR COLLECTION OF CELLS OF MM COLUMNS
C                  AND MY ROWS.
C             MX -- NUMBER OF COLUMNS OR X-INCREMENTS OF MATRIX THAT
C                  ARE ACTUALLY TO BE PROCESSED.
C             MY -- SECOND DIMENSION OF IZ AND NUMBER OF ROWS OR Y-
C                  INCREMENTS OF MATRIX TO BE PROCESSED.
C             MM -- FIRST DIMENSION OF IZ (MAY EXCEED MX).
C             PARM -- PARAMETER ARRAY
C                  PARM(1) - XOFF - DISTANCE FROM PLOTTER REFERENCE
C                        POINT (LOWER LEFT CORNER) TO LOWER LEFT CORNER
C                        OF CELL MAP
C                  PARM(2) - YOFF - SAME AS XOFF BUT IN Y DIRECTION
C                  PARM(3) - DELX - WIDTH (INCHES) OF CELL
C                  PARM(4) - DELY - HEIGHT (INCHES) OF CELL
C                  PARM(5) - CHRHGT - CHARACTER HEIGHT (INCHES)
C                  PARM(6) - NUMCHR - NUMBER OF CHARACTERS IN LABELS
C                        (LEFTMOST NUMCHR CHARACTERS WILL BE PLOTTED)
C
```

```
C---      OUTPUT PARAMETERS --
C              A CELL MAP WITH BORDER AND LABELS IS DRAWN
C
C---      NOTE --
C              CELPLT CALLS DIRECTLY AND INDIRECTLY--
C              SUBROUTINES CHKCEL, CFOLLW, AND DGBITR
C
C              AS CELPLT USES STANDARD CALCOMP CALLS IT REQUIRES
C                   CALL PLOTS(DUM,DUM) AT BEGINNING
C                   CALL PLOT(15.,0.,999) AT THE END
C
C---      REVISIONS --
C
C-----------------------------------------------------------------------
      INTEGER IZ(MM,MY)
      COMMON /DGUAA1/ XOF,YOF,DX,DY,IW1(400),IW2(400)
      COMMON /DGUAA2/ IMODX,IMODY
      COMMON /ZZ1/ IFACT
      REAL PARM(6)
      XOFF = PARM(1)
      YOFF = PARM(2)
      DELX = PARM(3)
      DELY = PARM(4)
      CHRHGT = PARM(5)
      NUMCHR = PARM(6)
      RN = NUMCHR
C  WID IS CHARACTER WIDTH AND IS DERIVED FROM THE PLOTTED CHARACTER SET.
      WID = RN*CHRHGT*4./7.+(RN-1.)*CHRHGT*3./7.
      XDEL = (DELX-WID)/2.
      YDEL = (DELY-CHRHGT)/2.
      DO 10 I = 1,400
      IW1(I) = 0
      IW2(I) = 0
   10 CONTINUE
      XOF = XOFF
      YOF = YOFF
      DX = DELX
      DY = DELY
C---      PLOT BORDER
      XL = DELX*FLOAT(MX)
      YL = DELY*FLOAT(MY)
      CALL PLOT(XOFF,YOFF,3)
      CALL PLOT(XOFF+XL,YOFF,2)
      CALL PLOT(XOFF+XL,YOFF+YL,2)
      CALL PLOT(XOFF,YOFF+YL,2)
      CALL PLOT(XOFF,YOFF,2)
      MXM = MX-1
      MYM = MY-1
C---      PROCESS COLUMN LINES
C  THIS RUNS THROUGH THE CELLS, INCREMENTING THE ROWS (Y) AFTER EACH RUN
C  THROUGH ALL THE COLUMNS (X).  IF THE RIGHT EDGE OF A CELL IS A
C  BOUNDARY,  A CALL TO CFOLLW IS GENERATED.
      YPOS = YOFF
      DO 100 J = 1,MY
      XPOS = XOFF+DELX
      DO 50 I = 1,MXM
      IF(IZ(I,J).EQ.IZ(I+1,J)) GO TO 40
C         PLOT COLUMN LINES
      CALL CFOLLW(IZ,MX,MY,MM,I,J,1)
   40 XPOS = XPOS+DELX
   50 CONTINUE
      YPOS = YPOS+DELY
  100 CONTINUE
C---      PROCESS ROW LINES
C  THIS RUNS THROUGH THE CELLS, INCREMENTING THE COLUMNS (X) AFTER EACH
C  RUN THROUGH ALL THE ROWS (Y).  IF THE TOP EDGE OF A CELL IS A
C  BOUNDARY,  A CALL TO CFOLLW IS GENERATED.
      XPOS = XOFF
      DO 200 I = 1,MX
      YPOS = YOFF+DELY
      DO 150 J = 1,MYM
      IF(IZ(I,J).EQ.IZ(I,J+1)) GO TO 140
```

```
C         PLOT ROW LINE
      CALL CFOLLW(IZ,MX,MY,MM,I,J,2)
  140 YPOS = YPOS+DELY
  150 CONTINUE
      XPOS = XPOS+DELX
  200 CONTINUE
C         PLOT TITLES
      YPOS = YOFF+YDEL
C---      IF ZERO HEIGHT LABELS -- DON"T BOTHER DRAWING LABELS
      IF(CHRHGT.LE.0.0) RETURN
C  THESE DEFINE STANDARD LABEL POSITIONS.
      INTX = 4
      INTY = 4
      IDELX = 8
      IDELY = 8
      DO 300 J = 1,MY
      XPOS = XOFF+XDEL
      IMODX = 3
      IMODY = 3
      CALL CHKCEL(IZ,MX,MY,MM,1,J,XPOS,YPOS,CHRHGT,NUMCHR)
      XPOS = XPOS+DELX
C  BASIC ALGORITHM OF 250 DO-LOOP--
C    IF THE CELL IS A STANDARD LABEL POSITION
C      THEN PLOT LABEL
C      ELSE IF THE CELL TO THE LEFT HAS A DIFFERENT VALUE
C         THEN CALL CHKCEL
C    GO TO NEXT CELL
      DO 250 I = 2,MX
      ZTST = IZ(I,J)
      IMODX = MOD(I-INTX,IDELX)
      IMODY = MOD(J-INTY,IDELY)
      IF((IMODX.EQ.1).AND.(IMODY.EQ.1)) GO TO 230
      IF(IZ(I,J).EQ.IZ(I-1,J)) GO TO 240
C         BOUNDARY SO CHECK
      CALL CHKCEL(IZ,MX,MY,MM,I,J,XPOS,YPOS,CHRHGT,NUMCHR)
      GO TO 240
  230 CALL SYMBOL(XPOS,YPOS,CHRHGT,IZ(I,J),0.,NUMCHR)
  240 XPOS = XPOS+DELX
  250 CONTINUE
      YPOS = YPOS+DELY
  300 CONTINUE
      RETURN
      END
      SUBROUTINE CFOLLW(IZ,MX,MY,MM,II,JJ,ITYPE)
C
C--------CFOLLW VERSION 1.0--------------------------------------------
C
C---      PURPOSE --
C              CFOLLW FOLLOWS AND PLOTS BOUNDARY LINES
C              A BOUNDARY LINE BETWEEN DIFFERENT CELL TYPES IS FOLLOWED
C              AND DRAWN SEGMENT BY SEGMENT UNTIL IT ENCLOSES A REGION BY
C              REACHING A BOUNDARY SECTION ALREADY DRAWN.  EACH SEGMENT
C              AFTER THE FIRST IS A CONTINUATION OF A SEGMENT AND IS BY
C              DEFINITION A BOUNDARY BETWEEN DIFFERENT CELL TYPES.  THE
C              CONTINUATION CAN ONLY BE IN ONE OF THREE DIRECTIONS.
C
C         OPERATION--
C              THE CENTRAL ALGORITHM FINDS AN UNDRAWN CONTINUATION OF THE
C              CURRENT SEGMENT, IF ONE EXISTS.
C              THE ALGORITHM IS IMPLEMENTED IN FOUR VARIATIONS, WHICH
C              DEPEND ON THE FORM OF THE LAST PREVIOUS SEGMENT AND
C              WHICH START AT LABELS 200, 300, 400, 500
C
C              THE FIRST SEGMENT OF EACH SEQUENCE OF SEGMENTS IS
C              GENERATED AS A SPECIAL CASE, PRECEDING LABEL 100, FROM
C              A BOUNDARY FOUND IN THE CALLING ROUTINE AND SPECIFIED BY
C              PARAMETERS II, JJ, ITYPE.
C
C              A RECORD OF WHICH BOUNDARY SEGMENTS HAVE BEEN DRAWN IS
C              KEPT IN ARRAYS IW1(-) AND IW2(-).
C              THE STATUS OF THE RIGHT HAND BOUNDARY OF EACH CELL(I,J)
C              IS REPRESENTED BY A BIT IN ARRAY IW1(-).
```

```
C              THE STATUS OF THE UPPER BORDER OF EACH CELL(I,J) IS
C              REPRESENTED BY A BIT IN ARRAY IW2(-).
C              INITIALLY ALL BOUNDARIES ARE OPEN AND ALL BITS ARE 0
C              WHEN A BOUNDARY IS CLOSED (DRAWN), IT IS MARKED BY A 1 IN
C              THE APPROPRIATE BIT.
C              MANIPULATION IS DONE BY SUBROUTINE DGBITR, WHICH TREATS
C              CELLS AS MATRIX ELEMENTS.
C              A LIMIT OF THE NUMBER OF CELLS IS THE DIMENSION OF IW1
C              (OR IW2) TIMES THE NUMBER OF BITS PER WORD OF THE MACHINE.
C
C---      INPUT PARAMETERS --
C              IZ -- TWO-DIMENSIONAL ARRAY OF VALUES IN ALPHA-NUMERIC
C                   FORM--A MATRIX OR COLLECTION OF CELLS OF MM COLUMNS
C                   AND MY ROWS.
C              MX -- NUMBER OF COLUMNS OR X-INCREMENTS OF MATRIX THAT
C                   ARE ACTUALLY TO BE PROCESSED.
C              MY -- SECOND DIMENSION OF IZ AND NUMBER OF ROWS OR Y-
C                   INCREMENTS OF MATRIX TO BE PROCESSED.
C              MM -- FIRST DIMENSION OF IZ (MAY EXCEED MX).
C              II --COLUMN NUMBER OF CURRENT CELL--
C                   ITS NEIGHBOR WAS FOUND BY CALLING ROUTINE TO HAVE
C                   A DIFFERENT VALUE.
C              JJ -- ROW NUMBER OF CURRENT CELL.
C              ITYPE -- SPECIFIES THAT NEIGHBOR IS TO THE RIGHT OF
C                   (ITYPE=1) OR ABOVE (ITYPE=2) CURRENT CELL.
C
C---      OUTPUT PARAMETERS --
C              NO VALUES RETURNED THROUGH PARAMETERS.
C              OUTPUT OF ROUTINE IS THROUGH PLOTTING CALLS.
C
C---      REVISIONS --
C
C-----------------------------------------------------------------------
      INTEGER IZ(MM,MY)
      COMMON /DGUAA1/ XOF,YOF,DX,DY,IW1(400),IW2(400)
      I = II
      J = JJ
      IT = ITYPE
      IF(ITYPE.EQ.2) GO TO 10
      CALL DGBITR(IW1,I,J,MX,IBIT)
C        RETURN IF ALREADY PLOTTED SEGMENT
      IF(IBIT.EQ.1) GO TO 999
C        PLOT SEGMENT
      X = XOF+FLOAT(II)*DX
      Y = YOF+FLOAT(JJ-1)*DY
      CALL PLOT(X,Y,3)
      Y = Y+DY
      CALL PLOT(X,Y,2)
      GO TO 100
   10 CALL DGBITR(IW2,I,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 999
      X = XOF+FLOAT(II-1)*DX
      Y = YOF+FLOAT(JJ)*DY
      CALL PLOT(X,Y,3)
      X = X+DX
      CALL PLOT(X,Y,2)
C---     PROCESS NEXT SEGMENT
  100 IF(IBIT.EQ.1) GO TO 999
      GO TO (200,300,400,500),IT
C   VERTICAL SEGMENT BETWEEN (I,J) AND (I+1,J) WITH TOP POINT ACTIVE
  200 IF(J.GE.MY) GO TO 999
      IF(IZ(I,J).EQ.IZ(I,J+1)) GO TO 210
      CALL DGBITR(IW2,I,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 210
      X = X-DX
      CALL PLOT(X,Y,2)
      IT = 4
      GO TO 100
  210 IF(IZ(I,J+1).EQ.IZ(I+1,J+1)) GO TO 220
      CALL DGBITR(IW1,I,J+1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 220
      Y = Y+DY
```

```
      CALL PLOT(X,Y,2)
      IT = 1
      J = J+1
      GO TO 100
  220 IF(IZ(I+1,J+1).EQ.IZ(I+1,J)) GO TO 999
      CALL DGBITR(IW2,I+1,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 100
      X = X+DX
      CALL PLOT(X,Y,2)
      IT = 2
      I = I+1
      GO TO 100
C   HORIZONTAL SEGMENT BETWEEN (I,J) AND (I,J+1) WITH RIGHT POINT ACTIVE
  300 IF(I.GE.MX) GO TO 999
      IF(IZ(I,J+1).EQ.IZ(I+1,J+1)) GO TO 310
      CALL DGBITR(IW1,I,J+1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 310
      Y = Y+DY
      CALL PLOT(X,Y,2)
      IT = 1
      J = J+1
      GO TO 100
  310 IF(IZ(I+1,J+1).EQ.IZ(I+1,J)) GO TO 320
      CALL DGBITR(IW2,I+1,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 320
      X = X+DX
      CALL PLOT(X,Y,2)
      IT = 2
      I = I+1
      GO TO 100
  320 IF(IZ(I,J).EQ.IZ(I+1,J)) GO TO 999
      CALL DGBITR(IW1,I,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 100
      Y = Y-DY
      CALL PLOT(X,Y,2)
      IT = 3
      GO TO 100
C   VERTICAL SEGMENT BETWEEN (I,J) AND (I+1,J) WITH BOTTOM POINT ACTIVE
  400 IF(J.LE.1) GO TO 999
      IF(IZ(I,J).EQ.IZ(I,J-1)) GO TO 410
      CALL DGBITR(IW2,I,J-1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 410
      X = X-DX
      CALL PLOT(X,Y,2)
      IT = 4
      J = J-1
      GO TO 100
  410 IF(IZ(I,J-1).EQ.IZ(I+1,J-1)) GO TO 420
      CALL DGBITR(IW1,I,J-1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 420
      Y = Y-DY
      CALL PLOT(X,Y,2)
      IT = 3
      J = J-1
      GO TO 100
  420 IF(IZ(I+1,J-1).EQ.IZ(I+1,J)) GO TO 999
      CALL DGBITR(IW2,I+1,J-1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 100
      X = X+DX
      CALL PLOT(X,Y,2)
      IT = 2
      I = I+1
      J = J-1
      GO TO 100
C   HORIZONTAL SEGMENT BETWEEN (I,J) AND (I,J+1) WITH LEFT POINT ACTIVE
  500 IF(I.LE.1) GO TO 999
      IF(IZ(I-1,J).EQ.IZ(I,J)) GO TO 510
      CALL DGBITR(IW1,I-1,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 510
      Y = Y-DY
      CALL PLOT(X,Y,2)
      IT = 3
```

```
      I = I-1
      GO TO 100
  510 IF(IZ(I-1,J).EQ.IZ(I-1,J+1)) GO TO 520
      CALL DGBITR(IW2,I-1,J,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 520
      X = X-DX
      CALL PLOT(X,Y,2)
      IT = 4
      I = I-1
      GO TO 100
  520 IF(IZ(I-1,J+1).EQ.IZ(I,J+1)) GO TO 999
      CALL DGBITR(IW1,I-1,J+1,MX,IBIT)
      IF(IBIT.EQ.1) GO TO 100
      Y = Y+DY
      CALL PLOT(X,Y,2)
      IT = 1
      I = I-1
      J = J+1
      GO TO 100
C---     EXIT
  999 CONTINUE
      RETURN
      END
      SUBROUTINE CHKCEL(IZ,MX,MY,MM,II,JJ,XPOS,YPOS,CHRHGT,NUMCHR)
C
C--------CHKCEL VERSION 1.0-------------------------------------------
C
C---      PURPOSE --
C              CHKCEL DETERMINES WHETHER A CELL IS A LOCAL LOWER LEFT
C              CORNER OF A REGION AND, IF IT IS, WHETHER IT IS ADJACENT
C              TO A STANDARD LABEL POSITION IN THE SAME REGION.  IF IT
C              IS A CORNER AND IS NOT ADJACENT TO A STANDARD LABEL, THEN
C              A LABEL IS PLOTTED.
C
C              A LOCAL LOWER LEFT CORNER CELL IS DEFINED BY--
C                  1. A BOUNDARY SEGMENT OR MAP BORDER IMMEDIATELY TO THE
C                   LEFT OF THE CELL
C                  2. A BOUNDARY  SEGMENT OR MAP BORDER IMMEDIATELY BELOW
C                   THE CELL
C                  3. A BOUNDARY SEGMENT OR MAP BORDER AT SOME DISTANCE
C                   TO THE RIGHT OF THE CELL
C                  4. THE BOUNDARY SEGMENTS BETWEEN THE LOWER EDGE
C                   SEGMENT AND THE RIGHT BOUNDARY SEGMENT ALL
C                   HAVE THE SAME Y VALUE
C
C---      OPERATION--
C              THE CALLING ROUTINE TRANSMITS A CELL(II,JJ) WHICH IS KNOWN
C              TO HAVE A BOUNDARY TO ITS LEFT.  CHKCEL EXAMINES THE CELLS
C              BELOW AND TO THE RIGHT TO SEE IF THE BOUNDARY TURNS
C              DOWNWARD  IN VALUE BEFORE IT TURNS UPWARD OR REACHES
C              THE MAP BORDER.
C
C---      INPUT PARAMETERS --
C              IZ -- TWO-DIMENSIONAL ARRAY OF VALUES IN ALPHA-NUMERIC
C                   FORM--A MATRIX OR COLLECTION OF CELLS OF MM COLUMNS
C                   AND MY ROWS.
C              MX -- NUMBER OF COLUMNS OR X-INCREMENTS OF MATRIX THAT
C                   ARE ACTUALLY TO BE PROCESSED.
C              MY -- SECOND DIMENSION OF IZ AND NUMBER OF ROWS OR Y-
C                   INCREMENTS OF MATRIX TO BE PROCESSED.
C              MM -- FIRST DIMENSION OF IZ (MAY EXCEED MX).
C              II --COLUMN NUMBER OF CURRENT CELL--
C                   ITS NEIGHBOR WAS FOUND BY CALLING ROUTINE TO HAVE
C                   A DIFFERENT VALUE.
C              JJ -- ROW NUMBER OF CURRENT CELL.
C              XPOS -- POSITION OF LEFT HAND EDGE OF LABEL RELATIVE TO
C                   MAP ORIGIN.
C              YPOS -- POSITION OF BASE LINE OF LABEL RELATIVE TO MAP
C                   ORIGIN.
C            CHRHGT -- CHARACTER HEIGHT (INCHES)
C              NUMCHR -- NUMBER OF CHARACTERS IN LABEL.
C
```

```
C---      OUTPUT PARAMETERS --
C              NO VALUES RETURNED THROUGH PARAMETERS.
C              OUTPUT OF ROUTINE IS THROUGH PLOTTING CALLS.
C
C---      REVISIONS --
C
C-----------------------------------------------------------------------
      COMMON /DGUAA2/ IMODX,IMODY
      INTEGER IZ(MM,MY)
      IZTST = IZ(II,JJ)
C         IF BOTTOM ROW -- PLOT IT
      IF(JJ.EQ.1) GO TO 200
      JTST = JJ-1
C  THIS LOOP MOVES ACROSS THE COLUMNS TO DETERMINE WHETHER THE
C  CONTINUATION OF THE LOWER CELL BOUNDARY SEGMENT TURNS UP OR DOWN.
      DO 100 I = II,MX
      IF(IZTST.NE.IZ(I,JJ)) GO TO200
      IF(IZTST.EQ.IZ(I,JTST)) GO TO 999
  100 CONTINUE
  200 ZTST = IZTST
C  DETERMINE WHETHER CURRENT CELL IS ONE OF EIGHT CELLS SURROUNDING
C  STANDARD LABEL.  IF NOT, PLOT LABEL.
      IF(IMODX.GT.2) GO TO 300
      IF(IMODY.GT.2) GO TO 300
      IF(IMODX.LT.0) GO TO 300
      IF(IMODY.LT.0) GO TO 300
      IDELX = 1-IMODX
      IDELY = 1-IMODY
      IF(IZ(II+IDELX,JJ+IDELY).NE.IZTST) GO TO 300
C  DETERMINE WHETHER CURRENT CELL IS DIAGONALLY OR DIRECTLY ADJACENT.
      IF(IDELX.EQ.0) GO TO 999
      IF(IDELY.EQ.0) GO TO 999
C  DIAGONAL CASE--DETERMINE IF CURRENT CELL IS IN SAME REGION AS
C  ADJACENT STANDARD LABEL CELL.
      IF(IZ(II+IDELX,JJ).EQ.IZTST) GO TO 999
      IF(IZ(II,JJ+IDELY).EQ.IZTST) GO TO 999
  300 CONTINUE
      CALL SYMBOL(XPOS,YPOS,CHRHGT,IZTST,0.,NUMCHR)
C---      EXIT
  999 CONTINUE
      RETURN
      END
      SUBROUTINE DGBITR(IW,I,J,MX,IBIT)
C
C---------DGBITR VERSION 1.0---------------------------------------------
C
C---      PURPOSE --
C              GIVEN A MATRIX ELEMENT(I,J)
C                   1. CHECK WHETHER IT IS 0 OR 1 AND RETURN RESULT
C                   2. SET ELEMENT TO 1
C
C              EACH ELEMENT IN THE MATRIX IS REPRESENTED BY ONE BIT IN
C              THE ARRAY IW(-).  THE SUBSCRIPTS (I,J) THAT DEFINE A
C              MATRIX ELEMENT ARE CONVERTED INTO SUBSCRIPTS IWORD AND
C              JBIT.  IWORD SPECIFIES AN ELEMENT OF ARRAY IW AND JBIT
C              SPECIFIES A BIT IN THAT WORD.
C
C              BITS ARE MANIPULATED USING MASKS CONSISTING OF A 1 IN THE
C              (61-JBIT)TH PLACE AND 0"S IN ALL OTHER PLACES.  THE MASKS
C              ARE STORED IN ARRAY ABIT(-).
C
C---      INPUT PARAMETERS --
C              IW -- ONE DIMENSIONAL ARRAY.  THE BITS OF THIS ARRAY ARE
C                   ASSIGNED IN ORDER TO ELEMENTS (I,J) OF THE MATRIX.
C              I -- COLUMN OF CURRENT MATRIX ELEMENT.
C              J -- ROW OF CURRENT MATRIX ELEMENT.
C              MX -- NUMBER OF COLUMNS IN THE MATRIX.
C
C---      OUTPUT PARAMETERS --
C              IBIT -- RETURNS 1 IF ELEMENT (I,J) WAS ALREADY 1,
C                   OTHERWISE IT RETURNS 0
C
```

```
C---      NOTE--
C              THIS SUBROUTINE  IS COMPUTER DEPENDENT
C
C---      REVISIONS --
C
C-----------------------------------------------------------------------
      INTEGER ABIT(60)
      INTEGER IW(1)
      DATA ABIT /40000000000000000000B,20000000000000000000B,
     1 10000000000000000000B,4000000000000000000B,2000000000000000000B,
     2 1000000000000000000B,400000000000000000B,200000000000000000B,
     3 100000000000000000B,40000000000000000B,20000000000000000B,
     4 10000000000000000B,4000000000000000B,2000000000000000B,
     5 1000000000000000B,400000000000000B,200000000000000B,
     6 100000000000000B,40000000000000B,20000000000000B,10000000000000B,
     7 4000000000000B,2000000000000B,1000000000000B,400000000000B,
     8 200000000000B,100000000000B,40000000000B,20000000000B,
     9 10000000000B,4000000000B,2000000000B,1000000000B,400000000B,
     A 200000000B,100000000B,40000000B,20000000B,10000000B,4000000B,
     B 2000000B,1000000B,400000B,200000B,100000B,40000B,20000B,10000B,
     C 4000B,2000B,1000B,400B,200B,100B,40B,20B,10B,4B,2B,1B/
      IPOS = (J-1)*MX+I
      IWORD = (IPOS-1)/60+1
      JBIT = IPOS-(IWORD-1)*60
C  THE MASK HAS EXACTLY ONE 1, SO THAT THE "AND" EFFECTIVELY CHECKS ONLY
C  THE (JBIT)TH BIT OF IW(IWORD).
C  IF THAT BIT IS 1, THEN KBIT .NE. 0, AND IBIT = 1
      KBIT = AND(ABIT(JBIT),IW(IWORD))
      IBIT = 0
      IF(KBIT.NE.0) IBIT = 1
C  THE MASK HAS EXACTLY ONE 1, SO  "OR"ING CAN AFFECT ONLY THE (JBIT)TH
C  BIT OF IW(IWORD), LEAVING IT 1 IN BOTH CASES.
      IW(IWORD) = OR(IW(IWORD),ABIT(JBIT))
C---     EXIT
  999 CONTINUE
      RETURN
      END
```