ISSN-0709-999 1

Environment Canada / Environnement Canada
Forestry Service / Service des Forêts

A Ground Survey Method For Estimating Loss Caused By *Phellinus weirii* Root Rot

II. Survey Procedures and Data Analysis

W. J. Bloomberg, P. M. Cumberbirch and G. W. Wallis
Canadian Forestry Service / Pacific Forest Research Centre
Victoria, B.C., BC-R-4, October, 1980

SUMMARY

Procedures are described for designing a ground-survey for *Phellinus weirii* root rot in forest stands. Office procedures include selection of baseline location and number, location and width of transect lines. Field procedures include measurement of infection centers by intersection length, rectangular, radial or linear methods. Five survey method options include measurement of infection centers, estimation of infection center area, assignment of infection centers to size classes, measurement of intersection length, and assignment of intersection length. Methods for recording, analyzing and summarizing field data are described for intersection length and probability of occurrence methods. A computer program for analyzing and mapping the data is described. Three mapping options provide a range of detail in maps of infection centers.

RÉSUMÉ

Description des étapes à suivre pour dresser un inventaire au sol du Pourridié causé par *Phellinus weirii* dans les peuplements forestiers. Au bureau, ces étapes comportent la sélection de la place et du numéro de la base métrique, de l'emplacement et de la largeur des lignes de transect. Sur le terrain, elles comprennent le mesurage des centres d'infection par les méthodes de longueur d'intersection, rectangulaire, radiale ou linéaire. Un choix de cinq méthodes d'inventaire s'offre: mesurage des centres d'infection, estimation de la région infectée, désignation des centres d'infection par catégories d'étendue, mesurage de la longueur des intersections, détermination de la longueur des intersections. Les méthodes d'enregistrement, d'analyse et de résumé des données sur le terrain sont décrites quant à la longueur des intersections et à la probabilité des méthodes d'occurrence. Un programme informatisé pour analyser et cartographier les données est aussi décrit. Trois possibilités de cartographie fournissent une gamme de détails sur les cartes des centres d'infection.

W. J. Bloomberg,[1] P. M. Cumberbirch[2] and G. W. Wallis[1]

[1] Forest Pathologist

[2] Statistical Consultant

TABLE OF CONTENTS

INTRODUCTION

Of the pests occurring in second-growth stands in British Columbia and the northwest United States, root rots are one of the most destructive (Pacific Northwest Pest Action Council 1975). Annual losses resulting from growth reduction and mortality are estimated to exceed 5 million cubic metres (Childs and Shea 1961; Wallis 1967). *Phellinus weirii* (Murr.) Gilb. is responsible for a major portion of this loss. The disease also imposes severe limitations on the number of intensive management options available for infected stands. The ability of the fungus to spread in stands throughout a rotation and the difficulty of detecting early stages of infection could result in the loss of large numbers of crop trees following precommercial and commercial thinnings in stands where the disease is common. *Phellinus weirii* survives for many decades in stumps and roots, infecting living roots when they contact the fungus, thereby perpetuating the disease in subsequent rotations. Complex and costly land clearing operations and species manipulation are required to reduce losses to acceptable levels.

Summary of Method

This report describes ground survey and data processing procedures for estimating the incidence, distribution and area of *P. weirii* infection centers in stands, based on above-ground symptoms, (as described by Wallis 1976). A knowlege of disease symptoms is prerequisite to applying the procedures. Derivation and testing of the method is described in Bloomberg *et al.* (1980). The method is summarized as follows. The survey design comprises sets of equidistant, parallel transect lines traversing a stand from one boundary to the opposite one. Each set of transect lines, called a grid, commences from a randomly chosen point on a baseline, running at right angles to the baseline (Fig. 1). The distance between lines is the grid interval.

Estimates of total root rot area are derived from the total length of the transect lines in a grid falling within infection centers (intersection length method, Fig. 2) or from the areas of individual infection centers falling within a specified strip width on either side of transect lines (probability of occurrence

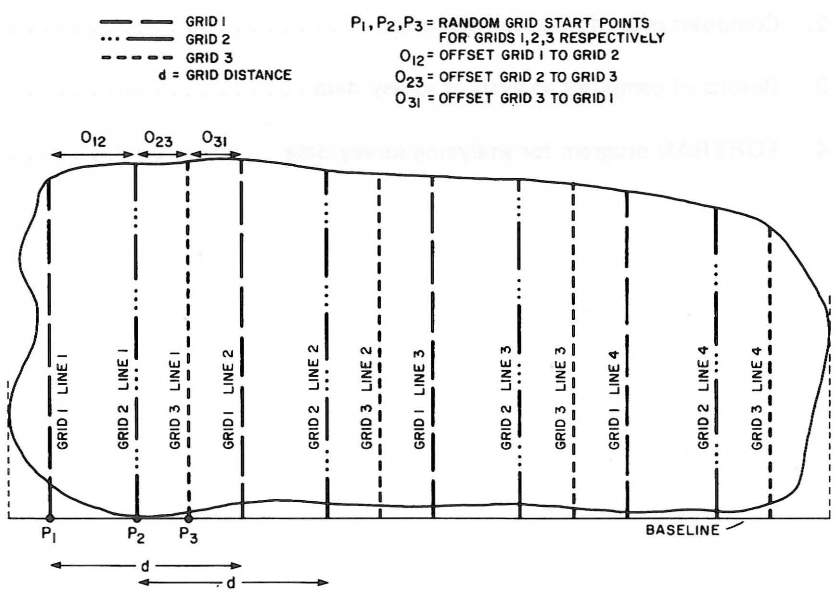


Fig. 1. Root rot survey design consisting of three grids each containing four transect lines.

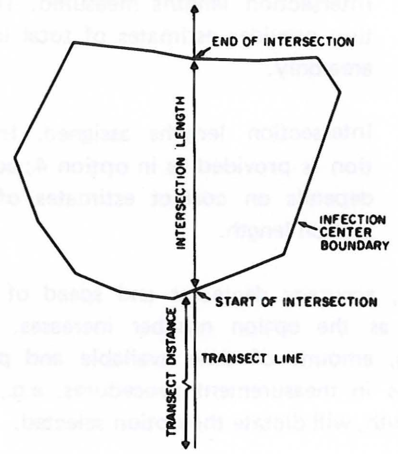


Fig. 2. Measurement of intersection length of an infection center.

---

method).

Areas of individual infection centers are measured by one or more of four methods:

1) Rectangular method (Fig. 3) using lengths of two axes, *i.e.* mid-length, parallel to the transect line, and projection length, parallel to the baseline

2) Radial method (Fig. 4), using distances from mid-point of an infection center to its boundary at equidistant angles

3) Linear method (Fig. 5), using a series of parallel, equidistant lines at right angles to a centrally located line within the infection center

4) Regression method, by estimating the area of rectangularly measured infection centers from their regression relationship with radially or linearly measured areas

Numbers and total area of infection centers within specified size classes are estimated by probability of occurrence based on the ratio of projection length of an infection center to the grid interval (Bloomberg *et al.* 1980). Independent estimates of total root rot area and numbers of infection centers are derived from two or more grids and provide a mean estimate for a stand with variance and confidence limits.

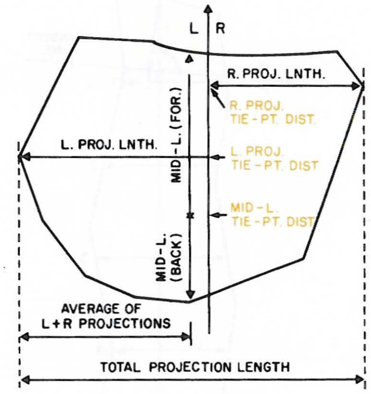


Fig. 3. Rectangular measurement of infection center requires measurement of left and right projection length and mid-length. Measurements for mapping option 2 (shown in color) are right and left projection tie-point distances and mid-length tie-point distance. Mid-length can be measured as a single segment (forward) or as two segments (forward and back) as shown in diagram. Side of mid-length is recorded as left (L) or right (R) of transect line.

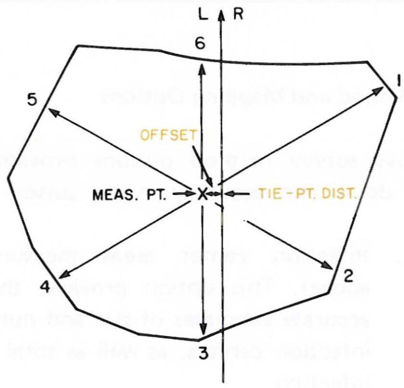


Fig. 4. Radial measurement of infection center requires measurement of distances from measurement point to center boundary. Angle between radials is constant. Measurements for mapping option 3 (shown in color) are tie-point distance and offset. Side of measurement point is recorded as left (L) or right (R) of transect line.

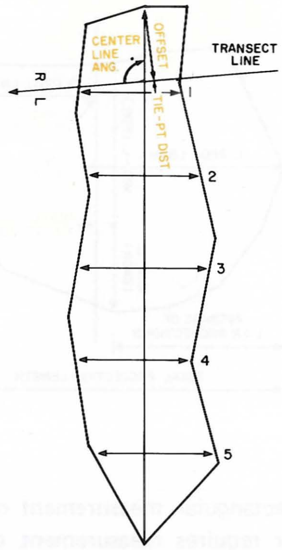


Fig. 5. Linear measurement of infection center requires location of center line with equidistant measurements at right angles from line to infection center boundary. Measurements are recorded in two segments, forward (in direction of transect) and back. Measurements required for mapping option 3 (shown in color) are tie-point distance, offset and center line angles. Side of measurement point (start of center line) is recorded as left (L) or right (R) of transect.

Survey Method and Mapping Options

Five survey method options provide for a range of detail and accuracy in the survey results.

Option 1. Infection center areas measured (as above). This option provides the most accurate estimates of size and number of infection centers, as well as total area of infection.

Option 2. Infection center areas estimated. Information is provided as in option 1; accuracy depends on the ability to judge center sizes.

Option 3. Infection centers assigned to size classes. Information is provided as in option 1; accuracy depends on correct assignment of infection centers to their size classes.

Option 4. Intersection lengths measured. This option provides estimates of total infected area only.

Option 5. Intersection lengths assigned. Information is provided as in option 4; accuracy depends on correct estimates of intersection length.

Generally, accuracy decreases and speed of survey increases as the option number increases. Survey objectives, amount of time available and physical difficulties in measurement procedures, e.g., heavy undergrowth, will dictate the option selected.

Mapping options can be combined with the survey method options, as follows.

Option 0. No map required.

Option 1. Map of intersection lengths only.

Option 2. Map of intersection lengths plus projection and mid-lengths.

Option 3. Map of radial or linear measurements.

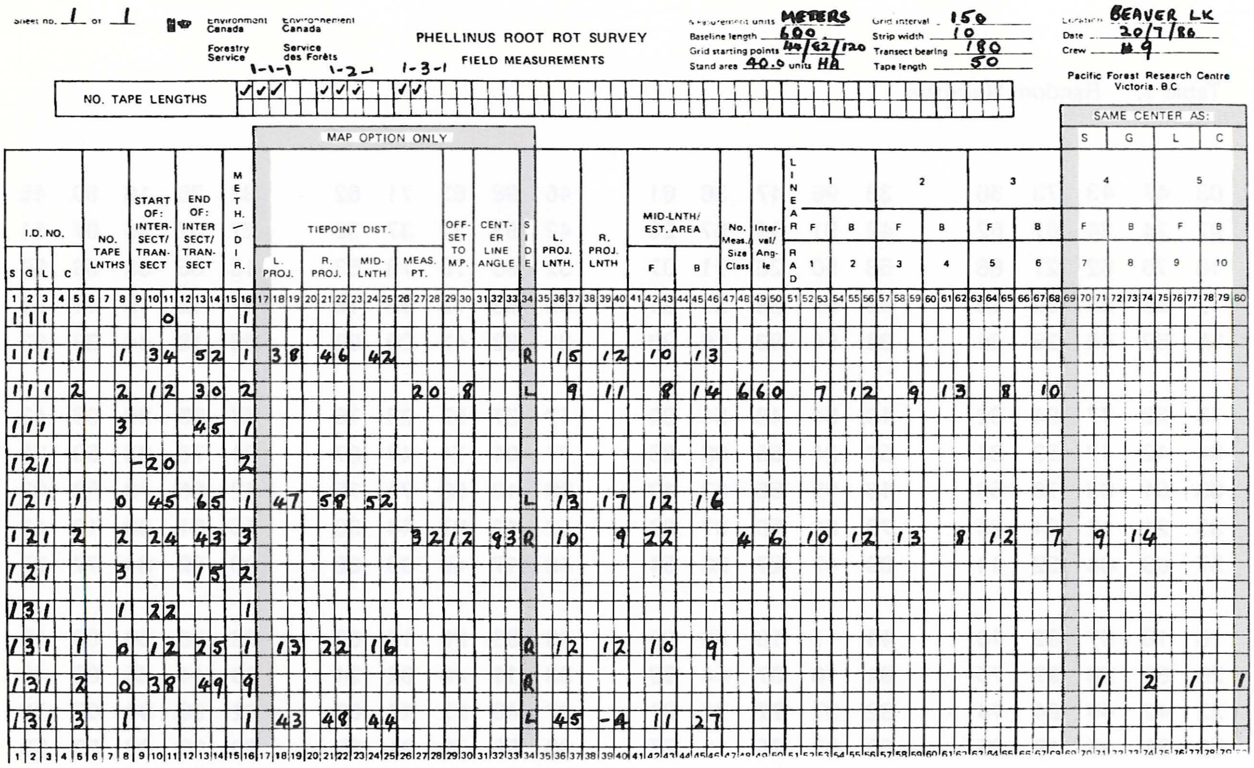
SURVEY PROCEDURES

The following sections describe office and field procedures used to apply the survey method. Strict adherence to these procedures is essential if the data are to be analyzed by the methods described in Data Analysis. Procedures for survey method option 1 without map will be described first, followed by modifications required for other options.

Office Preparations

1. Identify stand to be surveyed and record location and area in description section of tally sheet (Fig. 6). Record stand area units (acres, hectares, etc.). The stand should be accurately mapped and be relatively homogeneous with respect to root rot incidence. Forest cover types based on species composition, age and site will normally satisfy these requirements.

2. Record date and identification of crew on tally sheet.



Sheet no. 1 of 1 — Environment Canada / Environnement Canada — Forestry Service / Service des Forêts — PHELLINUS ROOT ROT SURVEY — FIELD MEASUREMENTS

Measurement units METERS; Baseline length 600; Grid starting points 44/62/120; Stand area 40.0 units HA; Grid interval 150; Strip width 10; Transect bearing 180; Tape length 50; Location BEAVER LK; Date 20/7/86; Crew #9

Pacific Forest Research Centre, Victoria, B.C.

NO. TAPE LENGTHS: 1-1-1 ✓✓✓; 1-2-1 ✓✓✓; 1-3-1 ✓✓

I.D. NO. S G L	C	NO. TAPE LNTHS	START OF: INTERSECT/TRANSECT	END OF: INTERSECT/TRANSECT	METH./DIR.	TIEPOINT DIST. L. PROJ.	TIEPOINT DIST. R. PROJ.	TIEPOINT DIST. MID-LNTH	TIEPOINT DIST. MEAS. PT.	OFFSET TO M.P.	CENTER LINE ANGLE	SIDE	L. PROJ. LNTH.	R. PROJ. LNTH	MID-LNTH/EST. AREA F	MID-LNTH/EST. AREA B	No. Meas./Size Class	Interval/Angle	1 F / 1	1 B / 2	2 F / 3	2 B / 4	3 F / 5	3 B / 6	4 F / 7 (SAME CENTER AS: S)	4 B / 8 (G)	5 F / 9 (L)	5 B / 10 (C)
111			0		1																							
111	1	1	34	52	1	38	46	42				R	15	12	10	13												
111	2	2	12	30	2				20	8		L	9	11	8	14	6	60	7	12	9	13	8	10				
111		3		45	1																							
121			-20		2																							
121	1	0	45	65	1	47	58	52				L	13	17	12	16												
121	2	2	24	43	3				32	12	93	R	10	9	22		4	6	10	12	13	8	12	7	9	14		
121		3		15	2																							
131		1	22		1																							
131	1	0	12	25	1	13	22	16				R	12	12	10	9												
131	2	0	38	49	9							R													1	2	1	1
131	3	1			1	43	48	44				L	45	-4	11	27												

Fig. 6. Tally sheet for recording descriptive information (upper section) and infection center measurements. Measurements within shaded section headed Map Option Only can be omitted if map is not required.

3. Locate a baseline (Fig. 7) according to the following guides and record length on tally sheet.
   a) Baseline must run full extent of the stand.
   b) As far as practical, baseline orientation should minimize differences in transect line lengths.
   c) Baseline may be inside or outside stand.
   d) Baselines should not exceed 2 000 m length. Larger stands should be subdivided into blocks which are surveyed separately.

4. Define the number of grids to be used and the number of transect lines per grid. These numbers will be determined by the time available for the survey, the size of the stand, distribution of infection centers, and the statistical accuracy required. The following guide is based on results of field tests (Bloomberg *et al.* 1980). In small stands, *i.e.,* baseline length less than 200 m, three grids with two transect lines each will probably be adequate. Medium-size stands, baseline length 200 - 600 m, probably require three grids of four lines. Larger stands should have four or five grids with three or four lines per grid. The larger and

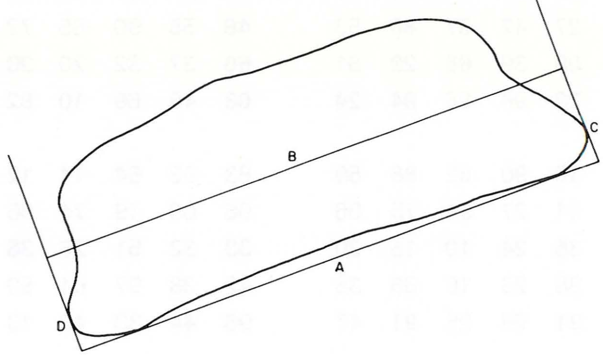


Fig. 7. Location of baseline may be outside (A) or inside stand (B), but must extend full length of stand, *i.e.,* between lines at right angles to baseline and passing through opposite extremes of the stand.

Table 1.   Random Numbers

03	47	43	73	86	36	96	47	36	61	46	98	63	71	62	33	26	16	80	45
97	74	24	67	62	42	81	14	57	20	42	53	32	37	32	27	07	36	07	51
16	76	62	27	66	56	50	26	71	07	32	90	79	78	53	13	55	38	58	59
12	56	85	99	26	96	96	68	27	31	05	03	72	93	15	57	12	10	14	21
55	59	56	35	64	38	54	82	46	22	31	62	43	09	90	06	18	44	32	53
16	22	77	94	39	49	54	43	54	82	17	37	93	23	78	87	35	20	96	43
84	42	17	53	31	57	24	55	06	88	77	04	74	47	67	21	76	33	50	25
63	01	63	78	59	16	95	55	67	19	98	10	50	71	75	12	86	73	58	07
33	21	12	34	29	78	64	56	07	82	52	42	07	44	38	15	51	00	13	42
57	60	86	32	44	09	47	27	96	54	49	17	46	09	62	90	52	84	77	27
18	18	07	92	46	44	17	16	58	09	79	83	86	19	62	06	76	50	03	10
26	62	38	97	75	84	16	07	44	99	83	11	46	32	24	20	14	85	88	45
23	42	40	64	74	82	97	77	77	81	07	45	32	14	08	32	98	94	07	72
52	36	28	19	95	50	92	26	11	97	00	56	76	31	38	80	22	02	53	53
37	85	94	35	12	83	39	50	08	30	42	34	07	96	88	54	42	06	87	98
70	29	17	12	13	40	33	20	38	26	13	89	51	03	74	17	76	37	13	04
56	62	18	37	35	96	83	50	87	75	97	12	25	93	47	70	33	24	03	54
99	49	57	22	77	88	42	95	45	72	16	64	36	16	00	04	43	18	66	79
16	08	15	04	72	33	27	14	34	09	45	59	34	68	49	12	72	07	34	45
31	16	93	32	43	50	27	89	87	19	20	15	37	00	49	52	85	66	60	44
68	34	30	13	70	55	74	30	77	40	44	22	78	84	26	04	33	46	09	52
74	57	25	65	76	59	29	97	68	60	71	91	38	67	54	13	58	18	24	76
27	42	37	86	53	48	55	90	65	72	96	57	69	36	10	96	46	92	42	45
00	39	68	29	61	66	37	32	20	30	77	84	57	03	29	10	45	65	04	26
29	94	98	94	24	68	49	69	10	82	53	75	91	93	30	34	25	20	57	27
16	90	82	66	59	83	62	64	11	12	67	19	00	71	74	60	47	21	29	68
11	27	94	75	06	06	09	19	74	66	02	94	37	34	02	76	70	90	30	86
35	24	10	16	20	33	32	51	26	38	79	78	45	04	91	16	92	53	56	16
38	23	16	86	38	42	38	97	01	50	87	75	66	81	41	40	01	74	91	62
31	96	25	91	47	96	44	33	49	13	34	86	82	53	91	00	52	43	48	85
66	67	40	67	14	64	05	71	95	86	11	05	65	09	68	76	83	20	37	90
14	90	84	45	11	75	73	88	05	90	52	27	41	14	86	22	98	12	22	08
68	05	51	18	00	33	96	02	75	19	07	60	62	93	55	59	33	82	43	90
20	46	78	73	90	97	51	40	14	02	04	02	33	31	08	39	54	16	49	36
64	19	58	97	79	15	06	15	93	20	01	90	10	75	06	40	78	78	89	62

more uniform the infection centers, the fewer lines per grid will be required; however, this information is generally not known in advance.

5. Select grid starting points and intervals using the following procedure:

   a) Divide baseline length by number of transect lines per grid to obtain interval between lines (grid interval); record on tally sheet.
   b) From a table of random numbers (Table 1), pick one three-digit number for each grid. Multiply the grid interval by the smallest random number and divide by 1000 to locate the first grid starting point (G.S.P.). Repeat procedure with next largest random number to obtain G.S.P. of second and subsequent grids.

Example: Baseline length = 500 m, number of grids = 3, number of transect lines per grid = 4.

Grid interval = 500 ÷ 4 = 125 m.

Random numbers picked are 567, 123, 811

First G.S.P. = (123 x 125) ÷ 1000 = 15.4, rounded to 15 m from start of baseline

Second G.S.P. = (567 x 125) ÷ 1000 = 71 m

Third G.S.P. = (811 x 125) ÷ 1000 = 101 m

   c) If G.S. P. is less than 10% of the grid interval from another G.S.P., pick another random number and recalculate. Record grid starting points on tally sheet.
   d) To calculate locations of subsequent transect lines in each grid, add the grid interval (125 m in example) to each G.S.P., *i.e.,* lines 1,2,3 and 4 in grid 1 are located at 15, 140, 265 and 390 m from the start of the baseline, respectively; lines 1 to 4 in grid 2 are at 71, 196, 321 and 446 m, and lines 1 to 4 in grid 3 are at 101, 226, 351, and 476 m, respectively.
   e) Calculate offset distances between adjacent transect lines by substracting one G.S.P. from the next one to it, *i.e.,* offset distance from any line in grid 1 to an adjacent line in grid 2 will be 71 – 15 = 56 m, from grid 2 to grid 3, 101 – 71 = 30 m, and from grid 3 to grid 1, 125 + 15 – 101 = 39 m. Offsets can be double-checked by their total, which should equal the grid distance, *i.e.,* 56 + 30 + 39 = 125 m.

6. Determine transect line locations (Fig. 1).
   a) Lines must be at right angles to the baseline. Orientation relative to topography is immaterial.
   b) Lines must run from one edge of stand to the opposite edge.
   c) Offsets must be at right angles to transect lines.
   d) Plot baseline, transect lines and offsets on map.
   e) Record transect bearing from baseline on tally sheet.

7. Determine transect line width (strip width) and record on tally sheet. The following guides should be used:
   a) Width is zero for intersection length method.
   b) Width can be 0 for probability of occurrence method if there is prior knowledge that at least 20 centers will be encountered (necessary for regression calculation).
   c) If no prior knowledge is available on center frequency, recommended strip width for medium to fully stocked stands is 5 m on either side of transect line.
   d) Strip width can be increased in sparsely stocked stands.
   e) Once selected, the strip width must remain constant throughout the survey.

8. Record on tally sheet measurement units (feet, metres, etc.) and length of tape used for measuring transect length.

Field Procedures

(Survey method option 1, no map required)

1. Record stand no., grid no., and line no. under S, G. and L, respectively (columns 1 - 3 of tally sheet, Fig. 6).

2. Commence transect line, observing trees along line and on either side if a strip width has been added. The following description assumes a strip width of 5 m on either side of the line. Correct for slope. Record number of tape lengths travelled in the spaces provided at the top of the tally sheet.

3. When an infection center is observed but its near boundary is more than 5 m from the line, proceed with transect line. If an infection center is less than 5 m from line but is not intersected by it, proceed to step 6. If intersected by line, proceed to step 4. An infection center boundary is defined in paragraph 8.

4. If transect line intersects boundary of an infection center (Fig. 2), take the following action:

   a) Assign an infection center I.D. number on a new line of the tally sheet (columns 1 - 5). The I.D. number consists of stand, grid, line and center numbers recorded under S, G, L, and C. respectively. Centers are numbered sequentially from 1 at the start of each transect line.

   b) Record distance from start of transect line (hereafter called transect distance) in whole number of tape lengths in columns 6 - 8 of tally sheet. Record start (columns 9 - 11) and end (columns 12 - 14) of intersection of line with boundary of infection center as part tape lengths. All entries on the tally sheet must be right-justified, *i.e.,* the rightmost digit of a measurement must be in the rightmost column of the measurement field. For example, start and end of intersection of 412 and 437 m respectively, is recorded as 8 in column 8 (assuming a 50 m tape), 12 in columns 10 - 11, and 37 in columns 13 - 14.

   c) Measure distance from transect line to farthest left extent of infection center (left projection length (Fig. 3) and record in columns 35 - 37. Measure distance from transect line to farther right extent (right projection) length and record in columns 38 - 40.

   d) Calculate mid-length measurement point, *i.e.,* average of left and right projections, and offset from the transect line to this point (Fig. 3). Measure mid-length from this point in one or two segments, whichever is more convenient. Record forward segment (in direction of transect) in columns 41 - 43 and backward segment in columns 44 - 46, or record as a single segment in columns 41 - 43.

5. If infection center is to be measured only by the rectangular method (see Introduction), record measurement method as 1 in column 16. If infection center is to be measured in more detail, choose one of the following methods:

   i) Radial method (Fig. 4), if infection center is more or less circular. Record measurement method as 2 in column 16. Locate approximate middle of infection center (measurement point). Depending on infection center size and irregularity of margin, take up to 10 equidistantly spaced measurements from measurement point to infection center boundary (radial distance). Record in columns 51 - 80 under heading RAD 1 - 10. Record number of radial distances measured (columns 47 - 48), and angle between them (columns 49 - 50), e.g. 4 measurements with 90^{o} angle between them, 6 with 60^{o} angle, 8 with 45^{o} angle.

   ii) Linear method (Fig. 5), for all other situations, e.g. if infection center is very elongated or irregular in shape. Record measurement method as 3 in column 16. Locate a line running the full length of the infection center (center line). Choose up to five measurement points at equidistant intervals along the line. Record number of measurement points and interval in columns 47 - 48 and 49 - 50 respectively. Measure distance from each point to infection center boundary at right angles to the line (crossdistance). Record as two segments, forward (in direction of transect) and backward (F and B in columns 51 - 80 under heading LINEAR) or as a single segment under F. To determine interval along center line, divide its length by the number of measurement points plus 1.

Example: Center line length = 40 m

No. measurement points required = 4

Measurement interval = 40 ÷ 5 = 8 m

Measurement points at 8, 16, 24 and 32 m along center line.

Proceed to Step 7.

6. If boundary of infection center lies between transect line and the strip width boundary, *i.e.,* less than 5 m from the line, take steps 4 and 5 except 4b, *i.e.,* measure infection center area but not

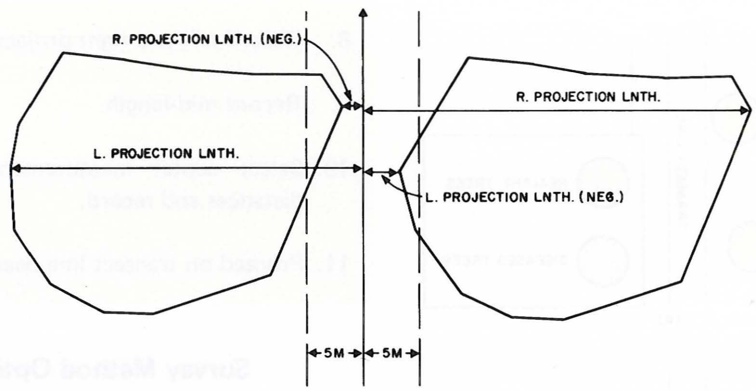


Fig. 8. Measurement of left and right projection length where infection center is intersected by strip boundary but not by transect line (see text for explanation).

intersection length. Measure left and right projection lengths as follows (Fig. 8): If infection center is to left of transect line, measure distance from line to right boundary and record under right projection length (columns 38 - 40) as a negative value. Record left projection length (columns 35 - 37) as distance from line to extreme left point of the infection center. If infection center is to right of line, measure distance from line to left boundary and record left projection length as a negative value. Record right projection length as the distance from line to extreme right point.

7. If infection center is intersected by the line but was already measured on a **different** grid, record measurement method as 9 in column 16. If infection center is intersected by the line but was already measured on the **same** grid, record measurement method as 8. In both cases, assign a new infection center I.D. number and record in columns 1 - 5. In addition, record in columns 69 - 80 under the headings "Same center as S,G, L, C the I.D. number assigned when the infection center was first encountered. Measure intersection length only. If the infection center is not intersected by the line, do not record it. Typically, a large infection center may extend over to the transect line of the next grid (measurement code 9), but occasionally may extend over to the next line of the same grid (measurement code 8).

8. The boundary of an infection center is defined by straight lines joining the inner faces of healthy-appearing, margin trees (Fig. 9).

   a) A healthy-appearing tree is defined as:
      i) lacking *Phellinus* root rot crown symptoms, as described by Wallis (1976). No attempt is made to locate root mycelium or decay in this procedure;
      ii) having a root-rot tree as nearest neighbor on the side toward the infection center;
      iii) having a healthy-appearing tree as nearest neighbor on side away from the infection center.
   b) Islands of healthy-appearing trees inside an infection center are not considered in marking the boundary.

Summary of field procedures.

1. Establish baseline.
2. Locate first grid starting point.
3. Record stand, grid and line no.
4. Proceed on transect line bearing.
5. Observe infection centers.
6. Check whether intersected by line or within strip width. Record infection center no.
7. Record start and end of line intersection with infection center boundary.

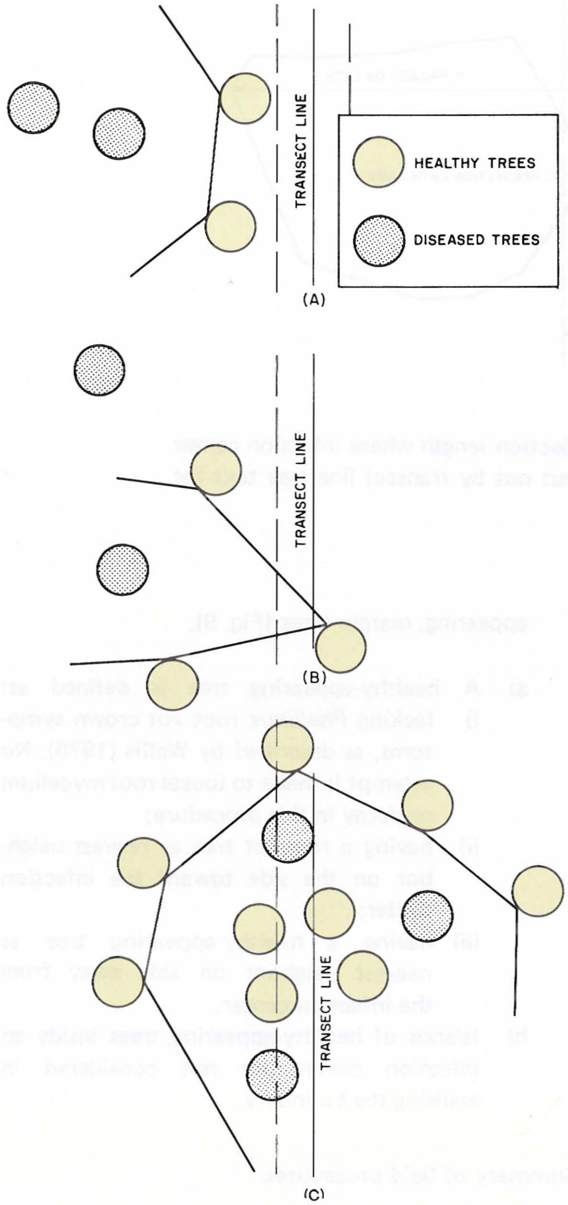


Fig. 9. Definition of infection center boundary relative to intersection by transect line. A, infection center located outside strip width is not recorded. B, transect line intersects extreme end of infection center. Boundary is defined by straight lines joining inner faces of healthy appearing trees having an infected tree as nearest neighbor on side facing infection center and a healthy appearing tree on the opposite side. C, infection center containing islands of healthy trees which are excluded from consideration in defining infection center boundary.

8. Record left and right projection.
9. Record mid-length.
10. Select center measurement method. Measure distances and record.
11. Proceed on transect line bearing.

Survey Method Options 2 - 5

Field procedures for survey method options, other than option 1 (Infection centers measured, described above) are described below.

Option 2 **Infection center areas estimated.** Follow field procedures for option 1, omitting steps 4(d) and 5. Record estimated area of each infection center in columns 41 - 46. Areas must be in survey measurement units (see Office Preparations).

Option 3. **Infection center size class assigned.** Size limits up to five classes must be set in advance (see Input file preparation). Class limits must be in stand measurement units. Classes must be numbered from 1 in ascending order of size. Follow field procedures for option 1, omitting steps 4(d) and 5. Record infection center size class in column 48.

Option 4. **Intersection length only measured.** Follow field procedures for option 1, omitting steps 4(c) - 6.

Option 5. **Intersection lengths only assigned.** Follow field procedures for option 1, omitting steps 4(b) - 6. Record assigned intersection length in columns 9 - 11. Note that it is not necessary to measure transect distance.

MAPPING OPTIONS

Four mapping options are available (see Introduction). Maps are produced by computer, showing the locations of measurement points for each center (Appendix 2).

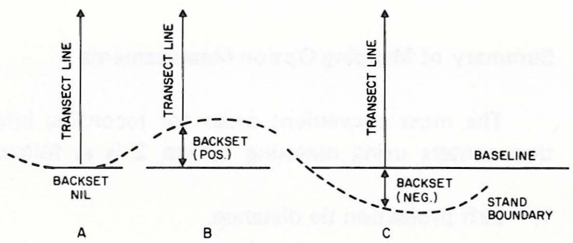


Fig. 10. Measurements required for mapping options at first transect (Grid 1, line 1). A, transect starts on baseline, backset is recorded as O. B, transect starts above baseline (direction 1), backset is recorded as positive value. C, transect starts below baseline (direction 2), backset is recorded as negative value.

Map Option 0 (no map required).

No measurements are required in addition to those described under Field Procedures. For options 1-4 the following additional measurements are necessary.

Map Option 1 (intersection length only).

1. On the first line of tally sheet, record stand, grid and line number and starting point of transect relative to baseline. If the transect starts at the baseline (Fig. 10A), record 0 in column 11. If transect starts above baseline (Fig. 10B), record the backset distance as whole tape lengths (columns 6 - 8) and partial length (columns 9 - 11). If the transect starts below the baseline (Fig. 10C), record the backset distance as whole and partial tape lengths preceded by minus signs.

2. Record transect direction as 1 in column 16.

3. At the end of each transect, record total transect length by whole tape lengths (columns 6 - 8) and partial length (columns 12 - 14). Record transect direction in column 16 as 1 if proceeding away from baseline (direction 1), or 2 if proceeding toward baseline (direction 2).

4. At start of all subsequent transect lines, record stand, grid and line number and direction of transect. Record in whole and partial tape lengths (columns 6 - 8 and 9 - 11 respectively) starting point of transect relative to end of the previous transect. Record 0 if there is no backset (Fig. 11A). Record backset distance as positive if backset is in direction 1 (Fig. 11B) Record backset distance as negative if backset is in direction 2 (Fig. 11C).

Map Option 2 (projection and mid-length only).

1. Record transect line measurements as in option 1.

2. Record in columns 17 - 19 transect distance to point from which left projection length is measured (tie-point distance, Fig. 3). All tie-point distances in this and other options are recorded in whole and part tape lengths, as previously described under Field Procedures paragraph 4(b). Record right projection tie-point distance in columns 20 - 22.

3. Record in columns 23 - 25 tie-point distance to

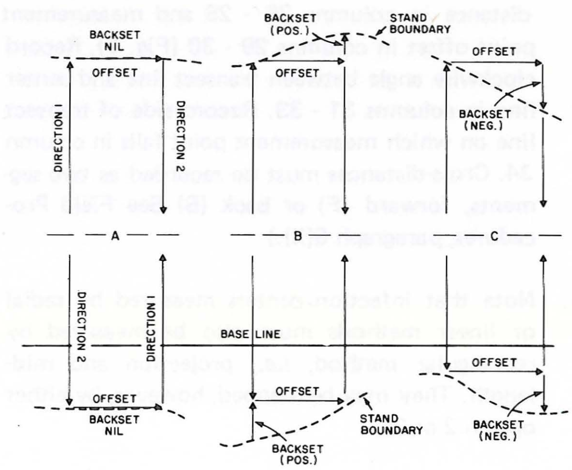


Fig. 11. Measurements required for mapping options at end of one transect line and start of next. A, start of one transect line is at right-angles to end of previous transect. Backset is recorded as O. B, start of one transect is in direction 1 relative to end of previous one. Backset is recorded as a positive value. C, start of one transect is in direction 2 relative to end of previous one. Backset is recorded as a negative value. Top, stand boundary further from baseline. Bottom, stand boundary nearer baseline.

point from which mid-length is measured (measurement point tie-distance Fig. 3).

4. Record L or R in column 34 for side of transect on which mid-length measurement point falls when facing the transect direction. A point falling exactly on the transect line is recorded as either L or R. Mid-length must be recorded as two segments; forward (F) and back (B) (see Field Procedures, paragraph 4(c)).

Map Option 3 (radial or linear measurements).

1. Record transect line measurements as in option 1.

2. If center is measured by radial method, record measurement point tie distance in columns 26 - 28 and offset distance from transect line to measurement point (measurement point offset) in columns 29 - 30 (Fig. 4). Record in column 34 side of transect on which measurement point falls.

3. If center is measured by linear method, record measurement point (beginning of center line) tie distance in columns 26 - 28 and measurement point offset in columns 29 - 30 (Fig. 5). Record clockwise angle between transect line and center line in columns 31 - 33. Record side of transect line on which measurement point falls in column 34. Cross-distances must be recorded as two segments, forward (F) or back (B) See Field Procedures, paragraph 5(ii).)

4. Note that infection centers measured by radial or linear methods must also be measured by rectangular method, *i.e.,* projection and mid-length. They may be mapped, however, by either option 2 or 3.

Summary of Mapping Option Measurements

The most convenient order for recording infection centers using **mapping option 2** is as follows:

1. Left projection tie distance.
2. Left projection length.
3. Right projection tie distance.
4. Right projection length.
5. Mid-length tie distance.
6. Mid-length forward and back.
7. Side of mid-length measurement point.

The most convenient order for recording infection center using **mapping option 3** is as follows:

1. Measurement point tie distance.
2. Measurement point offset.
3. Side of measurement point.
4. Radial measurements (for radial method).
5. Center line angle and interval distances (for linear method).

DATA ANALYSIS

The following section describes the computations required for estimating total area and numbers of infection centers in a stand. Methods for hand computation of results are described, as well as a method for computer analysis of data.

Hand Computation

Estimate of total root rot area by intersection length method

The theoretical basis and application of the intersection length sampling method is described in Bloomberg *et al.* Appendix 1 (1980). The estimate of total area of root rot $\hat{A}_{TOT}$ is the mean of m independent area estimates $\hat{A}_1, \hat{A}_2, ..., \hat{A}_m$, where m is the number of grids. The variance of $\hat{A}_{TOT}$ is estimated as S^2/m, where S^2 is the sample variance.

Let h_{ij} denote the total intersection length of infection centers on the i^{th} transect line in grid j. Let $h_{i.}$ denote the subtotal of intersection lengths by transect line, $h_{.j}$ the subtotal by grid, and $h_{..}$ the total for all lines and grids.

$$h_{i.} = \sum_{j=1}^{m} h_{ij} = (h_{i1} + h_{i2} + \ldots + h_{im})$$

$$h_{.j} = \sum_{i=1}^{n} h_{ij} = (h_{1j} + h_{2j} + \ldots + h_{nj}) \qquad (1)$$

$$h_{..} = \sum_{i=1}^{n} h_{i.} = \sum_{j=1}^{m} h_{.j} = \sum_{i=1}^{n} \sum_{j=1}^{n} h_{ij},$$

where m and n are, respectively, the number of grid systems and the number of lines within each grid system.

For each grid, an unbiased estimate of the total area of infection within the stand $\hat{A}_1, ...,\hat{A}_m$ is obtained

$$\hat{A}_1 = d \times h_{.1}$$

$$\hat{A}_2 = d \times h_{.2} \qquad (2)$$

$$\vdots$$

$$\hat{A}_m = d \times h_{.m},$$

where d is the distance between consecutive lines within grids (grid interval). A more precise estimate of the total area is obtained by averaging the $\hat{A}_i$'s,

$$\hat{A}_{TOT} = (\hat{A}_1 + \hat{A}_2 + \ldots + \hat{A}_m)/m \qquad (3)$$

The variance estimate of $\hat{A}_{TOT}$ is

$$\hat{V}_{TOT} = \sum_{i=1}^{m} (\hat{A}_i - \hat{A}_{TOT})^2 / m(m-1). \qquad (4)$$

From equation 2 these values become

$$\hat{A}_{TOT} = d \times h_{..}/m \qquad (3A)$$

and

$$\hat{V}_{TOT} = d^2 \left[\sum_{j=1}^{m} (h_{.j})^2 - (h_{..})^2/m \right] /m(m-1), \qquad (4A)$$

where $h_{.j}$ and $h_{..}$ are defined as in (1).

Approximate confidence limits for the total area of root rot can be obtained from the Student - t distribution with m − 1 degrees of freedom (Cochran 1963) *i.e.,*

$$\hat{A}_{TOT} \pm t_0 \times \hat{V}^{1/2}_{TOT}$$

representing the 100 x (1 − α) % confidence limit

Table 2. Example of calculation of total area of *Phellinus weirii* infection centers by intersection length method (see text for explanation).

Line no.	Total intersection length (m) grid 1	grid 2	grid 3	Totals
1	100	400	0	
2	200	0	200	
3	100	300	300	
4	300	100	200	
5	0	100	100	
length totals	$h_{.1} = 700$	$h_{.2} = 900$	$h_{.3} = 800$	$h_{..} = 2400$
	$h_{.1}^2 = 490000$	$h_{.2}^2 = 810000$	$h_{.3}^2 = 640000$	$h_{.1}^2 + h_{.2}^2 + h_{.3}^2 = 1940000$

for the true area of root rot where t_0 is the t-value corresponding to a probability of $\alpha / 2$ in the upper tail of the t-distribution with m — 1 degrees of freedom.

Example

The following is an example for calculating total and variance of the infection area, using intersection length.

Sum intersection length for each transect line (Table 2). From Table 2, $h_{.1} = 700$, $h_{.2} = 900$, and $h_3 = 800$. If grid interval (d) = 1000, then from (3A) and (4A).

$$\hat{A}_{TOT} = 1000 \times 2400 / 3$$

$$= 800000 \text{ m}^2$$

and

$$\hat{V}_{TOT} = (1000)^2 \left[1940000 - (2400)^2/3\right] / 3 \times 2$$

$$= (1000)^2 \times 20000/6 = 3.33 \times 10^9 \text{ m}^2$$

From equation (5), 90% confidence intervals are given by

$$800000 \pm (2.92) \times (3.33 \times 10^9)^{1/2}$$

$$= 800000 \pm 168000 \text{ m}^2$$

$$= 80 \pm 16.8 \text{ ha}$$

Estimate of total root rot area and number of infection centers by probability of occurrence method

The probability p with which an infection center is included in the sample is w/d, where w is the projection length and d is the grid interval (Bloomberg *et al.*, 1980 Appendix 2). Let N_s denote the total number of infection centers in a stand belonging to size class C_s. Then for each grid j, an unbiased estimate of the number of infection centers in size class s is obtained from

Table 3. Table for recording numbers of *Phellinus weirii* infection centers by size class using probability of occurrence method (see text for explanation).

Grid 1		Grid 2		Grid 3	
d/w	size class no.	d/w	size class no.	d/w	size class no.
3.0	*1*				
2.1	*1*				
3.5	*2*				
3.1	*2*				
3.4	*2*				
5.1	*3*				
5.2	*3*				
5.4	*4*				

Note: There will not, in general, be the same number of entries for each of the grid systems.

$$\hat{N}_{sj} = d \sum^{n} 1/w \qquad (5)$$

and the area of infection centers in the size class from

$$\hat{A}_{sj} = d \sum^{n} a_{sj}/w \qquad (5A)$$

where d is the grid interval, a_{sj} is the area of infection centers of size class s in grid j, and n is the number of infection centers encountered.

Since p = w/d, (5) and (5A) can be rewritten as

$$\hat{N}_{sj} = \sum^{n} 1/p \text{ and } \hat{A}_{sj} = \sum^{n} a/p \qquad (6)$$

With m grids there will be m estimates $\hat{A}_{sl},\ldots,\hat{A}_{sm}$ and $\hat{N}_{sl},\ldots,\hat{N}_{sm}$ of area and number of infection centers, respectively. These estimates are independent and can be averaged to obtain a more precise estimate $\hat{A}_s$ and $\hat{N}_s$

$$\hat{N}_s = (\hat{N}_{sl} + \ldots + \hat{N}_{sm})/m, \qquad (7)$$

$$\hat{A}_s = (\hat{A}_{sl} + \ldots \hat{A}_{sm})/m \qquad (7A)$$

where m is the number of grid systems used for size classification and $\hat{N}_{sj}$ is defined as in (6) for

Table 4. Example of calculation of numbers and variance of *Phellinus weirii* centers by size class using probability of occurrence method (see text for explanation).

Size class	Grid 1	Grid 2	Grid 3	Total all grids	Sum of squares	Mean number ($\hat{N}_s$)
1	5.1	4.7	3.0	12.8	57.1	4.3
2	10.0	8.1	12.6	30.7	324.4	10.3
3	10.3	9.5	11.0	30.8	317.3	10.3
4	5.4	4.5	6.0	15.9	85.4	5.3

$j = 1,2,\ldots, m$. The variance of $\hat{N}_s$ is estimated by

$$\hat{V}_s = \sum_{j=1}^{m} (N_{sj} - \hat{N}_s)^2 \,/m\,(m-1). \qquad (8)$$

Variance of area estimated by size class is obtained by substituting $\hat{A}$ for $\hat{N}$.

Example

For each infection center, record its size class and value for probability of occurrence (d/w, Table 3). For each grid, total all the d/w values for each size class. Accumulate the sums and sums of squares of the row entries (Table 4), recording the totals. The estimates of number of centers in each size class ($\hat{N}_s$) for s = 1,....,m are obtained by dividing the row totals by the number of grids (m).

e.g. $N_3 = (10.3 + 9.5 + 11.0)/3$

$$\hat{V}_s = \left[\sum_{j=1}^{m} (N_{sj})^2 - (\sum_{j=1} N_{sj})^2 /m\right] /m\,(m-1),$$

thus, the entries of the last two columns of Table 4 can be substituted into (8) to obtain $\hat{V}_s$ for $s = 1,\ldots, m$.

Assuming four size classes and three grids, Table 4 shows estimates of $\hat{N}_s$ (s = 1, 2, 3, 4) the variance estimated for $\hat{V}_1$, $\hat{V}_2$, $\hat{V}_3$ and $\hat{V}_4$.

$$\hat{V}_1 = 57.1 - (12.8)\ ^2/3\ /\ (3 \times 2) = 0.41$$

$$\hat{V}_2 = 324.4 - (30.7)\ ^2/3\ /\ (3 \times 2) = 1.71$$

$$\hat{V}_3 = 317.3 - (30.8)\ ^2/3\ /\ (3 \times 2) = 0.18$$

$$\hat{V}_4 = 85.4 - (15.9)\ ^2/3\ /\ (3 \times 2) = 0.19 \qquad (10)$$

Estimates of total area of infection centers in each size class are obtained by multiplying the probability of occurrence (d/w) of each center by its area. The products are entered in Table 3 and their sums in Table 4, as for d/w. Calculation of mean and variance are the same as for estimates of numbers.

Estimates of area of individual infection centers

Three methods of area estimation have been described (Bloomberg *et al.* 1980).

a) Linear Method

This estimate ($\hat{A}_{LIN}$) is analogous to $\hat{A}_{TOT}$

applied to an individual infection center rather than the stand. The calculations are also analogous to those for estimating $\hat{A}_{TOT}$ (see Intersection Length Method).

Let l_i be the measurement distance across the infection center (Fig. 4) at the i^{th} measurement interval, then

$$\hat{A}_{L1} = d_L \times l_1$$

$$\hat{A}_{L2} = d_L \times l_2$$

$$\vdots$$

$$\hat{A}_{Ln} = d_L \times l_n$$

where d_L is the measurement interval and n is the number of intervals. The estimates $\hat{A}_{L1}, \ldots, \hat{A}_{Ln}$ are independent, unbiased estimates of the area of infection center. The average of these estimates is

$$\hat{A}_{LIN} = (\hat{A}_{L1} + \hat{A}_{L2} + \ldots \hat{A}_{Ln}) / n$$

b) Radial Method

The radial sampling estimate $\hat{A}_{RAD}$ is the average of m independent area estimates $\hat{A}_{R1}, \ldots, \hat{A}_{Rm}$, calculated from the lengths of radial lines from the measurement point of an infection center to its boundary (Fig. 3). Let l_i denote the length of the i^{th} radial line and θ_{RAD} denote the angle in radians between consecutive radial lines, and n the number of lines, then

$$Z_i = l_i^2 \text{ for } i = 1, 2, \ldots, n$$

$$\hat{A}_{R1} = \tfrac{1}{2} \times \theta_{RAD} \times Z_{.1}$$

$$\hat{A}_{Rm} = \tfrac{1}{2} \times \theta_{RAD} \times Z_{.m} \, .$$

The circular sampling estimate $\hat{A}_{RAD}$ is defined by

$$\hat{A}_{RAD} = (\hat{A}_{R1} + \ldots + \hat{A}_{Rm}) / m$$

c) Regression Method

The regression method estimates areas of rectangularly measured infection centers using regression between radially or linearly measured areas and areas measured by the product (w . z) of projection on baseline (w) and mid-length (z) (Bloomberg *et al.* 1980). Although w . z will be zero only when the area is zero, if too few small infection centers are measured, a statistically significant non-zero intercept could result. Area estimates will therefore be more accurate by inclusion of the intercept in the linear equation (Snedecor 1953).

The regression coefficient is computed from the rectangularly measured area of each infection center in the regression sample as independent variable and its radially or linearly measured area as dependent variable. $\hat{A}_{RAD}$ or $\hat{A}_{LIN}$. The projection w on the baseline L is the sum of the left and right projection lengths of an infection center and z denotes the mid-length, then

$$V = w \, . \, z$$

and

$$\hat{A}_{REG} = a + b \, . \, V$$

where a is the intercept value, significantly different from zero, b is the regression coefficient and $\hat{A}_{REG}$ is the area estimate using the regression method. This estimate is applied to infection centers measured by the rectangular method only. Fifteen to 30 infection centers should be included in the regression sample, distributed over the size range.

Computer Analysis of Results

A computer program called RRSAMP written in FORTRAN IV was developed to calculate estimates and confidence limits of root rot area and numbers of infection centers in each stand. The program calculates the following values (page numbers refer to selected computer output pages in Appendix 3):

1. Estimates of total root rot area by intersection length method

Total intersection length by grids and transect lines, and averages for the sample (p. 11, Table 1).

Estimates of total root rot area by grids and lines, based on the intersection length totals, and average root rot area for the stand (p. 11, Table 3).

Coefficient of variation, standard error and confidence limits (p = .05) for the estimated root rot area in stand (p. 11, Table 2).

2. Estimates of total root rot area by probability of occurrence method

Projection length and probability of encounter (Bloomberg *et al.* 1980) for each infection center (p. 10). Total root rot area in sample by grid and lines and center size class, based on probability of encounter and averages for sample (p. 11, Table 1). Estimates of total root rot area in stand by grid, lines and center size class, based on probability of occurrence and average area for stand (p. 11, Table 3). Coefficient of variation, standard error and confidence limits (p = .05) for the estimated root rot area for stand (p. 11, Table 2).

3. Estimates of total number of infection centers by size class

Number of root rot centers in sample by grid, line and infection center size class, based on probability of encounter (p. 15, Table 1).

Estimates of number of root rot centers in stand by grid, line and size class, based on probability of occurrence, and average for stand (p. 15, Table 3).

Coefficient of variation, standard error and confidence limits (p = .05) for number of root rot centers in each size class in the stand (p. 15, Table 2).

4. Estimates of infection center areas

Areas of infection centers measured by rectangular, linear and radial methods.

Equation for regression of linearly or radially measured on rectangularly measured infection center area and tests for significance (p. 2).

Estimates of area of rectangularly measured infection centers by application of regression equation (p. 10).

Input file preparation

Inputs for program RRSAMP are assigned to logical unit number 5 in a file named RRSAMP.IN. New data, normally on punched cards, is transferred to this file. Data consist of 4-11 control cards (survey specifications), followed by field data cards. Format for control cards must conform to that specified in the Data Analysis Inputs form (Fig. 12). Format for field measurement data follows that of the tally sheet (Fig. 6).

The following control information must be supplied to execute the program (numbers in brackets refer to column numbers on Data Analysis Input form).

Card 1. – Survey reference number (1-8), consisting of a four-character stem and a three character extension, e.g. BCMF.B10.

– Survey measurement units (9-16), e.g. meters, feet.

– Stand measurement units (17-24), e.g. hectares, acres.

– Survey / stand measurement units conversion factor (25-34), e.g. 10000 for metres to hectares.

– Map scale in units per inch (35-40). (if map option is requested).

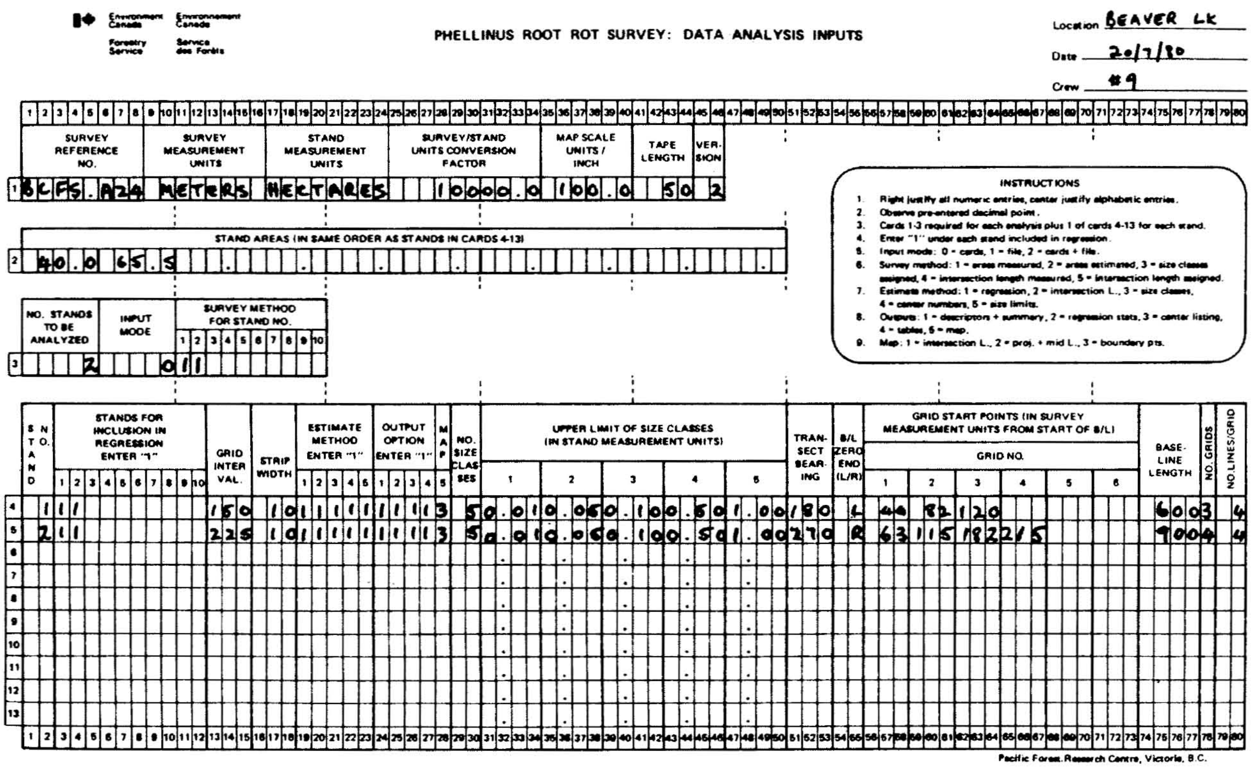
– Length of tape used in measuring transect distance (41-44).

– Version number (45-46). Default is 2, *i.e.,* current version of program. Version no. 1 refers to a previous tally sheet format (Bloomberg, 1978).

Card 2. – Stand area to one decimal place. If more than one stand is being analyzed, the areas must be in the same order as specified in Card 4.

Card 3. – Number of stands to be analyzed (1-5). Up to 10 stands can be analyzed in one analysis.

– Data input mode (10). Enter 0 for card input of new data, 2 for input from existing data in file CENTER.DTA (see Output


Environment Canada / Environnement Canada — Forestry Service / Service des Forêts

PHELLINUS ROOT ROT SURVEY: DATA ANALYSIS INPUTS

Location BEAVER LK
Date 20/7/80
Crew #9

SURVEY REFERENCE NO. | SURVEY MEASUREMENT UNITS | STAND MEASUREMENT UNITS | SURVEY/STAND UNITS CONVERSION FACTOR | MAP SCALE UNITS / INCH | TAPE LENGTH | VERSION

1 BCFS.A24 METERS HECTARES 10000.0 100.0 50 2

STAND AREAS (IN SAME ORDER AS STANDS IN CARDS 4-13)

2 40.0 65.5

NO. STANDS TO BE ANALYZED | INPUT MODE | SURVEY METHOD FOR STAND NO.

3 2 011

INSTRUCTIONS
1. Right justify all numeric entries, center justify alphabetic entries.
2. Observe pre-entered decimal point.
3. Cards 1-3 required for each analysis plus 1 of cards 4-13 for each stand.
4. Enter "1" under each stand included in regression.
5. Input mode: 0 = cards, 1 = file, 2 = cards + file.
6. Survey method: 1 = areas measured, 2 = areas estimated, 3 = size classes assigned, 4 = intersection length measured, 5 = intersection length assigned.
7. Estimate method: 1 = regression, 2 = intersection L., 3 = size classes, 4 = center numbers, 5 = size limits.
8. Outputs: 1 = descriptors + summary, 2 = regression stats, 3 = center listing, 4 = tables, 5 = map.
9. Map: 1 = intersection L., 2 = proj. + mid L., 3 = boundary pts.

STAND NO. | STANDS FOR INCLUSION IN REGRESSION ENTER "1" | GRID INTER VAL. | STRIP WIDTH | ESTIMATE METHOD ENTER "1" | OUTPUT OPTION ENTER "1" | MAP | NO. SIZE CLASSES | UPPER LIMIT OF SIZE CLASSES (IN STAND MEASUREMENT UNITS) | TRANSECT BEARING | B/L ZERO END (L/R) | GRID START POINTS (IN SURVEY MEASUREMENT UNITS FROM START OF B/L) GRID NO. | BASE-LINE LENGTH | NO. GRIDS | NO. LINES/GRID

4 111 150 10 1 1 1 1 1 1 1 1 1 3 5 0.010 .050 .100 .501 .00 180 L 44 82 120 6003 4
5 211 225 10 1 1 1 1 1 1 1 1 1 3 5 0.010 .050 .100 .501 .00 270 R 63 115 182 215 1004 4

Pacific Forest Research Centre, Victoria, B.C.

Fig. 12. Root rot survey data analysis input form. This form must be completed according to specifications (see text) for each survey analysis.

files), 3 if new data from cards is to be combined with existing data on file.

— Survey method option for each stand to be analyzed (see Introduction).

Cards 4 - 13.

— One card is punched for each stand in the analysis. Each card contains the following information:

— Stand number (1-2). This can be any one or two digit number. Stands must be in same order as in card 2.

— Stands to be included for calculation of regression of infection center area rectangular area (3-12, see Introduction). If only one stand is to be analyzed, enter "1" under Stand 1 (column 3) on card 4. If more than one stand is analyzed, enter "1" under Stand 2 (column 2) on card 5 for second stand, "1" under Stand 3 (column 5) on card 6 for third stand, and so on. If data from several stands are to be pooled for calculating the regression, enter "1" under each stand included. For example, if the regression for Stand no. 1 is to be calculated from data of Stand nos. 1, 2 and 3, enter "1" under Stands 1, 2 and 3 (columns 3, 4, 5) on card for Stand 1.

— Grid interval in survey measurement units (13-15).

— Strip width in survey measurement units (16-18).

— Infection center area estimate method (19-23). Enter "1" under code for estimate method. Codes denote the following methods: (1) regression, (2) intersection length, (3) probability of occurrence - area, (4) probability of occurrence - numbers, (5) area by size classes.

- Output option (24-28). Enter "1" under code for required output. Codes denote the following outputs: (1) survey descriptors plus summary table, (2) regression statistics, (3) listing of infection centers and measurements, (4) computation tables. Enter map option (see Mapping Options) under code 5.

- Number of size classes (29-30). Enter 1 - 5.

- Upper limit of size classes (31-50). Enter size class limit in stand measurement units (hectares, acres, etc.) to two decimal places in ascending order of class size.

- Transect bearing (51-53). Enter true north bearing of transect lines for baseline.

- Zero end of baseline (54-55). Enter L or R to denote at which end of baseline (facing direction 1) survey commences, *i.e.,* Grid 1, line 1.

- Grid start points (56-73). Enter grid start points (see Office Preparations) in survey measurement units.

- Baseline length (74-77). Enter baseline length in survey measurement units.

- Number of grids (78).

- Number of lines per grid (79-80).

All entries in the data analysis inputs form must conform to the format including pre-entered decimal points. All numeric data must be right-justified. All aphabetic variables, *i.e.,* reference number, survey and stand units, must be center-justified in the field.

Output Files

Analysis Results File

Results of analysis are assigned to logical unit number 6 in a file named RRSAMP.OUT which must be directed to an output device.

The following results are produced by the computer analysis (page numbers refer to selected computer output pages in Appendix 3).

1) A survey descriptor page showing all specification information (1).

2) Calculation of the regression equation including analyses of variance tables for significance of regression and intercept (2). Scatter plot of regression points (3).

3) A table showing measurements, area estimates, measurement method, and probability of encounter for each infection center measured (10).

4) Tables of sample estimates and total stand estimates of root rot area and calculations of confidence limits for intersection length (11), and probability of occurrence (12) methods.

5) Tables of estimated numbers of infection centers by size class (15).

6) A summary table showing estimated area of root rot and confidence limits in each stand by intersection length or by center size class (18).

7) A map showing baseline, transect lines (numbered by grid and line), and measurement points of infection centers as recorded on the tally sheet (Appendix 2).

Center Data File

A file containing measurement data for each infection center is created by each run and is assigned to logical unit number 2, named CENTER.DTA. This file is used only if existing data are to be merged with new data (see Input File Preparation, card 3). This option is specified by setting input mode at code 2. Selection of specific files for merging is achieved by transferring the appropriate file to CENTER.DTA.

Program RRSAMP is written in FORTRAN IV for use on a Digital Equipment Corporation PDP 11/45 and requires a total of 47.3 K decimal words of memory. The program consists of a main program and 27 subroutines. Program structure is shown in Figure 13. Descriptions of subroutine operations are given in Appendix 1. A program listing is given in Appendix 4.

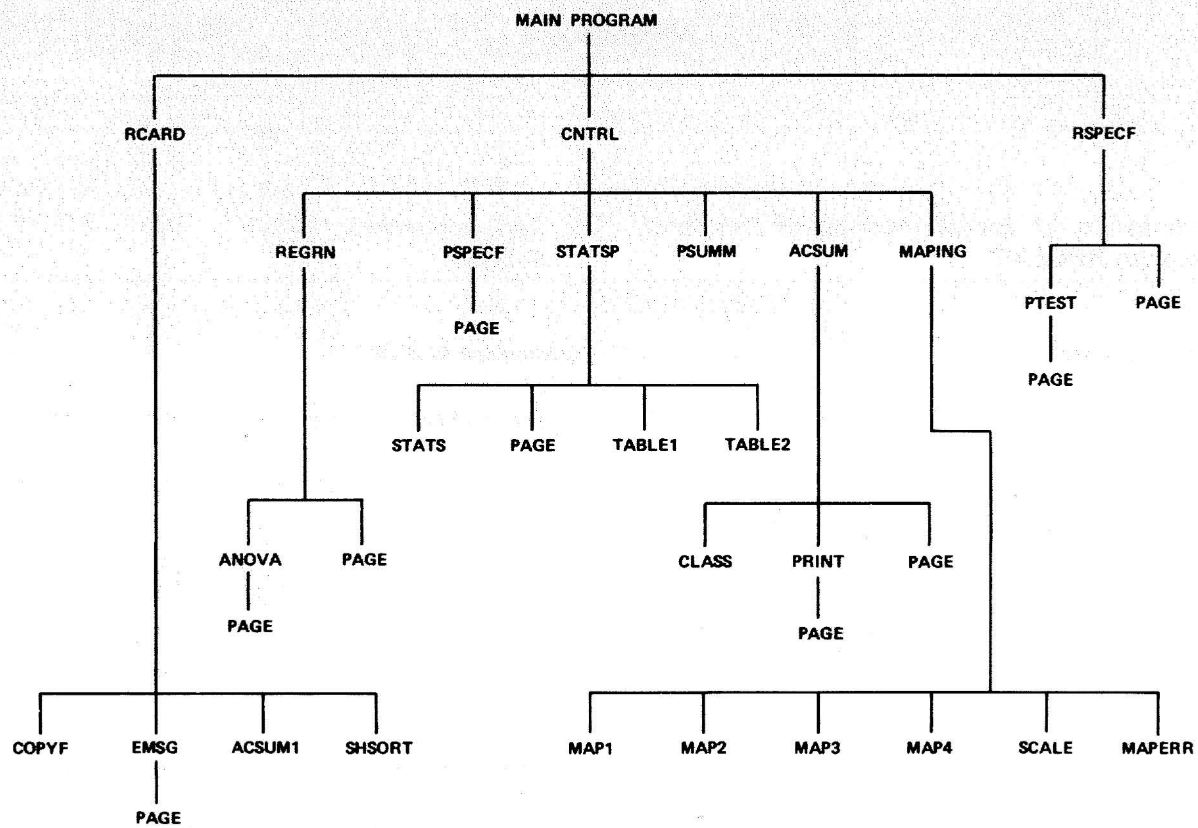


Fig. 13. Structure of computer program RRSAMP for analyzing survey data, showing names of subroutines and order in which they are called.

ACKNOWLEDGMENTS

Technical assistance of A.L.S. Johnson and G. Reynolds is gratefully acknowledged. Helpful suggestions on presentation were made by L.S. Safranyik.

LITERATURE CITED

Anon. 1975. Forest disease priorities in the Pacific Northwest. Special Task Force of the Disease Comm., Pacific Northwest Forest Pest Action Council. Mimeo Rept. 15 pp.

Bloomberg, W.J. 1978. Outline of root rot survey - revised May 1978. Mimeo. rept.

Bloomberg, W.J., P.M. Cumberbirch and G.W. Wallis. 1980. A ground-survey method for estimating loss caused by *Phellinus weirii* root rot. I. Development of survey design. Can. For. Serv. Rept. BC-R-3, 23 p.

Childs, T.W. and K.R. Shea. 1967. Annual losses from disease in Pacific Northwest forests. U.S. For. Serv. Res. Bull. PNW-20.

Cochran, W.G. 1963. Sampling techniques. John Wiley and Sons, New York, 411 p.

Snedecor, G. W. and W. G. Cochran, 1967. Statistical Methods. Iowa Sta. Univ. Press, Ames, Iowa.

Wallis, G.W. 1967. *Poria weirii* root rot. *In* Important insects and diseases of mutual concern to Canada, the United States and Mexico. Canada Dept. Forest. and Rural Development.

Wallis, G.W. 1967. *Phellinus (Poria) weirii* root rot. Detection and management proposals in Douglas-fir stands. Envir. Canada, Forest Tech. Rept. 12. 16 p.

APPENDIX 1

Description of source modules of computer program RRSAMP

MAIN program

Assigns the following files:
- Scratch file for sorting records,
- Sorted record file,
- File to hold inputs,
- File to hold outputs,
- File to hold map.

Calls subroutines RSPECF, RCARD, CNTRL.

Subroutine RSPECF

Reads survey specifications from input file,

Checks validity and returns error messages on specifications.

Subroutine RCARD

Reads survey data from input file, calculates areas of individual infection centers, accumulates sums for regression analysis, and creates a sorted disk file of records,

Calculates intersection length,

Calculates area of centers by regression, radial, or linear method, depending on measurement method specified,

Tests for inclusion of centers in regression analysis,

Accumulates sums, sum squares, and cross-products for regression analysis,

Writes processed record data on a scratch file,

Sorts scratch file records in ascending order by stand, grid, line and center.

Writes sorted file records on sorted file,

Calls subroutines ACSUM1, EMSG, COPYF, SHSORT.

Subroutine ACSUM1

Accumulates sums, sum squares, and cross-products of infection center areas as measured by rectangular, radial, or linear for use in calculation of regression parameters.

Subroutine EMSC

Prints error messages for errors in input file.

Subroutine COPYF

Copies existing file of sorted infection center data (if any) on to scratch file for merging with new data.

Subroutine SHSORT

Sorts centers in ascending order by stand, grid, line and center number.

Subroutine CNTRL

Controls order of all calculations and print-outs according to specified options in the input file,

Calls subroutines PSPECF, REGRN, ACSUM, STATSP, PSUMM, MAPING.

Subroutine PSPECF

Prints the specifications for each analysis.

Subroutine REGRN

Controls the regression calculations for regression of infection center areas measured by radial or linear methods on rectangularly measured areas,

Identifies stands to be included in the regression analysis and prints error message if invalid stand numbers are specified,

Calls subroutine ANOVA.

Subroutine ANOVA

Calculates intercept, regression coefficient, F-test, standard error and confidence limits,

Prints regression equation and analysis of variance table,

Arranges pairs of independent and dependent variables for display as a scatter plot,

Calls subroutine SCPLOT.

Subroutine SCPLOT

Produces plot of infection center areas measured by radial or linear methods over areas measured by rectangular method.

Subroutine ACSUM

Reads the sorted disk file records for each infection center, accumulates numbers and areas of centers and prints a listing of all centers measured,

Identifies data format specified,

Identifies measurement method,

Identifies regression option and estimates area by regression method,

Identifies interection length measurement method and accumulates intersection lengths,

Identifies option for estimating infection center area by size class and assigns size class to each center. Accumulates areas by size class,

Identifies option for estimating number of centers by size class and accumulates number of centers by size class,

Calls subroutines CLASS, PRINT.

Subroutine CLASS

Determines to which size class each infection center belongs.

Subroutine PRINT

Prints out list of center measurements and area estimates.

Subroutine STATSP

Calculates and prints sample estimates of area and number of infection centers ("Table 1"), estimates of variation ("Table 2"), estimates of total area and numbers of centers ("Table 3"), and confidence limits,

Calculates sampling fraction to estimate total from sample data,

Calls subroutine STATS, TABLE1, TABLE2.

Subroutine STATS

Calculates sample estimates of area and numbers of infection centers.

Subroutine TABLE1

Prints a table heading with row and column headings for Tables 1 and 3.

Subroutine TABLE2

Calculates standard error, coefficient of variance and confidence limits for total estimates. Prints Table 2.

Subroutine PSUMM

Prints out summary of total estimates.

Subroutine MAPING

Reads mapping inputs from the input file and writes a map of the infection centers on a file, then transfers file to the output file,

Calls subroutines MAPERR, MAP1, MAP2, MAP3, MAP4.

Subroutine MAPERR

Checks mapping inputs for faults and returns error messages.

Subroutine MAP1

Draws a map of the baseline and transect lines on which a map of the infection centers can be superimposed,

Calls subroutine SCALE.

Subroutine SCALE

Places scales along the X and Y axes of the map.

Subroutine MAP2

Superimposes boundary points of rectangularly measured infection centers on the transect lines.

Subroutine MAP3

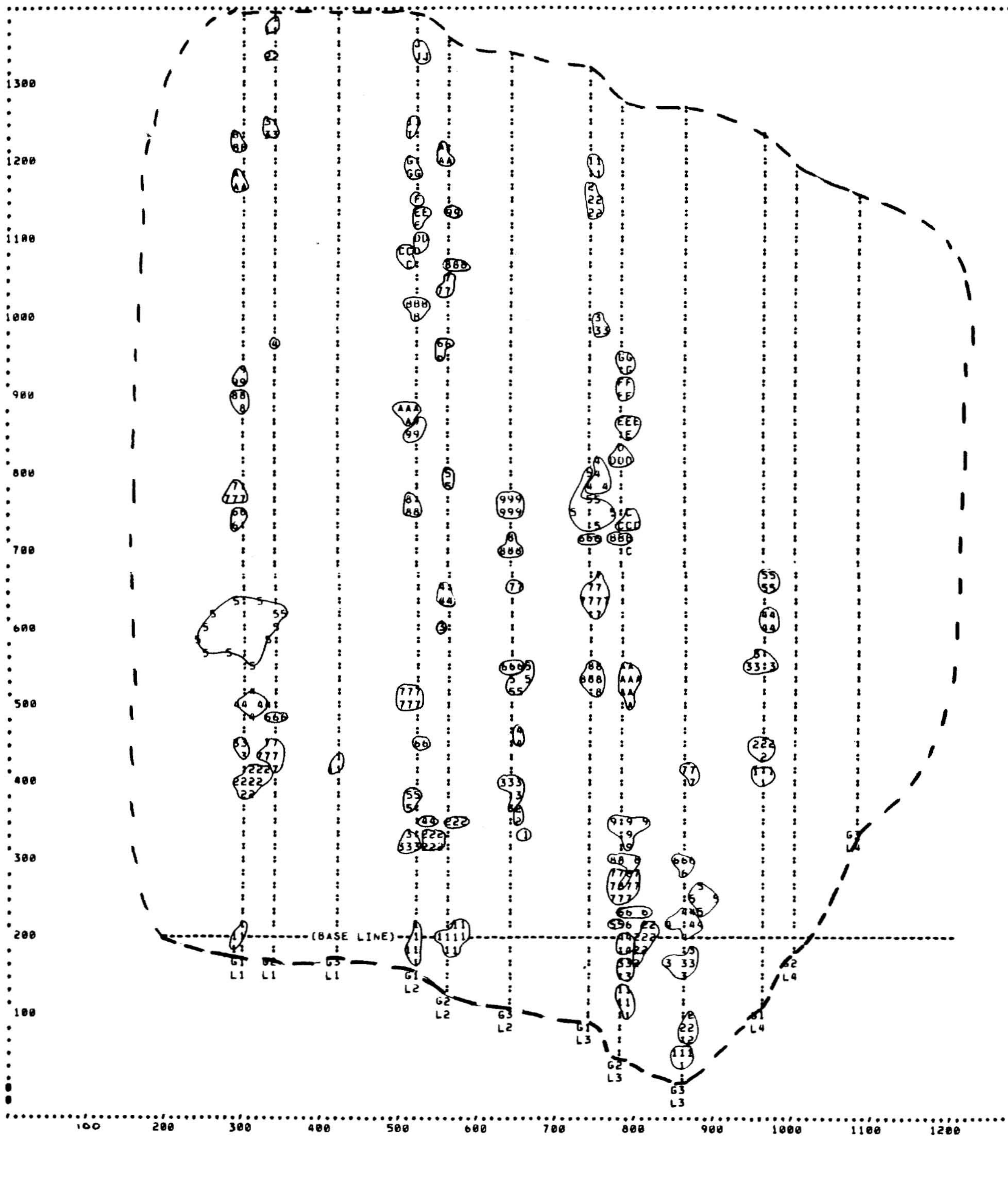
Superimposes boundary points of radially measured infection centers on the transect lines.

Subroutine MAP4

Superimposes boundary points of linearly measured infection centers on the transect lines.

APPENDIX 2

PROGRAM: ROOT ROT SAMPLE SURVEY ANALYSIS   PAGE 9
MAP OF INFECTION CENTRES   SCALE : 1 INCH = 100.0 METERS   SEE LIST FOR CENTER SYMBOLS



APPENDIX 3-1

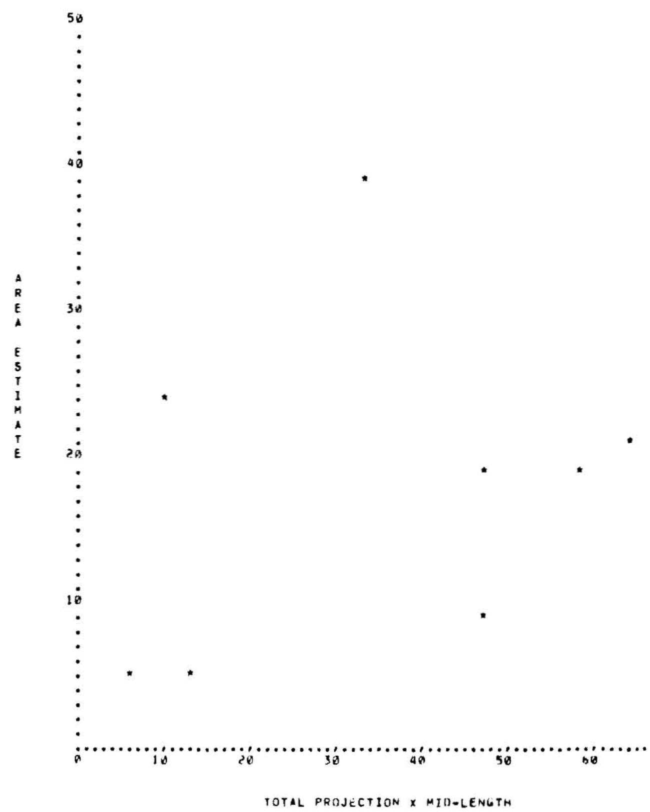
```
PROGRAM:                          ROOT ROT SAMPLE SURVEY ANALYSIS                          PAGE      1

                                  SURVEY REFERENCE NO.:BCF5.U28

   NUMBER OF ANALYSES REQUESTED:  1

   DATA ENTRY MODE: CARDS

                                 * * * * * *  ANALYSES REQUESTED  * * * * * *

ANALYSIS  1:    STAND NUMBER:  1

                SURVEY METHOD:       INFECTION CENTERS MEASURED
                SAMPLING DESIGN:      3 GRIDS X    5 TRANSECTS

                STATISTICS REQUESTED:

                     REGRESSION ESTIMATE OF INDIVIDUAL CENTER SIZES
                     INTERSECTION LENGTH AREA ESTIMATE
                     TOTAL AREA ESTIMATES BY CENTER SIZE CLASS
                     TOTAL NUMBER ESTIMATES BY SIZE CLASS

                OUTPUTS REQUESTED:

                    DESCRIPTORS AND SUMMARY
                    REGRESSION STATISTICS
                    LIST OF CENTERS MEASURED
                    COMPUTATION TABLES
                    MAP  (INTERSECTION LENGTH ONLY)

                STANDS TO BE INCLUDED IN REGRESSION CALCULATIONS OF CENTER SIZES

                     STAND NUMBERS        :  1  2  3  4  5  6  7  8  9 10
                     REGRESSION INDICATOR:  1  0  0  0  0  0  0  0  0  0

                CENTER SIZE CLASSIFICATION ( HECTARES)

                     DETERMINED FROM UPPER BOUNDARY LIMITS:      0.05       0.10       0.50

                SURVEY MEASUREMENT UNITS:                      METERS

                BASELINE LENGTH:                                  1500

                GRID DISTANCE:                                   300.0

                STRIP WIDTH :                                     10.0

                GRID START POINTS FROM R END OF BASELINE            50         150        250

                TRANSECT BEARING:                                    0          TAPE LENGTH  100

                MAP SCALE:                                   100.0 METERS  TO 1 INCH

                STAND AREA UNITS:                             HECTARES

                STAND AREA:                                      177.0

                AREA UNITS CONVERSION FACTOR:                  10000.0
```

APPENDIX 3-2

```
PROGRAM:                          ROOT ROT SAMPLE SURVEY ANALYSIS                        PAGE     2

ANALYSIS 1:        * * * * * * * *   CENTER SIZE REGRESSION ANALYSIS   * * * * * * * *
     * * * THE REGRESSION EQUATION IS USED TO ESTIMATE AREA OF CENTERS FROM TOTAL PROJECTION AND MID-LENGTH * * *

        REGRESSION BASED ON STAND(S)    1

      INDEPENDENT VARIABLE:  X = TOTAL PROJECTION * MID-LENGTH

      DEPENDENT VARIABLE  :  Y = AREA

            ANOVA TABLE (ZERO INTERCEPT MODEL:  E(Y) = B * X)

      SOURCE              SS               DF                MS                 F

      SLOPE          292215.781            1            292215.781          8.412  (REFER TO F DISTRIBUTION
      RESIDUAL       312659.344            9             34739.926                 TABLE WITH   9
      TOTAL          604875.125           10                                       DEGREES OF FREEDOM)

      SLOPE:  B=   0.78896              STANDARD ERROR OF B =       0.272

      CONFIDENCE LIMITS OF ESTIMATE         + OR -  0.614790

      ***** REGRESSION IS SIGNIFICANT


            T TEST FOR INCLUSION OF THE INTERCEPT

      INTERCEPT           VARIANCE              DF              SE           T
        108.222  2705644176015359.8              8         52015808.0      0.00

      PREDICTION EQUATION:  Y = (   0.78896) * X
```

APPENDIX 3-3

```
PROGRAM:                              ROOT ROT SAMPLE SURVEY ANALYSIS
SCATTER PLOT FOR LINEAR REGRESSION           STAND NO.   1 ONLY
SCALE:  X AXIS:  1 UNIT =   7.6 SQ. METERS
        Y AXIS:  1 UNIT =  16.2 SQ. METERS
```



APPENDIX 3-4

ANALYSIS 1: * * * * * * * * CALCULATIONS FOR STAND NO. 1 * * * * * * * *
MEASUREMENTS IN METERS

CENTER ID.	MAP SYMBOL	INTERSECTION LENGTH	AREA ESTIMATE	TOTAL PROJECTION	MID-LENGTH	SIZE CLASS	SELECTION PROBABILITY	ESTIMATION METHOD
1 1 1	1	24.	706.	35.	16.	2	0.15000	RADIAL
1 1 2	2	30.	313.	20.	13.	1	0.10000	REGRESSION
1 1 3	3	6.	286.	15.	15.	1	0.08333	REGRESSION
1 1 4	4	5.	364.	18.	18.	1	0.09333	REGRESSION
1 1 5	5	22.	222.	12.	12.	1	0.07333	REGRESSION
1 1 6	6	30.	272.	16.	13.	1	0.08667	REGRESSION
1 1 7	7	4.	231.	12.	13.	1	0.07333	REGRESSION
1 1 8	8	5.	165.	6.	9.	1	0.06000	REGRESSION
1 1 9	9	25.	242.	13.	13.	1	0.07667	REGRESSION
1 1 10	A	5.	215.	15.	9.	1	0.08333	REGRESSION
1 1 11	B	55.	812.	27.	28.	2	0.12333	RADIAL
1 1 12	C	0.	132.	5.	6.	1	0.05000	REGRESSION
1 1 13	D	22.	286.	15.	15.	1	0.08333	REGRESSION
1 1 14	E	0.	242.	13.	13.	1	0.07667	REGRESSION
1 1 15	F	13.	187.	10.	10.	1	0.06667	REGRESSION
1 1 16	G	0.	250.	18.	10.	1	0.09333	REGRESSION
1 1 17	H	0.	317.	22.	12.	1	0.10667	REGRESSION
1 1 18	I	7.	203.	10.	12.	1	0.06667	REGRESSION
1 2 1	1	36.	313.	21.	21.	1	0.10333	RADIAL

APPENDIX 3-5

ANALYSIS 1: * * * * * ESTIMATION OF ROOT ROT AREA USING THE INTERSECT LENGTH METHOD * * * * *

STAND NO. 1

TABLE 1: ESTIMATE OF MEAN INTERSECTION LENGTH IN SAMPLE (METERS)

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	253.00	104.00	37.00	131.33
2	60.00	90.00	111.00	87.00
3	130.00	165.00	41.00	112.00
4	209.00	115.00	175.00	166.33
5	21.00	0.00	0.00	7.00
MEANS	134.60	94.80	72.80	100.73

TABLE 2: ESTIMATE OF VARIATION FOR TOTAL STAND

ESTIMATION METHOD	VARIANCE ESTIMATE	DEGREES OF FREEDOM	STANDARD DEVIATION	COEFFICIENT OF VARIATION	CONFIDENCE LIMIT (+OR-)
METHOD 1 (BY GRIDS)	29.436	2	5.426	0.359	10.433
METHOD 2 (BY LINES)	8.763	14	2.960	0.196	6.335

TABLE 3: ESTIMATES OF MEAN ROOT ROT AREA IN TOTAL STAND (HECTARES)

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	37.95	15.60	5.55	19.70
2	9.00	13.50	16.65	13.05
3	19.50	24.75	6.15	16.80
4	31.35	17.25	26.25	24.95
5	3.15	0.00	0.00	1.05
MEANS	20.19	14.22	10.92	15.11

TOTAL ESTIMATED ROOT ROT AREA FOR STAND 15.11 + OR - 6.33 HECTARES

APPENDIX 3-6

PROGRAM: ROOT ROT SAMPLE SURVEY ANALYSIS PAGE 12

ANALYSIS 1: * * * * * ESTIMATION OF ROOT ROT AREA BY CENTER SIZE CLASS USING AREA/PROBABILITY OF OCCURRENCE * * * * *

SIZE CLASS, 1 STAND NO. 1 SIZE CLASS = 0.00 - 0.05 HECTARES

TABLE 1: ESTIMATE OF MEAN CENTER AREA IN SAMPLE (SQUARE METERS)

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	48975.91	3016.95	10201.56	20731.47
2	14701.24	21357.69	23195.69	19751.54
3	18367.99	25713.19	18764.30	20940.49
4	47722.82	16622.00	31079.53	31808.12
5	3446.63	0.00	0.00	1148.88
MEANS	26642.92	13341.97	16648.22	18877.70

TABLE 2: ESTIMATE OF VARIATION FOR TOTAL STAND

ESTIMATION METHOD	VARIANCE ESTIMATE	DEGREES OF FREEDOM	STANDARD DEVIATION	COEFFICIENT OF VARIATION	CONFIDENCE LIMIT (+OR-)
METHOD 1 (BY GRIDS)	15.986	2	3.998	0.424	7.689
METHOD 2 (BY LINES)	3.880	14	1.970	0.209	4.215

TABLE 3: ESTIMATE OF MEAN CENTER AREA IN TOTAL STAND (HECTARES)

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	24.49	1.51	5.10	10.37
2	7.35	10.68	11.60	9.88
3	9.18	12.86	9.38	10.47
4	23.86	8.31	15.54	15.90
5	1.72	0.00	0.00	0.57
MEANS	13.32	6.67	8.32	9.44

TOTAL ESTIMATED ROOT ROT AREA FOR SIZE CLASS 1: 9.44 + OR - 4.22 HECTARES

APPENDIX 3 - 7

PROGRAM: ROOT ROT SAMPLE SURVEY ANALYSIS PAGE 15

ANALYSIS 11: * * * * * ESTIMATION OF TOTAL NUMBER OF CENTERS BY SIZE CLASS USING PROBABILITY OF OCCURRENCE

SIZE CLASS, 1 STAND NO. 1 SIZE CLASS = 0.00 - 0.05 HECTARES

TABLE 1: ESTIMATE OF MEAN NUMBER OF CENTERS IN SAMPLE

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	208.37	13.04	35.39	85.60
2	64.93	87.12	83.36	78.47
3	88.13	106.76	77.51	90.80
4	193.43	61.61	141.01	132.02
5	18.45	0.00	0.00	6.15
MEANS	114.66	53.71	67.45	78.61

TABLE 2: ESTIMATE OF VARIATION FOR TOTAL STAND

ESTIMATION METHOD	VARIANCE ESTIMATE	DEGREES OF FREEDOM	STANDARD DEVIATION	COEFFICIENT OF VARIATION	CONFIDENCE LIMIT (+OR-)
METHOD 1 (BY GRIDS)	34074.418	2	184.593	0.470	354.975
METHOD 2 (BY LINES)	6847.035	14	82.747	0.211	177.078

TABLE 3: ESTIMATE OF MEAN NUMBER OF CENTERS IN TOTAL STAND

LINE	GRID 1	GRID 2	GRID 3	MEANS
1	1041.64	65.22	176.94	428.00
2	324.64	435.58	416.78	392.33
3	440.64	533.00	387.56	454.00
4	967.16	308.03	705.04	660.08
5	92.26	0.00	0.00	30.75
MEANS	573.31	268.53	337.26	393.03

TOTAL ESTIMATED NO. OF CENTERS FOR SIZE CLASS 1:393.0 + OR -177.1

APPENDIX 3 - 8

PROGRAM: ROOT ROT SAMPLE SURVEY ANALYSIS PAGE 16

**********SUMMARY OF ESTIMATES OF ROOT ROT AREA**********

SURVEY REFERENCE NO.: BCF5,U28

STAND NO. 1

AREA 177.0 HECTARES

METHOD OF ESTIMATE	CENTER SIZE CLASSES HECTARES	TOTAL AREA HECTARES	% OF TOTAL	% OF STAND	CV(%) LINES	NO. OF CENTERS	% OF TOTAL	AV.NO. /A.U.	CV(%) LINES
INTERSECTION LENGTH		15.1 + OR - 6.3		8.5	19.6				
CENTER SIZE CLASS	0.00 - 0.05	9.4 + OR - 4.2	81.6	5.3	28.9	393.0 + OR -177.1	93.2	2.2	21.1
	0.05 - 0.10	1.6 + OR - 2.2	14.1	0.9	31.8	24.4 + OR - 6.9	5.8	2.1	31.5
	0.10 - 0.50	0.5 + OR - 1.1	4.3	0.3	100.0	4.1 + OR - 8.0	1.0	2.0	100.0
	TOTAL	11.6		6.5		421.6		2.4	

* * * * * * * * * END OF JOB * * * * * * * * *

APPENDIX 4

```
C**********************************************************************
C       PROGRAM RRSAMP ACCEPTS OATA RECORDED ON ROOT ROT SURVEY ANALYSIS
C       INPUT AND FIELD MEASUREMENT SHEETS. IT PERFORMS ANALYSES ON THE
C       DATA AS DESCRIBED IN THE FOLLOWING REFERENCE:
C
C       BLOOMBERG,W.J., P.M.CUMBERBIRCH & G.W.WALLIS.1980. A GROUND SURVEY
C       METHOD FOR ESTIMATING LOSS CAUSED BY PHELLINUS WEIRII ROOT ROT.
C       II.SURVEY PROCEDURES AND DATA ANALYSIS. REPORT NO. BCR-4, PACIFIC
C       FOREST RESEARCH CENTER,506 W. BURNSIDE RD., VICTORIA, B.C. V8Z 1M5.
C
C       PROGRAM WRITTEN BY W.J. BLOOMBERG AND P.M. CUMBERBIRCH,
C       LANGUAGE: FORTRAN IV
C       SYSTEM: PDP 11/45
C       MEMORY REQUIREMENT: 48 DECIMAL WORDS
C       LATEST UPDATE: SEPT. 1980
C**********************************************************************
C  MAIN PROGRAM TO CALL SUBROUTINES FOR SPECIFICATIONS
C  INPUT (RSPECF) AND CONTROL OF SUBSEQUENT CALCULATIONS
C  AND INPUT (CNTRL)
C
C**********************************************************************
C
C       VARIABLES CONTAINED IN COMMON BLOCKS.
C
C**********************************************************************
C       BLOCK /ALL/
C
C       REFNO       SURVEY REFERENCE NO.
C       MUNITS      SURVEY MEASUREMENT UNITS
C       SUNITS      STAND MEASUREMENT UNITS
C       FACTOR      STAND UNITS/SURVEY UNITS CONVERSION FACTOR
C       MANALY      NO. OF ANALYSES TO BE RUN
C       IOATA       INPUT DATA MODE
C       IANALY      ESTIMATE METHOD OPTIONS FOR EACH ANALYSIS
C       IREGR       STANDS FOR INCLUSION IN REGRESSION IN EACH ANALYSIS
C       IFORM       SURVEY MEASUREMENT OPTION
C       ISWTCH      OUTPUT OPTIONS
C       SIZE        STAND AREA
C       NGRID       NO. OF GRIDS IN EACH STAND
C       NTRANS      NO. OF TRANSECTS IN PER GRID
C       STRPW       STRIP WIDTH
C       DBTWN       GRID INTERVAL
C       CUTPTS      CUTPOINTS (UPPER SIZE LIMITS FOR EACH SIZE CLASS)
C       NCTPS       NO. OF CUTPOINTS
C       MCLASS      NO. OF SIZE CLASSES
C       SUMBNO      INFECTION CENTER SIZE CLASS LIMITS
C       SUMTOT      TOTAL ROOT ROT AREA BY SIZE CLASS
C       SUMNO       TOTAL NO. OF INFECTION CENTERS BY SIZE CLASS
C       SUMARE      TOTAL ROOT ROT AREA
C       B           REGRESSION COEFFICIENT
C       CEPT        REGRESSION INTERCEPT
C       FLAG        ERROR MESSAGE LABEL
C**********************************************************************
C
C       BLOCK /CK/
C
C       NGRD       GRID NO.
C       NLIN       LINE NO.
C       NCEN       CENTER NO.
C       XLEN       INTERSECTION LENGTH
C       OIAM       MID-LENGTH
C       AREA       CENTER AREA
C       METH       MEASUREMENT METHOD
C       PROJ       PROJECTION LENGTH
C       NCLASS     SIZE CLASS
C       PRB        PROBABILITY OF ENCOUNTER
C       NSTO1
C         -        I.D. OF PREVIOUSLY ENCOUNTERED CENTER
C       NCEN1
C       IVERS       PROGRAM VERSION
C**********************************************************************
C
C       BLØCK /MAP/
C
C       SCALMP        MAP SCALE
C       SCALX         X COORDINATE SCALE FACTOR
C       SCALY         Y     "         "     "
C       IMPLTH        MAP LENGTH
C       LBASE         BASELINE LENGTH
C       BLEND         ZERO END OF BASELINE
C       IBRNG         TRANSECT BEARING
C       LTAPE         TAPE LENGTH
C       STRREF        SURVEY START POINT
C       GSP           GRID START POINT
C       STRPNT        LINE START POINT
C       IBND1         Y COORDINATES OF TRANSECT LINE
C       LINDIR        LINE OIRECTION
C       SIOE          SIDE ON TRANSECT LINE
C      PT1        START OF INTERSECTION
C      PT2        END OF INTERSECTION
C      YOST12     DISTANCE TO L. PROJECTION TIE-POINT
C      YDST22     DISTANOE TO R. PROJECTION TIE-POINT
C      OTRDST     OISTANCE TO MID-LENGTH TIE-POINT
C      YOIST3     OISTANCE TO MEASUREMENT POINT TIE-POINT
C      XDIST3     OFFSET TO MEASUREMENT POINT
C      CNTANG     CENTER LINE ANGLE
C      PRJ1       LEFT PROJECTION
C      PRJ2       RIGHT PROJECTION
C      OIAM       MID LENGTH
C      NLINES     NO. MEASUREMENTS
C      DATA       MEASUREMENTS
C
C**********************************************************************
C
C
C**********************************************************************
C
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      DIMENSION SUMX(10),SUMY(10),SUMXY(10),SUMXX(10),SUMYY(10),NSAMP(10
     *),INOEX(10)
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRO,MLIN,NSTO,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBNO(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRIO(10),NTRNS(10)
      COMMON /MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTROST,PRJ1,PRJ2,
     *    DST12,YOST22,XOIST2,PT1,PT2,YOIST3,XOIST3,OIST,
     *              OATA,NLINES,LINOIR,XOIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLENO
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINOIR(6.10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 SIDE,L,R,BLENO(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/OHB/SUMXY,SUMXX,SUMYY,NSAMP,INOEX
      COMMON/CK/NGRO,NLIN,NCEN,XLEN,OIAM,AREA,METH,PRDJ,NCLASS,PRB,
     1NSTD1,NGRO1,NLIN1,NCEN1,IVERS
C
C  SET UP OIRECT ACCESS FILES FOR SORTING RECORDS
C
      CALL ASSIGN(1,'SCRATCH.TMP',11)
      CALL ASSIGN(2,'SY0:CENTER.DTA',14)
      DEFINE FILE 1(2000,19,U,IDUM1)
      DEFINE FILE 2(2000,19,U,IDUM2)
C
C  FOR OUTPUT AND INPUT FILES
C
      CALL ASSIGN(6,'RRSAMP.OUT',10)
      CALL ASSIGN(5,'RRSAMP.IN',9)
C
C       SET UP FILE TO HOLD MAP
C
        CALL ASSIGN(4,'MAP.OUT',7)
        OEFINE FILE 4(160,198,U,IDUM3)
C
C  BEGIN EXECUTION - GET PROBLEM SPECIFICATIONS
C
      CALL RSPECF
D      WRITE(6,1000)
1000   FORMAT(T10,'RSPECF OUT')
      IF(.NOT.FLAG)GOTO 100
C
C       TEST FOR INPUT MODE - CARDS OR DISC FILE
C
      IF(IOATA.EQ.1)GO TO 9
C
C       READ THE CARD OATA AND CREATE SORTED DISK FILE
      CALL RCARD
D     WRITE(6,2000)
2000  FORMAT(T10,'RCARD OUT')
C
C       TEST FOR ERROR STOP
C
      IF(.NOT.FLAG)GO TO 100
C  CALL CONTROLING PROGRAM FOR CALCULATIONS AND PRINTOUT
C
    9 CALL CNTRL
      IF(.NOT.FLAG)GOTO 100
      WRITE(6,10)
   10 FORMAT(//T30, '  * * * * * * * *  END OF JOB  * * * * * * * * ')
      STOP
  100 WRITE(6,101)
  101 FORMAT(//T30, ' * * * * *  ABNORMAL END  * * * * *')
      STOP
      END
C
C
C*************************************************************
      SUBROUTINE PAGE
C  THIS SUBROUTINE PRINTS THE TITLE AND PAGE NUMBER AT THE TOP
C  OF EACH PAGE OF PRINTOUT.
C*************************************************************
      DATA NPAGE/0/
      NPAGE=NPAGE+1
      WRITE(6,11)NPAGE
   11 FORMAT('1','PROGRAM:',30X,'ROOT ROT SAMPLE SURVEY ANALYSIS',30X,'P
     *AGE',I6)
      RETURN
      END
C
C*************************************************************
      SUBROUTINE CNTRL
C  THIS SUBROUTINE CONTROLS ALL CALCULATIONS AND PRINTOUTS
C
C*************************************************************
      DIMENSION IDEX1(10),IDEX2(10),KREGR(10)
      DIMENSION XMAT(6,20),TSMAT(10,6,20),SMAT(10,6,20)
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      DIMENSION SUMX(10),SUMY(10),SUMXY(10),SUMXX(10),SUMYY(10),NSAMP(10
     *),INDEX(10)
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IOATA,MANALY,
     * ISTD,NCTPS,MGRO,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/DHB/SUMXY,SUMXX,SUMYY,NSAMP,INDEX
      COMMON/SLT/TOTX,TOTY,TOTXY,TOTXX,TOTYY,NTOT
      COMMON/FKT/XMAT,TSMAT,SMAT,D
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 SIOE,L,R,BLEND(10)
      INTEGER DTRDST
      COMMON /MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,OIST,
     *              DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
C
C        ASSIGN SURVEY MEASUREMENT AND MAPPING METHOD
```

```
C
      DO 10 NANALY=1,MANALY
       IFM=IFORM(NANALY)
       IMAP=ISWTCH(NANALY,5)
C  PRINT THE SPECIFICATIONS FOR EACH ANALYSIS
      IF(ISWTCH(NANALY,1).EQ.1)CALL PSPECF
0      WRITE(6,1000)
1000   FORMAT(T10,'PSPECF OUT')
C  TEST FOR REGRESSION OPTION TO ESTIMATE AREAS OF ROOT ROT CENTRE
      IF(IANALY(NANALY,1).EQ.0)GOTO 21
      CALL REGRN(IERR)
C  TEST ERROR RETURN CODE
      IF(IERR.EQ.1)GOTO 10
C  CALCULATE AND ACCUMULATE AREAS OF ROOT ROT CENTRES AND PRINT
C  LIST OF CENTRES WITH THEIR MEASUREMENTS
   21 CALL ACSUM(IERR)
C  TEST ERROR RETURN CODE
      IF(IERR.EQ.1)GOTO 10
C  TEST FOR INTERSECTION LENGTH METHOD OPTION
      IF(IANALY(NANALY,2).EQ.0.AND.IFM.EQ.1)GOTO 20
      IF(IMAP.GT.0)CALL MAP(NANALY,IFM,IMAP)
      IF(.NOT.FLAG)RETURN
C  PRINT TABLES SHOWING SAMPLE STATISTICS (TABLE 1), ESTIMATE OF
C  VARIATION (TABLE 2), AND ESTIMATES FOR TOTAL STAND (TABLE 3)
      CALL STATSP(1,0)
C  TEST FOR METHOD OF AREA ESTIMATES BY SIZE CLASSES OPTION
   20 IF((IANALY(NANALY,3).EQ.0.AND.IFM.EQ.1).OR.IFM.GT.2)GOTO 30
C  PRINT TABLES 1-3 FOR EACH SIZE CLASS
      DO 40 KCLASS=1,MCLASS
   40 CALL STATSP(2,KCLASS)
C  TEST FOR METHOD OF CENTRE NUMBER ESTIMATES BY SIZE CLASSES OPTION
   30 IF((IANALY(NANALY,4).EQ.0.AND.IFM.EQ.1).OR.IFM.GT.3)GOTO 9
C  PRINT TABLES 1-3
      DO 60 KCLASS=1,MCLASS
   60 CALL STATSP(3,KCLASS)
C  PRINT SUMMARY OF ESTIMATED AREA AND NUMBERS OF ROOT ROT CENTRES
9     IF(ISWTCH(NANALY,1).EQ.1)CALL PSUMM
   10 CONTINUE
   50 RETURN
      END
C
C*********************************************************************
      SUBROUTINE RSPECF
C
C  THIS SUBROUTINE READS THE SPECIFICATION CAROS AND SCANS
C  THE INPUTS FOR VALIO VALUES
C*********************************************************************
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 SIDE,L,R,BLEND(10)
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1,IVERS
      COMMON/MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *   DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *              DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
       DATA L/'L'/,R/'R'/
      FLAG=.TRUE.
C  READ SPECIFICATIONS
      READ(5,11)REFNO,MUNITS,SUNITS,FACTOR,SCALMP,LTAPE,IVERS
C     SURVEY REFERENCE NO.,SURVEY MEASUREMENT UNITS, STAND MEASUREMENT
C     UNITS, CONVERSION FACTOR
      IF(IVERS.EQ.1.OR.IVERS.EQ.2)GO TO 5
      CALL PTEST
      WRITE(6,6)
    6 FORMAT(' ','INVALID VERSION NO.')
  5   READ(5,12)(SIZE(I),I=1,10)
C     AREA OF EACH STAND INCLUDED IN ANALYSIS
   12 FORMAT(10F5.1)
   11 FORMAT(3A8,F10.1,F6.1,I4,I2)
      READ(5,10)MANALY,IDATA,(IFORM(I),I=1,10)
C     NUMBER OF ANALYSES, DATA INPUT METHOD (0=CARDS, 1=DISC FILE)
   10 FORMAT(3X,I2,4X,I1,10I1)
C  TEST VALIDITY OF FORMAT CODES REQUESTED
      DO 20 I=1,10
      DO 19 J=1,6
      K=J-1
   19 IF(IFORM(I).EQ.K)GOTO 20
      CALL PTEST
      WRITE(6,21)
   21 FORMAT('0','CARD 1:',5X,'INVALIO FORMAT CODE')
      GOTO 30
   20 CONTINUE
C  TEST VALIOITY OF NUMBER OF ANALYSES REQUESTED
   30 IF((MANALY.LE.0).OR.(MANALY.GT.10))GOTO 31
      GOTO 33
   31 CALL PTEST
      WRITE(6,32)
   32 FORMAT('0','CARD 1:',5X,'NUMBER OF ANALYSES REQUESTED OUTSIDE RANG
     *E 1 TO 10')
C  TEST VALIDITY OF THE DATA INPUT OPTION
   33 DO 40 I=1,3
      K=I-1
   40 IF(IDATA.EQ.K)GOTO 41
      CALL PTEST
      WRITE(6,42)
   42 FORMAT('0','CARD 1:',5X,'INVALIO DATA ENTRY MODE SPECIFIED')
   41 IF(.NOT.FLAG)RETURN
      ICTPS=0
      II=1
C  INPUTS FOR STAND NO., CENTRE AREA BY REGRESSION METHOD OPTION,
C  GRID DISTANCE, AND STRIP WIDTH
      DO 80 I=1,MANALY
      IF(ICTPS.GT.4)II=II+1
      READ(5,101)ISTD(I),(IREGR(I,J),J=1,10),DBTWN(I),STRPW(I),
     *(IANALY(I,J),J=1,5),(ISWTCH(I,J),J=1,5),ICTPS,(CUTPTS(I,J),J=1,5),
     *IBRNG(I),BLEND(I),(GSP(I,J),J=1,6),LBASE(I),NGRID(I),NTRNS(I)
C
C      OMIT MAP OPTION FOR SURVEY METHOD 5
C
      IF(IFORM(I).EQ.5)ISWTCH(I,5)=0
      DSTWN(I) = DBTWN(I)
C  NO. OF CENTRE SIZE CLASSES AND UPPER BOUNDARIES OF EACH SIZE
C  CLASS
      II=II+1
      NCTPS(I)=ICTPS
 101  FORMAT(I2,10I1,2F3.0,2(5I1),I2,5F4.2,I3,1X,A1,6I3,I4,I1,I2)
C  TEST FOR INVALID STAND NUMBER
      IF((ISTD(I).LE.0).OR.(ISTD(I).GT.10))GOTO 200
      GOTO 201
  200 CALL PTEST
      WRITE(6,202)II
  202 FORMAT('0','CARD',I2,':',5X,'INVALID STAND NUMBER')
C  TEST FOR INVALIO SIZE CLASSIFICATION METHOD
  201 IF((ICTPS.EQ.0).AND.(IANALY(I,5).EQ.1))GOTO 103
      GOTO 102
  103 CALL PTEST
      WRITE(6,104)II
  104 FORMAT('0','CARD',I2,':',5X,'NO CUTPOINTS SPECIFIED')
  102 IF(ICTPS.LE.5)GOTO 106
      CALL PTEST
      WRITE(6,105)II,ICTPS,MANALY
  105 FORMAT('0','CARO',I2,':',5X,'TOO MANY CUTPTS SPECIFIEO',I5,I4)
  106 CONTINUE
C
C     TEST FOR VALIO OUTPUT SWITCHES
C
      DO 300 IL=1,5
         IF(ISWTCH(I,IL).LE.3) GO TO 300
         CALL PTEST
         WRITE(6,301)IL
 301     FORMAT(' ','OUTPUT SWITCH VALUE ',I1,' IS INVALID')
 300  CONTINUE
C  TEST FOR CORRECT ORDER OF CUTPOINTS
      ICTP1=ICTPS + 1
      CUTPTS(I,ICTP1)=10.E30
      IF(ICTPS.LE.1)GOTO 110
      CMIN=CUTPTS(I,1)
      DO 111 J=2,ICTPS
      IF(CMIN.GT.CUTPTS(I,J))GOTO 112
      CMIN=CUTPTS(I,J)
      GOTO 111
  112 CALL PTEST
      WRITE(6,113)II
  113 FORMAT('0','CARD',I2,':',5X,'CUTPOINTS NOT IN ASCENDING ORDER')
      GOTO 110
  111 CONTINUE
  110 CONTINUE
C  TEST FOR VALIO REGRESSION OPTION VALUE
      DO 49 J=1,10
      IF((IREGR(I,J).EQ.0).OR.(IREGR(I,J).EQ.1))GOTO 49
      CALL PTEST
      WRITE(6,51)II
   51 FORMAT('0CARD',I2,':',5X,'INVALIO REGRESSION INDICATOR SPECIFIED')
      GOTO 50
   49 CONTINUE
   50 CONTINUE
C  TEST FOR VALID VALUE OF ANALYSES OPTIONS
      DO 59 J=1,4
      IF((IANALY(I,J).EQ.0).OR.(IANALY(I,J).EQ.1))GOTO 59
      CALL PTEST
      WRITE(6,61)II
   61 FORMAT('0CARD',I2,':',5X,'INVALIO ANALYSIS INOICATOR SPECIFIEO')
      GOTO 60
   59 CONTINUE
   60 CONTINUE
C  TEST FOR VALIOITY OF SIZE CLASSIFICATION METHOD OPTION
      K=IANALY(I,5)
      IF((K.EQ.0).OR.(K.EQ.1).OR.(K.EQ.2))GOTO 80
      CALL PTEST
      WRITE(6,81)II
   81 FORMAT('0CARO',I2,':',5X,'INVALID SIZE CLASSIFICATION METHOD SPECI
     *FIED')
   80 CONTINUE
      IF(.NOT.FLAG)RETURN
      RETURN
      END
C*********************************************************************
      SUBROUTINE PTEST
C   THIS SUBROUTINE PRINTS HEADING FOR ERROR MESSAGES
C
C*********************************************************************
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      IF(.NOT.FLAG)RETURN
      FLAG=.FALSE.
      CALL PAGE
      WRITE(6,15)
   15 FORMAT(//  6X,'* * * * *  ERROR MESSAGES  * * * * *')
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE PSPECF
C  THIS SUBROUTINE PRINTS THE PROBLEM SPECIFICATIONS FOR EACH ANALYSIS
C
C*********************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
```

```
      LOGICAL FLAG
C
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      DIMENSION SUMX(10),SUMY(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTDT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      COMMON /MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *              DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
      LOGICAL*1 SIDE,L,R,BLEND(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      DATA L/'L'/,R/'R'/
      IFM=IFORM(NANALY)
      N=NANALY
C   PRINT THE TITLE
      CALL PAGE
      WRITE(6,89)REFNO
   89 FORMAT(/38X,'SURVEY REFERENCE NO.:',A8)
      WRITE(6,91)MANALY
   91 FORMAT(/  6X,'NUMBER OF ANALYSES REQUESTED:',I4)
      IF(IDATA.EQ.1)GOTO 93
      IF(IDATA.EQ.2)GOTO 95
      WRITE(6,92)
   92 FORMAT(/  6X,'DATA ENTRY MODE: CARDS')
      GOTO 99
   93 WRITE(6,94)
   94 FORMAT(/  6X,'DATA ENTRY MODE: DISK')
      RETURN
   95 WRITE(6,96)
   96 FORMAT(/  6X,'DATA ENTRY MODE: CARDS AND DISK')
   99 CONTINUE
      WRITE(6,100)
      WRITE(6,1)N,ISTO(N)
      WRITE(6,30)
   30 FORMAT(/16X,'SURVEY METHOD:')
   18 FORMAT(16X,'SAMPLING DESIGN:',4X,I3,' GRIDS X ',I3,' TRANSECTS')
      IF(IFORM(N).EQ.1)WRITE(6,31)
   31 FORMAT('+',35X,'INFECTION CENTERS MEASURED')
      IF(IFORM(N).EQ.2)WRITE(6,32)
   32 FORMAT('+',35X,'INFECTION CENTER AREA ESTIMATED')
      IF(IFORM(N).EQ.3)WRITE(6,33)
   33 FORMAT('+',35X,'INFECTION CENTERS ASSIGNED TO SIZE CLASSES')
      IF(IFORM(N).EQ.5)WRITE(6,34)
   34 FORMAT('+',35X,'INTERSECTION LENGTHS ONLY ASSIGNED')
      IF(IFORM(N).EQ.4)WRITE(6,35)
   35 FORMAT('+',35X,'INTERSECTION LENGTHS ONLY MEASURED')
      WRITE(6,18)NGRID(N),NTRNS(N)
      WRITE(6,29)
C   TEST FOR REGRESSION
      IF(IANALY(N,1).EQ.1)WRITE(6,2)
C   TEST FOR LINE INTERSECTION LENGTH METHOD OPTION
      IF(IANALY(N,2).EQ.1.OR.IFM.GT.1)WRITE(6,3)
C   TEST FOR AREA BY SIZE CLASS OPTION
      IF((IANALY(N,3).EQ.1.AND.IFM.EQ.1).OR.IFM.EQ.2)WRITE(6,4)
C   TEST FOR NUMBER PER SIZE CLASS OPTION
      IF((IANALY(N,4).EQ.1.AND.IFM.EQ.1).OR.IFM.EQ.3)WRITE(6,5)
      WRITE(6,21)
   21 FORMAT(//16X,'OUTPUTS REQUESTED:'/)
      IF(ISWTCH(N,1).EQ.1)WRITE(6,22)
   22 FORMAT(20X,'DESCRIPTORS AND SUMMARY')
      IF (ISWTCH(N,2).EQ.1.AND.IFM.EQ.1)WRITE(6,23)
   23 FORMAT(20X,'REGRESSION STATISTICS')
      IF(ISWTCH(N,3).EQ.1)WRITE(6,24)
   24 FORMAT(20X,'LIST OF CENTERS MEASURED')
      IF(ISWTCH(N,4).EQ.1)WRITE(6,25)
   25 FORMAT(20X,'COMPUTATION TABLES')
      IF(ISWTCH(N,5).EQ.1)WRITE(6,26)
   26 FORMAT(20X,'MAP  (INTERSECTION LENGTH ONLY)')
      IF(ISWTCH(N,5).EQ.2)WRITE(6,27)
   27 FORMAT(20X,'MAP  (PROJECTIONS AND MIDLENGTH)')
      IF(ISWTCH(N,5).EQ.3)WRITE(6,28)
   28 FORMAT(20X,'MAP (BOUNDARY POINTS)')
C   STANDS FOR INCLUSION IN REGRESSION ANALYSIS
      IF(IFM.EQ.1)WRITE(6,6)(I,I=1,10),(IREGR(N,I),I=1,10)
      WRITE(6,7)SUNITS
      ICTPS=NCTPS(N)
      IF(ICTPS.EQ.0)WRITE(6,8)
      ICTPS=NCTPS(N)
      IF(ICTPS.GT.0)WRITE(6,9)(CUTPTS(N,I),I=1,ICTPS)
      WRITE (6,13)MUNITS
      WRITE(6,19)LBASE(N)
   19 FORMAT(/16X,'BASELINE LENGTH:',T60,I10)
      IF(IANALY(N,5).EQ.2)WRITE(6,10)
      WRITE(6,11)DBTWN(N)
      WRITE(6,12)STRPW(N)
      IGLIST=0
      DO 50 IG=1,6
      IF(GSP(N,IG).EQ.0)GO TO 50
      IGLIST=IGLIST+1
   50 CONTINUE
      WRITE(6,17)BLEND(N),(GSP(N,IG),IG=1,IGLIST)
   17 FORMAT(/16X,'GRID START POINTS FROM ',A1,' END OF BASELINE',T60,
     16I10)
      WRITE(6,41)IBRNG(N),LTAPE
   41 FORMAT(/16X,'TRANSECT BEARING:', T60,I10,10X,'TAPE LENGTH',I5)
      IF(ISWTCH(N,5).EQ.1)WRITE(6,42)SCALMP,MUNITS
   42 FORMAT(/16X,'MAP SCALE:',T60,F6.1,A8,' TO 1 INCH')
    1 FORMAT(/' ANALYSIS',I3,':',3X,'STAND NUMBER:',I3)
   29 FORMAT(/  16X,'STATISTICS REQUESTED:'/)
    2 FORMAT(' ',20X,'REGRESSION ESTIMATE OF INDIVIDUAL CENTER SIZES')
    3 FORMAT(' ',20X,'INTERSECTION LENGTH AREA ESTIMATE')
    4 FORMAT(' ',20X,'TOTAL AREA ESTIMATES BY CENTER SIZE CLASS')
    5 FORMAT(' ',20X,'TOTAL NUMBER ESTIMATES BY SIZE CLASS')
    6 FORMAT(//  16X,'STANDS TO BE INCLUDED IN REGRESSION CALCULATIONS
     * OF CENTER SIZES '//20X,' STAND NUMBERS        :',10I3/
     *' ',20X,'REGRESSION INDICATOR:',10I3)
    7 FORMAT(//  16X,'CENTER SIZE CLASSIFICATION ( ',A8,')'/)
    8 FORMAT(' ',20X,'ONE SIZE CLASS ONLY')
    9 FORMAT(' ',20X,'DETERMINED FROM UPPER BOUNDARY LIMITS:',10F10.2)
   10 FORMAT(' ',20X,'SPECIFIED UNDER FORMAT CODE 3')
   11 FORMAT(/  16X,'GRID DISTANCE:',T60,F10.1)
      WRITE(6,15)SUNITS
      WRITE(6,14)SIZE(N)
      WRITE(6,16)FACTOR
   13 FORMAT(/16X,'SURVEY MEASUREMENT UNITS:',T62,A8)
   14 FORMAT(/16X,'STAND AREA:',T60,F10.1)
   15 FORMAT(/16X,'STAND AREA UNITS:',T62,A8)
   16 FORMAT(/16X,'AREA UNITS CONVERSION FACTOR:',T60,F10.1)
   12 FORMAT(/  16X,'STRIP WIDTH :',T60,F10.1)
  100 FORMAT(/  35X,'* * * * * *  ANALYSES REQUESTED  * * * * * *')
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE RCARD
      LOGICAL FLAG
C
C    THIS SUBROUTINE READS FROM UNIT 5 CALCULATES AREAS OF INDIVIDUAL
C    CENTERS, ACCUMULATES SUMS FOR REGRESSION AND CREATES A SORTED
C    DISK FILE.
C       METHOD=0: NO CENTER MEASUREMENT
C       METHOD=1: MID-DIAMETER AND PROJECTION MEASURED
C       METHOD=2: RADIAL MEASUREMENT
C       METHOD=3: LINEAR MEASUREMENT
C
C*********************************************************************
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
C
      DIMENSION KREGR(10),ISRTD(2000)
      DIMENSION SUMX(10),SUMY(10),SUMXY(10),SUMXX(10),SUMYY(10),NSAMP(10
     *),INDEX(10)
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTO(10)
      REAL*8 XSRT(2000)
      COMMON/DHB/SUMXY,SUMXX,SUMYY,NSAMP,INDEX
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1,IVERS
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTO,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 SIDE,L,R,BLEND(10)
      COMMON/MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *   DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *             DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
      DATA PI/3.141529/
      FLAG=.TRUE.
C
C       ASIGN SURVEY METHOD
C
C   INITIALIZE VARIABLES
      DO 51 I=1,10
      INDEX(I)=0
      SUMX(I)=0.
      SUMY(I)=0.
      SUMXY(I)=0.
      SUMXX(I)=0.
      SUMYY(I)=0.
      KREGR(I)=0
C   DETERMINE WHICH STANDS ARE ELIGIBLE FOR INCLUSION IN THE
C   CALCULATIONS OF REGRESSION COEFFICIENTS ACCORDING TO FORMAT TYPE
      IF(IFORM(I).EQ.1)KREGR(I)=1
   51 NSAMP(I)=0
      IREAD =0
      IREC=1
C   INPUT ROOT ROT CENTRE MEASUREMENTS FROM DISC FILE OR CARDS
      IF(IDATA.NE.2)GOTO 111
C   COPY THE OLD DISK FILE TO THE SCRATCH FILE
      CALL COPYF(IREC)
C   READ NEXT RECORD
C       NSTD     = STAND NO.
C       NGRD     = GRID NO.
C       NLIN     = LINE NO.
C       NCEN     = CENTRE NO.
C       PT1      = START OF INTERSECTION
C       PT2      = END OF INTERSECTION
C       METH     = MEASUREMENT METHOD
C       NTAPE    = ACCUMULATED NO. OF TAPE LENGTHS
C       YDST12   = DISTANCE TO L. PROJECTION TIE-POINT
C       YDST22   = DISTANDE TO R. PROJECTION TIE-POINT
C       DTRDST   = DISTANCE TO MID-LENGTH TIE-POINT
C       YDIST3   = DISTANCE TO MEASUREMENT POINT TIE-POINT
C       XDIST3   = OFFSET TO MEASUREMENT POINT
C       CNTANG   = CENTER LINE ANGLE
C       PRJ1     = LEFT PROJECTION
C       PRJ2     = RIGHT PROJECTION
C       DIAM     = MID LENGTH
C       DIST     = MEASUREMENT INTERVAL
C       NLINES   = NO. MEASUREMENTS
C       DATA     = MEASUREMENTS
C       NSTD1 - NCEN1 = ID OF CENTRE IF PREVIOUSLY ENCOUNTERED
C
  111 DO 53 I=1,5
      DO 53 J=1,10
   53 DATA(I,J)=0.
      NSTD1=0
      NGRD1=0
      NLIN1=0
      NCEN1=0
C       MORE THAN ONE DATA SET PER CENTRE NOT IMPLEMENTED
    2 IF(IVERS.EQ.1)
```

```
     1READ(5,900,END=600)NSTD,NGRD,NLIN,NCEN,PT1,PT2,METH,PRJ1,PRJ2,DIAM
     *,DIST,NLINES,(DATA(1,J),J=1,10),NSTD1,NGRD1,NLIN1,NCEN1
  900 FORMAT(4I2,2F5.0,I2,3F3.0,F5.0,I2,10I3,4I2)
      IF(IVERS.EQ.2)
     1READ(5,901,END=600)NSTD,NGRD,NLIN,NCEN,NTAPE,PT1,PT2,METH,YDST12,
     1YDST22,DTRDST,YDIST3,XDIST3,CNTANG,SIDE,PRJ1,PRJ2,DSTFOR,DSTBCK,
     2NLINES,DIST,(DATA(1,J),J=1,10)
  901 FORMAT(3I1,I2,I3,2F3.0,I2,4F3.0,F2.0,F3.0,A1,4F3.0,I2,F2.0,10I3)
      IF(METH.NE.8.AND.METH.NE.9)GO TO 222
      IF(IVERS.EQ.1)GO TO 222
      NSTD1 = DATA(1,7)
      NGRD1 = DATA(1,8)
      NLIN1 = DATA(1,9)
      NCEN1 = DATA(1,10)
 222  IF(METH.NE.3)GO TO 889
      IF(IVERS.EQ.1)GO TO 889
      DO 888 LL=1,NLINES
         LLL = LL * 2 - 1
         DATA(1,LL) = DATA(1,LLL) + DATA(1,LLL+1)
 888  CONTINUE
C
C       CALCULATE ACCUMULATED DISTANCE
C
      TDIST=NTAPE*LTAPE
      PT1=PT1+TDIST
      PT2=PT2+TDIST
      YDST12=YDST12+TDIST
      YDST22=YDST22+TDIST
      YDIST3=YDIST3+TDIST
      DTRDST=DTRDST+TDIST
 889  IF(NCEN.EQ.0)GO TO 111
      IF(IVERS.EQ.2)DIAM = DSTFOR + DSTBCK
      PROJ=PRJ1+PRJ2
C     CACULATE INTERSECTION LENGTH
      XLEN=ABS(PT1-PT2)
      NGRIDS=1
      IREAD = IREAD + 1
C
C     CHECK IF STAND DATA EXISTS ON DISK
C
      IF(INDEX(NSTD).EQ.0)GOTO 52
      CALL EMSG(FLAG,NSTD,NGRD,NLIN,7,IREAD)
      GOTO 111
C  TEST VALIDITY OF THE IDENTIFICATION FIELD VALUES
   52 IF(NSTD.GT.10)GOTO 70
      IF((NSTD.LT.1).OR.(NGRD.LT.1).OR.(NLIN.LT.1))GOTO 70
      GOTO 80
   70 CALL EMSG(FLAG,NSTD,NGRD,NLIN,1,IREAD)
      GOTO 111
C  SET NUMBER OF SIZE CLASS TO DEFAULT VALUE.
   80 NCLASS=1
C  TEST SELECTED FORMAT AND BRANCH FOR APPROPRIATE PROCESSING
      GOTO(10,20,30,40,50),IFORM(NSTD)
C  INVALID FORMAT NUMBER
      CALL EMSG(FLAG,NSTD,NGRD,NLIN,5,IREAD)
      GOTO 111
C  DATA FORMAT SELECTION 1
   10 AREA=0.
C  CALCULATE AREA OF CENTER BY REGRESSION
      IF(METH.EQ.1)GOTO 200
C                            BY RADIAL METHOD
      IF(METH.EQ.2)GOTO 300
C                            BY LINEAR METHOD
      IF(METH.NE.3)GOTO 400
C  LINEAR METHOD.
      SUM=0.
      DO 100 I=1,NGRIDS
      DO 100 J=1,NLINES
      SUM=SUM + DATA(I,J)
  100 CONTINUE
      AREA=DIST*SUM/NGRIDS
C  TEST FOR ZERO AREA.
      IF(AREA.LE.0.)CALL EMSG(FLAG,NSTD,NGRD,NLIN,2,IREAD)
      GOTO 200
C  RADIAL METHOD
  300 SUM=0.
      DO 110 I=1,NGRIDS
      DO 110 J=1,NLINES
      SUM=SUM+DATA(1,J)**2
  110 CONTINUE
      AREA=PI*SUM/(NGRIDS*NLINES)
C  TEST FOR ZERO AREA
      IF(AREA.LE.0.)CALL EMSG(FLAG,NSTD,NGRD,NLIN,3,IREAD)
      GOTO 200
  400 IF(METH.EQ.0)GOTO 202
      GO TO 145
C  INPUTS FOR FORMAT OPTION 2
   20 AREA=DSTBCK+1000*DSTFOR
      DIST=0.
      GO TO 145
C  INPUTS FOR FORMAT OPTION 3
   30 AREA=0.
      NCLASS=NLINES
      GO TO 145
C  INPUTS FOR FORMAT OPTION 4
   40 AREA=0.
C  INPUTS FOR FORMAT OPTION 5
   50 AREA=0.
      GO TO 145
C       CHECK MULTIPLE ENCOUNTERS ON SAME OR DIFFERENT GRID
C
145   IF(METH.EQ.8.OR.METH.EQ.9)GO TO 150
      IF(METH.NE.1)CALL EMSG(FLAG,NSTD,NGRD,NLIN,4,IREAD)
C       METHOD VALUE INVALID,SUBSTITUTE 1
      METH=1
      GO TO 200
C  COPY MEASUREMENTS OF PREVIOUSLY ENCOUNTERED CENTRE
  150 DO 199 I=1,NOREC
      READ(1'I)NSTD2,NGRD2,NLIN2,NCEN2,XLEN2,METH2,PROJ2,AREA2,
     1DIAM2,NCLASS2
      IF(NSTD1.NE.NSTD2.OR.NGRD1.NE.NGRD2.OR.NLIN1.NE.NLIN2.OR.
     1NCEN1.NE.NCEN2)GO TO 199
      PROJ=PROJ2
      AREA=AREA2
      DIAM=DIAM2
      NCLASS=NCLASS2
  199 CONTINUE
      GO TO 202
C  TEST FOR INCLUSION OF CENTRE IN REGRESSION CALCULATION
  200 IF(KREGR(NSTD).EQ.1)GOTO 201
      GOTO 202
  201 IF((PROJ*DIAM).LE.0.)GOTO 203
      IF(AREA.LE.0.)GOTO 202
C  ACCUMULATE SUMS, SUMSQUARES, AND CROSS-PRODUCTS FOR REGRESSION
C  ANALYSIS
      CALL ACSUM1(NSTD)
D      WRITE(6,1000)SUMX(NSTD),SUMY(NSTD),SUMXX(NSTD),SUMYY(NSTD),
D    1NSAMP(NSTD)
D1000 FORMAT(T10,4F10.0,I5)
      GOTO 202
  203 KREGR(NSTD)=0
      CALL EMSG(FLAG,NSTD,NGRD,NLIN,6,IREAD)
      GOTO 111
  202 IF(.NOT.FLAG)GOTO 111
C  WRITE THE RECORD ON SCRATCH DISK AND INCREMENT RECORD COUNTER
      WRITE(1'IREC)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
     1NSTD1,NGRD1,NLIN1,NCEN1
D     WRITE(6,2000)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
D    1NSTD1,NGRD1,NLIN1,NCEN1
 2000 FORMAT(T10,'RCARD 2000',4I5,F5.0,I5,3F10.0,5I5)
      NOREC=IREC
      IREC=IREC+1
      GOTO 111
C  TEST FLAG BEFORE CONTINUING
  600 IF(.NOT.FLAG)RETURN
      I10=10*10
      NREC=IREC-1
C  READ SCRATCH FILE SORT FIELD AND INITIALIZE INDEXING VARIABLE
      DO 650 I=1,NREC
      READ(1'I)NSTD,NGRD,NLIN,NCEN
      XSTD=NSTD
      XGRD=NGRD
      XLIN=NLIN
      XCEN=NCEN
C      CREATE VECTOR FROM RECORD I.D.
      XSRT(I)=((XSTD*I10+XGRD)*I10+XLIN)*I10+XCEN
      ISRTD(I)=I
  650 CONTINUE
      DO 651 I=1,10
  651 INDEX(I)=0
      NTEST=0
C  SORT THE RECORDS BY STAND, GRID, LINE, AND CENTRE NO.
      CALL SHSORT(XSRT,ISRTD,NREC)
      DO 652 I=1,NREC
      READ(1'ISRTD(I))NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLAS
     1S,NSTD1,NGRD1,NLIN1,NCEN1
D      WRITE(6,3000)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
D    1NSTD1,NGRD1,NLIN1,NCEN1
 3000 FORMAT(T10,'RCARD 3000',4I5,F5.0,I5,3F10.0,5I5)
      K=I+1
      WRITE(2'K)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
     1NSTD1,NGRD1,NLIN1,NCEN1
D     WRITE(6,4000)NREC,K,IREC,NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,
D    1AREA,DIAM,NCLASS
 4000 FORMAT(T10,'RCARD 4000',7I5,F10.0,I5,3F10.0,I5)
      IF(NSTD.LE.NTEST)GOTO 652
      NTEST=NSTD
      INDEX(NSTD)=K
  652 CONTINUE
      IREC=IREC+1
      NREC=K+10
C
C  WRITE THE FILE INDEX - INDEX(NSTD) RECORDS THE NUMBER OF THE FIRST
C  OCCURRANCE OF STAND NSTD.
C
C  NREC IS THE TOTAL NUMBER OF RECORDS IN THE SORTED FILE
C
      WRITE(2'1)NREC,(INDEX(I),I=1,10)
      DO 601 I=1,10
      K=100+I
      WRITE(2'IREC)K,SUMX(I),SUMY(I),SUMXY(I),SUMXX(I),SUMYY(I),NSAMP(I)
D     WRITE(6,5000)NREC,IREC,(INDEX(L),L=1,10),K,SUMX(I)
 5000     FORMAT(T10,'RCARD 5000',13I5,F10.0)
      IREC=IREC+1
  601 CONTINUE
      REWIND 5
      DO 1111 I=1,MANALY+3
         READ(5,1112)ISKIP
 1111 CONTINUE
 1112 FORMAT(A2)
      RETURN
      END
C
C
C*******************************************************************
      SUBROUTINE EMSG(FLAG,NSTD,NGRD,NLIN,NERR,IREAD)
C  THIS SUBROUTINE PRINTS ERROR MESSAGES FOR CARD INPUT FAULTS
C*******************************************************************
      LOGICAL FLAG
      DATA DASH/4H----/
      DATA IERR/0/
      FLAG=.FALSE.
      IERR=IERR+1
      IF((IERR.NE.1).AND.(IERR.LT.29))GOTO 10
      CALL PAGE
      IERR=1
      WRITE(6,11)(DASH,I=1,8)
   11 FORMAT(// ' ERROR MESSAGES FOR SUBROUTINE RCARD'/' ',8A4,'---'/'0'
     *,'RECD. NO.  ID. NO.',20X,'ERROR MESSAGE')
   10 CONTINUE
C  SELECT APPROPRIATE ERROR MESSAGE
      GOTO(21,22,23,24,25,26,27)NERR
   21 WRITE(6,12)IREAD,NSTD,NGRD,NLIN
```

```
   12 FORMAT('0',I6,I7,2I2,5X,'ERROR IN IDENTIFICATION FIELD.')
      GO TO 30
   22 WRITE(6,13)IREAD,NSTD,NGRD,NLIN
   13 FORMAT('0',I6,I7,2I2,5X,'LINEAR METHOD YIELDS ZERO AREA.')
      GOTO 30
   23 WRITE(6,14)IREAD,NSTD,NGRD,NLIN
   14 FORMAT('0',I6,I7,2I2,5X,'RADIAL METHOD YIELDS ZERO AREA.')
      GOTO 30
   24 WRITE(6,15)IREAD,NSTD,NGRD,NLIN
   15 FORMAT('0',I6,I7,2I2,5X,'ESTIMATION METHOD NO. IS INVALID.  USING *
     *DEFAULT VALUE 1.')
      GOTO 30
   25 WRITE(6,16)IREAD,NSTD,NGRD,NLIN
   16 FORMAT('0',I6,I7,2I2,5X,'INVALID FORMAT NUMBER FOR THIS STAND.')
      GOTO 30
   26 WRITE(6,17)IREAD,NSTD,NGRD,NLIN
   17 FORMAT('0',I6,I7,2I2,5X,'INVALID ENTRY FOR PROJECTION OR MID-DIAME
     *TER.')
      GOTO 30
   27 WRITE(6,18)IREAD,NSTD,NGRD,NLIN
   18 FORMAT('0',I6,I7,2I2,5X,'THIS STAND ALREADY EXISTS ON DISK')
   30 RETURN
      END
C
C******************************************************************
      SUBROUTINE ACSUM1(I)
C  THIS SUBROUTINE ACCUMULATES SUMS, SUM SQUARES, AND CROSS PRODUCTS
C  OF CENTRE AREAS AS MEASURED BY APPROXIMATE OR ACCURATE METHODS FOR
C  USE IN THE CALCULATION OF REGRESSION PARAMETERS
C******************************************************************
      DIMENSION SUMX(10),SUMY(10),SUMXY(10),SUMXX(10),SUMYY(10),NSAMP(10
     *),INDEX(10)
      COMMON/DHB/SUMXY,SUMXX,SUMYY,NSAMP,INDEX
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/CK/NGRO,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
      SUMX(I)=SUMX(I)+PROJ*DIAM
      SUMY(I)=SUMY(I)+AREA
      SUMXY(I)=SUMXY(I)+AREA*PROJ*DIAM
      SUMXX(I)=SUMXX(I)+(PROJ*DIAM)**2
      SUMYY(I)=SUMYY(I)+AREA*AREA
      NSAMP(I)=NSAMP(I)+1
      RETURN
      END
C
C******************************************************************
      SUBROUTINE SHSORT(A, B, N)
C
C     SORTS 'N' RECORDS BY VECTOR FOR STAND, GRID, LINE, AND CENTRE
C     NO. (A), AND RETURNS THE SORTED RECORD POSITION (B)
C     SHELL SORT
C******************************************************************
C     THE 'N' ELEMENTS OF REAL ARRAY 'A' ARE SORTED IN ASCENDING ORDER
C     AND THE ELEMENTS OF INTEGER ARRAY 'B' ARE MOVED TO CORRESPONDING
C     POSITIONS
C
      INTEGER B(N)
      REAL*8 A(N),TEMP
C
      M=N
    4 M=M/2
      IF(M) 10,10,5
    5 K=N-M
      DO 9 J=1,K
      I=J+M
    6 I=I-M
      IF(I) 9,9,7
    7 L=I+M
      IF(A(L).GE.A(I)) GO TO 9
      TEMP=A(I)
      A(I)=A(L)
      A(L)=TEMP
      ITEMP=B(I)
      B(I)=B(L)
      B(L)=ITEMP
      GO TO 6
    9 CONTINUE
      GO TO 4
   10 RETURN
      END
C
C******************************************************************
      SUBROUTINE COPYF(IREC)
C  THIS SUBROUTINE COPIES THE OLD FILE TO THE SCRATCH FILE
C******************************************************************
      DIMENSION SUMX(10),SUMY(10),SUMXY(10),SUMXX(10),SUMYY(10),NSAMP(10
     *),INDEX(10)
      COMMON/DHB/SUMXY,SUMXX,SUMYY,NSAMP,INDEX
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/CK/NGRO,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
C  FOR EACH ROOT ROT CENTRE READ IN STAND NO., GRID NO., LINE NO.,
C  CENTRE NO., INTERSECTION LENGTH, MEASUREMENT METHOD, PROJECTION
C  LENGTH, CALCULATED AREA, MID-LENGTH, SIZE CLASS, AND ID (IF ANY)
C  OF CENTRE WHEN PREVIOUSLY ENCOUNTERED
      READ(2'1)NREC,(INDEX(I),I=1,10)
      NSTOP=NREC-10
      DO 10 I=2,NSTOP
      READ(2'I)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
     1NSTD1,NGRD1,NLIN1,NCEN1
      K=I-1
      WRITE(1'K)NSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
     1NSTD1,NGRD1,NLIN1,NCEN1
   10 CONTINUE
C  SET WRITE COUNTER FOR ADDING THE NEXT RECORD
      IREC=K+1
      DO 20 I=1,10
      K=NSTOP+I
C  READ IN SUMS, SUM SQUARES, AND CROSS-PRODUCTS FOR CENTRE AREA
C  REGRESSION ESTIMATES
      READ(2'K)NDUM,SUMX(I),SUMY(I),SUMXY(I),SUMXX(I),SUMYY(I),NSAMP(I)
   20 CONTINUE
      RETURN
      END
C
C******************************************************************
      SUBROUTINE REGRN(IERR)
C  THIS SUBROUTINE CONTROLS THE REGRESSION CALCULATIONS
C
C******************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),OBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      DIMENSION SUMX(10),SUMY(10)
      DIMENSION KREGR(10),IDEX1(10),IDEX2(10)
      COMMON/SLT/TOTX,TOTY,TOTXY,TOTXX,TOTYY,NTOT
      COMMON/ALL/IANALY,IREGR,STRPW,OBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      DATA NREGRN/0/
C  SET ERROR COUNT TO 0
      IERR=0
C  INITIALIZE KREGR FROM DISK
      READ(2'1)NREC
      DO 29 I=1,10
      K=I+NREC-10
      KREGR(I)=0
      READ(2'K)NDUM,TEST
C  SET INDICATOR FOR INCLUSION OF STAND IN REGRESSIONAL ANALYSIS
   29 IF(TEST.GT.0.)KREGR(I)=1
C  TEST FOR FIRST STAND TO BE INCLUDED IN REGRESSION
      IF((NANALY.EQ.1).OR.(NREGRN.EQ.0))GOTO 333
C  TEST FOR ADDITIONAL STANDS TO BE INCLUDED IN REGRESSION
      DO 210 J=1,10
  210 IF((IREGR(NANALY,J)).NE.(IREGR(NREGRN,J)))GOTO 320
      CALL ANOVA
      RETURN
C  TEST FOR ELIGIBILITY OF STANDS
  320 DO 310 J=1,10
  310 IF(KREGR(J).LT.IREGR(NANALY,J))GOTO 700
C  BEGIN NEW REGRESSION
  333 NREGRN=NANALY
C  INITIALIZE SUMS, SUMSQUARES, AND CROSS-PRODUCTSUMS TO ZERO
      TOTX=0.
      TOTY=0.
      TOTXY=0.
      TOTXX=0.
      TOTYY=0.
      NTOT=0
C  ACCUMULATE SUMS
      DO 410 I=1,10
      K=NREC-10+I
      IF(IREGR(NANALY,I).EQ.0)GOTO 410
      READ(2'K)NDUM,SX,SY,SXY,SXX,SYY,NS
      TOTX=TOTX+SX
      TOTY=TOTY+SY
      TOTXY=TOTXY+SXY
      TOTXX=TOTXX+SXX
      TOTYY=TOTYY+SYY
      NTOT=NTOT+NS
D      WRITE(6,1000)NDUM,SX,SY,SXY,SXX,SYY,NS
D1000 FORMAT(T10,'ACSUM 1000',I5,5F10.0,I5)
  410 CONTINUE
      CALL ANOVA
      RETURN
C  PRINT ERROR MESSAGE AND SET ERROR COUNT
  700 NDEX1=0
      NDEX2=0
C  OBTAIN LIST OF STAND NUMBERS FOR PRINTING
      DO 710 I=1,10
      IF(KREGR(I).EQ.0)GOTO 711
      NDEX1=NDEX1+1
      IDEX1(NDEX1)=I
  711 IF(IREGR(NANALY,I).EQ.0)GOTO 710
      NDEX2=NDEX2+1
      IDEX2(NDEX2)=I
  710 CONTINUE
      CALL PAGE
      WRITE(6,707)NANALY,(IDEX1(I),I=1,NDEX1),(IDEX2(K),K=1,NDEX2)
      IERR=1
  707 FORMAT(// ' ANALY',I2,':',5X,'* * * * *  ERROR MESSAGE  * * * * *'
     */'0',10X,'INVALID REQUEST FOR INCLUSION OF STANDS IN REGRESSION'
     */'0',20X,'ELIGIBLE STANDS ARE:',10I4
     */'0',20X,'STANDS REQUESTED   :',10I4
     */'0',10X,'**********  SKIPPING TO NEXT ANALYSIS  **********')
      RETURN
      END
C
C******************************************************************
      SUBROUTINE ANOVA
C  THIS SUBROUTINE PRINTS THE ANOVA TABLE FOR THE REGRESSION FORCED
C  THROUGH THE ORIGIN AND TESTS FOR THE INTERCEPT
C
C******************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),OBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10),TVAL(30),FVAL(30),A(500,2),
     *  INDEX(10),LIST(10)
      DIMENSION SUMX(10),SUMY(10)
      REAL MSE
      COMMON/SLT/TOTX,TOTY,TOTXY,TOTXX,TOTYY,NTOT
      COMMON/ALL/IANALY,IREGR,STRPW,OBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      DATA STARB/4H* * /
      DATA FVAL/161.,18.51,10.13,7.71,6.61,5.99,5.59,5.32,5.12,4.96,
```

```
     14.84,4.75,4.67,4.60,4.54,4.49,4.45,4.41,4.38,4.35,4.32,4.30,
     24.28,4.26,4.24,4.22,4.21,4.20,4.18,4.17/
      DATA TVAL/12.71,4.30,3.18,2.78,2.57,2.45,2.36,2.31,2.26,2.23,
     12.20,2.18,2.16,2.14,2.13,2.12,2.11,2.10,2.09,2.09,2.08,2.07,
     22.07,2.06,2.06,2.06,2.05,2.05,2.04,2.04/
C
C       CALCULATE SUM OF SQUARES AND PRODUCTS
C
        TOTXY=TOTXY-TOTX*TOTY/NTOT
        TOTXSQ=TOTXX
        TOTXX=TOTXX-TOTX*TOTX/NTOT
        TOTYY=TOTYY-TOTY*TOTY/NTOT
      NTOT1=NTOT-1
      CEPT=0.
      B=TOTXY/TOTXX
      SSB=B*TOTXY
      SSE=TOTYY-SSB
      MSE=SSE/NTOT1
      F=SSB/MSE
      SEB=SQRT(MSE/TOTXX)
      NOF=NTOT1
      IF(NTOT1.GT.30)NOF=30
      CONLIM=TVAL(NOF)*SEB
C
C IDENTIFY STANDS IN REGRESSION CALCULATION
C
      NLIST=0
      NL=0
      DO 10 I = 1,10
      IF(IREGR(NANALY,I).NE.1) GO TO 10
      NLIST=NLIST+1
      NL=NL+1
      LIST(NL)=ISTD(I)
 10   CONTINUE
      IF(ISWTCH(NANALY,2).NE.1)GO TO 11
C  PRINT THE ANOVA TABLE
      CALL PAGE
      WRITE(6,901)NANALY,(STARB,I=1,8),(LIST(NL),NL=1,NLIST)
      WRITE(6,911)SSB,1,SSB,F,SSE,NTOT1,MSE,NTOT1,TOTYY,NTOT,B,SEB,
     1CONLIM
      IF(F.GE.FVAL(NDF))WRITE(6,800)
  800 FORMAT(11X,'***** REGRESSION IS SIGNIFICANT'//)
      IF(F.LT.FVAL(NDF))WRITE(6,801)
C
  801 FORMAT(11X,'***** REGRESSION IS NOT SIGNIFICANT DUE TO ',
     1'INSUFFICIENT CENTERS MEASURED'//)
C       CALCULATE STATS FOR TEST OF INTERCEPT (SEE DRAPER AND SMITH P. 21)
   11 NTOT2=NTOT-2
      NOF2=NTOT2
      CEPT=TOTY/NTOT-(B*TOTX/NTOT)
      VARB0=MSE*TOTXSQ/NTOT*TOTXX
      SE=SQRT(VARB0)
      T=CEPT/SE
      IF(CEPT.LE.0)CEPT=0.
      IF(NTOT2.GT.30)NOF2=30
      IF(ISWTCH(NANALY,2).NE.1)GO TO 81
      WRITE(6,902)CEPT,VARB0,NOF2,SE,T,B
  901 FORMAT(// ' ANALYSIS',I2,'1',10X,4A4,'  CENTER SIZE',
     1' REGRESSION ANALYSIS  ',4A4//
     2T10,'* * * THE REGRESSION EQUATION IS USED TO ESTIMATE AREA ',
     3'OF CENTERS FROM TOTAL PROJECTION AND MID-LENGTH * * *'//
     4T16,'REGRESSION BASED ON STAND(S)',<NLIST>I5//
     *'0',10X,'INDEPENDENT VARIABLE:  X = TOTAL PROJECTION * MID-',
     *'LENGTH'/'0',10X,'DEPENDENT VARIABLE  :  Y = AREA'//
     *'0',20X,'ANOVA TABLE (ZERO INTERCEPT MODEL:  E(Y) = B * X)'/
     *'0',10X,'SOURCE',17X,'SS',18X,'DF',18X,'MS',19X,'F'/)
  911 FORMAT('0',10X,'SLOPE',F23.3,I17,3X,2F20.3,T102,
     1'(REFER TO F DISTRIBUTION'/
     *'0',10X,'RESIDUAL',F20.3,I17,3X,F20.3,
     3T102,'TABLE WITH',I4,/
     *'0',10X,'TOTAL',F23.3,I17,3X,T102,'DEGREES OF FREEDOM)'//
     *//  11X,'SLOPE:  B=',F10.5,20X,'STANDARD ERROR OF B =',F12.3//
     411X,'CONFIDENCE LIMITS OF ESTIMATE',10X,'+ OR -',F10.6//)
  902 FORMAT(//  21X,'T TEST FOR INCLUSION OF THE INTERCEPT'//
     *'0',10X,'INTERCEPT',10X,'VARIANCE',15X,'DF',15X,'SE',10X,'T'/
     1F20.3,F20.1,12X,I3,F20.1,6X,F4.2///
     *'0',10X,'PREDICTION EQUATION:  Y = (',F10.5,') * X')
      IF(T.GT.TVAL(NOF2).AND.CEPT.GT.0.)WRITE(6,903)CEPT
  903 FORMAT('+',T55,' + ',F10.2)
      WRITE(6,904)
  904 FORMAT(//10X,'IF INTERCEPT IS SIGNIFICANT, IT IS INCLUDED IN THE
     1 REGRESSION EQUATION'//)
C
C     PRODUCE SCATTER PLOT FOR LINEAR REGRESSION
C
C     READ DATA FOR SCATTER PLOT
      READ(2'1)NREC,(INDEX(I),I=1,10)
      NSTD = ISTD(NANALY)
      IREC = INDEX(NSTD)
      MAXX = 0
      MAXY = 0
      IMXIND = 0
 1    READ(2'IREC)KSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
     1NSTD1,NGRD1,NLIN1,NCEN1
      IREC = IREC + 1
      IF(AREA.EQ.0.OR.KSTD.NE.NSTD) GO TO 2
C     PREPARE INPUT PARAMTERS FOR SCPLOT
      XVALUE = PROJ * DIAM
      IF(MAXX.LT.XVALUE) MAXX = XVALUE
      IF(MAXY.LT.AREA) MAXY = AREA
      IMXIND = IMXIND + 1
      A(IMXIND,1) = XVALUE
      A(IMXIND,2) = AREA
 2    IF(KSTD.EQ.NSTD) GO TO 1
 3    CALL SCPLOT(A,MAXX,MAXY,IMXIND,MUNITS,NSTD)
 81   RETURN
      END
C
C**********************************************************************
      SUBROUTINE SCPLOT(A,MAXX,MAXY,IMXIND,MUNITS,NSTD)
C
C     THIS SUBROUTINE TAKES AN ARRAY OF X AND Y VALUES AND PRODUCES
C     A SCATTER-PLOT FROM IT
C**********************************************************************
      DIMENSION A(500,2)
      LOGICAL*1 VERT(50),PRINTL(100)
      REAL*8 MUNITS
      DATA VERT / 18*' ','A','R','E','A',' ','E','S','T','I','M','A',
     1            'T','E',19*' ' /
      DATA IBL / ' ' /
      DATA IASK / '*' /
C
C     DETERMINE SUITABLE SCALES
C
      YSCALE = MAXY / 50.
      XSCALE = MAXX / 100.
C
C     ORDER ARRAY IN DESCENDING ORDER
C
      IMX = IMXIND - 1
      DO 1 I=1,IMX
         MAX = A(I,2)
         ISAV = I
         K = I + 1
         DO 2 J=K,IMXIND
            IF(A(J,2).LT.MAX) GO TO 2
            MAX = A(J,2)
            ISAV = J
 2       CONTINUE
         TEMP1 = A(I,1)
         TEMP2 = A(I,2)
         A(I,1) = A(ISAV,1)
         A(I,2) = A(ISAV,2)
         A(ISAV,1) = TEMP1
         A(ISAV,2) = TEMP2
 1    CONTINUE
C
C     PRINT HEADING STARTING ON NEW PAGE
C
      CALL PAGE
      WRITE(6,10) NSTD,XSCALE,MUNITS,YSCALE,MUNITS
 10   FORMAT('0','SCATTER PLOT FOR LINEAR REGRESSION',
     1T50,'STAND NO.',I4,' ONLY'//,' SCALE:',
     2      '  X AXIS;  1 UNIT =',F6.1,' SQ.',A8,/,9X,'Y AXIS;',
     3      '  1 UNIT =',F6.1,' SQ.'A8,/)
C
C     PRINT SCATTER PLOT
C
      IARR = 1
      INCRNT = 51
      DO 4 I=1,50
         INCRNT = INCRNT - 1
C        BLANK OUT PRINT LINE
         DO 5 J=1,100
            PRINTL(J) = IBL
 5       CONTINUE
C        FILL IN PRINT LINE WITH APPROPRIATE DATA
         RNGY = INCRNT * YSCALE - (YSCALE/2)
 7       IF(A(IARR,2).LT.RNGY.OR.IARR.GT.IMXIND) GO TO 6
         IXCOR = A(IARR,1) / XSCALE
         IF(IXCOR.LT.1)IXCOR=1
         PRINTL(IXCOR) = IASK
         IARR = IARR + 1
         GO TO 7
 6       INT = INCRNT / 10
C        WRITE PRINT LINE
         IF(INCRNT.EQ.(INT*10)) GO TO 8
         WRITE(6,11) VERT(I),(PRINTL(M),M=1,100)
 11      FORMAT(' ',A1,6X,'.',100A1)
         GO TO 4
 8        WRITE(6,12) VERT(I),INCRNT,(PRINTL(M),M=1,100)
 12       FORMAT(' ',A1,5X,I2,100A1)
 4    CONTINUE
C
C     PRINT BOTTOM LINES
C
      WRITE(6,13)
 13   FORMAT(' ',7X,'.',10(9('.'),','),/,8X,'0',8X,'10',8X,'20',
     1      8X,'30',8X,'40',8X,'50',8X,'60',8X,'70',8X,'80',8X,
     2      '90',7X,'100',///,30X,'TOTAL PROJECTION X MID-LENGTH')
      RETURN
      END
C
C**********************************************************************
      SUBROUTINE ACSUM(IERR)
C
C  THIS SUBROUTINE READS THE SORTED DISK FILE RECORDS FOR EACH CENTER,
C  ACCUMULATES SUMS OF AREAS AND NUMBERS, AND PRINTS A LISTING OF THE
C  DATA BY CENTRE
C
C**********************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      DIMENSION XMAT(6,20),TSMAT(10,6,20),SMAT(10,6,20)
      DIMENSION SUMX(10),SUMY(10)
      DIMENSION INDEX(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBNO(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
      COMMON/DHB/SUMXY,SUMXX,SUMYY,NSAMP,INDEX
      COMMON/FKT/XMAT,TSMAT,SMAT,D
      IERR=0
C
```

```
C       ASSIGN SURVEY METHOD
C
        IFM=IFORM(NANALY)
C   READ INDEX TO DISK FILE
        READ(2'1)NREC,(INDEX(I),I=1,10)
C   INITIALIZE VARIABLES
        DO 10 I=1,6
        DO 10 J=1,20
     10 XMAT(I,J)=0.
        DO 20 I=1,10
        DO 20 J=1,6
        DO 20 K=1,20
        TSMAT(I,J,K)=0.
     20 SMAT(I,J,K)=0.
        DO 25 I = 1,10
        DO 25 J = 1,2
        SUMTOT(I,J) = 0.
     25 SUMNO(I,J) = 0.
        SUMARE(1) = 0.
        SUMARE(2) = 0.
        NSTD=ISTD(NANALY)
        IREC=INDEX(NSTD)
        MGRD=NGRID(NSTD)
        MLIN=NTRNS(NSTD)
        MCLASS=NCTPS(NANALY)
        LCOUNT=0
        SW=STRPW(NANALY)
        D=DBTWN(NANALY)
C   READ RECORD
     30 READ(2'IREC)KSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
       1NSTD1,NGRD1,NLIN1,NCEN1
D       WRITE(6,1000)KSTD,NGRD,NLIN,NCEN,XLEN,METH,PROJ,AREA,DIAM,NCLASS,
D      1NSTD1,NGRD1,NLIN1,NCEN1,IREC,NSTD
   1000 FORMAT(T10,'ACSUM 1000',4I5,F10.0,I5,3F10.0,7I5)
        IF(KSTD.NE.NSTD)GOTO 100
        IREC=IREC+1
        IMETH=METH
C        TEST FOR SHORT FORMAT DATA OPTION AT TIME OF CREATION
C
        IF(IFM.GE.4)GO TO 40
C
C        TEST FOR CENTER MEASUREMENT OPTION
        IF(METH.EQ.0)GO TO 40
C   CALCULATE PROBABILITY OF SELECTION
     32 PRB=(SW+PROJ)/D
        IF((PRB.LT.0.).OR.(PRB.GT.1.))PRB=1.
C
C    ***  ACCUMULATE SUMS  ***
C
C   TEST FOR REGRESSION ESTIMATION OPTION
C
        IF(IANALY(NANALY,1).EQ.0)GOTO 40
        IF(AREA.GT.0.)GOTO 40
C   ESTIMATE CENTRE AREA BY REGRESSION EQUATION
        AREA=B*PROJ*DIAM+CEPT
        METH=1
C   TEST FOR INTERSECTION LENGTH METHOD OPTION
     40 IF(IFM.EQ.1.AND.IANALY(NANALY,2).EQ.0)GOTO 50
C   ACCUMULATE INTERSECTION LENGTHS BY GRID AND LINE
        XMAT(NGRD,NLIN)=XMAT(NGRD,NLIN)+XLEN
C   TEST FOR SHORT FORMAT OPTION
     50 IF(IFM.EQ.3)GO TO 81
        IF(IFM.GE.4)GO TO 70
C   TEST FOR TOTAL ESTIMATION BY CENTRE SIZE CLASSES OPTION
        IF(IFM.EQ.1.AND.IANALY(NANALY,3).EQ.0)GOTO 60
C
C    OMIT MULTIPLE ENCOUNTERS OF CENTRES ON THE SAME GRID
C
        IF(METH.EQ.8)GO TO 70
        IF(PROJ.LE.0.)GOTO 90
C   TEST FOR SIZE CLASSIFICATION METHOD OPTION FOR AREA ESTIMATES
        IF(IFM.EQ.2)GO TO 65
        IF(IANALY(NANALY,5).EQ.0)GOTO 92
        IF(IANALY(NANALY,5).EQ.2)GOTO 91
     65 IF(NCTPS(NANALY).EQ.0)GO TO 92
        CALL CLASS(CUTPTS,AREA,NANALY,NCLASS,FACTOR)
        GOTO 91
     92 NCLASS=1
C   ACCUMULATE CENTRE AREAS BY SIZE CLASS, GRID, AND LINE
     91 TSMAT(NCLASS,NGRD,NLIN)=TSMAT(NCLASS,NGRD,NLIN)+AREA/PRB
C   TEST FOR NUMBER ESTIMATION BY SIZE CLASSES OPTION
     60 IF(IFM.EQ.1.AND.IANALY(NANALY,4).EQ.0)GOTO 70
        IF(METH.EQ.8)GO TO 70
        IF(PROJ.EQ.0.)GOTO 90
C   TEST TO SEE WHETHER SIZE CLASSES ARE SPECIFIED BY CUTPOINTS, CODE,
C   OR ONE CLASS ONLY
        IF(IFM.EQ.2)GO TO 79
        IF(IANALY(NANALY,3).NE.0)GOTO 81
        IF(IANALY(NANALY,5).EQ.2)GO TO 81
        IF(IANALY(NANALY,5).EQ.0)GOTO 82
     79 IF(IFM.EQ.2.AND.NCTPS(NANALY).EQ.0)GOTO 82
C   DETERMINE CLASS NUMBER
        CALL CLASS(CUTPTS,AREA,NANALY,NCLASS,FACTOR)
        GOTO 81
     82 NCLASS=1
C   ACCUMULATE CENTRE NUMBERS BY SIZE CLASS, GRID, AND LINE
     81 SMAT(NCLASS,NGRD,NLIN)=SMAT(NCLASS,NGRD,NLIN)+1/PRB
C   ***  SUMS ACCUMULATED  ***
        IF(NCLASS.GT.MCLASS)MCLASS=NCLASS
     70 IF(ISWTCH(NANALY,3).EQ.1)CALL PRINT(LCOUNT)
        GOTO 30
C   ERROR MESSAGE
     90 CALL PAGE
C   SET ERROR INDICATOR
        IERR=1
        WRITE(6,85)NANALY,NSTD,NGRD,NLIN,NCEN
     85 FORMAT(//' ANALY',I2,'1',5X,'* * * * *  ERROR MESSAGE  * * * * *'/
       *'0',10X,'DIVISION BY ZERO - PROJECTION LENGTH',5X,'CENTER NO.1',4I
       *4///11X,'**********  SKIPPING TO NEXT ANALYSIS  **********')
    100 RETURN
        END
```

```
C
C*****************************************************************
        SUBROUTINE CLASS(CUTPTS,AREA,NANALY,NCLASS,FACTOR)
C   THIS SUBROUTINE DETERMINES WHICH SIZE CLASS EACH CENTRE BELONGS TO.
C
C*****************************************************************
        DIMENSION CUTPTS(10,10)
        NCLASS=1
C   CONVERT AREA TO STAND MEASUREMENT UNITS
        FAREA=AREA/FACTOR
        IF(FAREA.LE.CUTPTS(NANALY,1))GOTO 81
        DO 80 I=1,8
        NCLASS=I+1
     80 IF((CUTPTS(NANALY,I).LT.FAREA).AND.(FAREA.LE.CUTPTS(NANALY,
       1NCLASS)))
       *GOTO 81
        NCLASS=10
     81 RETURN
        END
C
C*****************************************************************
        SUBROUTINE STATSP(IOPT,NCLASS)
C   THIS SUBROUTINE PRINTS TOTAL THE SAMPLE ESTIMATES OF AREA AND
C   NUMBERS OF CENTRES (TABLE 1), ESIMATES OF VARIATION (TABLE 2)
C   AND ESTIMATES FOR TOTAL STAND (TABLE 3)
C*****************************************************************
C     IF IOPT = 1  -  LINE INTERSECTION METHOD
C               2  -  AREA ESTIMATES BY SIZE CLASS
C               3  -  NUMBER ESTIMATES BY SIZE CLASS
C*****************************************************************
        REAL*8 REFNO,MUNITS,SUNITS
        LOGICAL FLAG
C
        DIMENSION OMAT(6,20)
        DIMENSION XMAT(6,20),TSMAT(10,6,20),SMAT(10,6,20)
        DIMENSION SUMX(10),SUMY(10)
        DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
       *  CUTPTS(10,10),NCTPS(10),ISTD(10)
        COMMON/FKT/XMAT,TSMAT,SMAT,D
        COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
        COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
       * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
       1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
       1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
        IFM=IFORM(NANALY)
C   CALCULATE SAMPLE ESTIMATES
C   TEST FOR ESTIMATE METHOD OPTIONS
        IF(IOPT.EQ.2)GOTO 401
        IF(IOPT.GE.3)GOTO 402
C   AREA BY INTERSECTION LENGTH OPTION
        DO 400 I=1,MGRD
        DO 400 J=1,MLIN
D        WRITE(6,1000)MGRD,MLIN,XMAT(I,J)
D1000 FORMAT(T10,'STATSP 1000',2I5,F10.0)
    400 OMAT(I,J)=XMAT(I,J)
C   SET GRID DISTANCE AND AREA FACTORS
        DD=D
        FF=1.0
        GOTO 405
C   AREA BY SIZE CLASS OPTION
    401 DO 403 I=1,MGRD
        DO 403 J=1,MLIN
    403 OMAT(I,J)=TSMAT(NCLASS,I,J)
        DD=1.
        GOTO 405
C   NUMBERS BY SIZE CLASS OPTION
    402 DO 404 I=1,MGRD
        DO 404 J=1,MLIN
    404 OMAT(I,J)=SMAT(NCLASS,I,J)
        DD=1.
    405 CALL STATS(DD,OMAT,IOPT)
         IF(IOPT.EQ.1)GO TO 222
        IF(NCTPS(NANALY).LE.0)GO TO 222
C   CALCULATE SIZE CLASS UPPER AND LOWER BOUNDARIES
        IF(NCLASS.EQ.1)GO TO 66
        BOUND1=CUTPTS(NANALY,NCLASS-1)
     66 IF(NCLASS.EQ.1)BOUND1=0.0
        SUMBND(NCLASS,1)=BOUND1
        BOUND2=CUTPTS(NANALY,NCLASS)
        SUMBND(NCLASS,2)=BOUND2
222     IF(ATOT.LE.0.0.AND.ISWTCH(NANALY,4).EQ.0)GO TO 105
        IF(ISWTCH(NANALY,4).EQ.1)GO TO 300
99      IF(IOPT.GT.1)DD=1.
        IF(IOPT.NE.3)FF=FACTOR
        IF(IOPT.EQ.3)FF=1.00
        CALL TABLE2(ATOT,VTOT,VTOTA,CONLIM,NCLASS,IOPT)
C   CALCULATE ESTIMATES OF TOTAL STAND
        DO 100 I=1,MGRD
        DO 100 J=1,MLIN
    100 OMAT(I,J)=DD*OMAT(I,J)*MLIN/FF
        DO 101 I=1,MGRD
    101 SUMX(I)=DD*MLIN*SUMX(I)/FF
        DO 102 I=1,MLIN
    102 SUMY(I)=DD*MLIN*SUMY(I)/FF
        SX=DD*MLIN*SX/FF
C   STORE ESTIMATES FOR STAND SUMMARY TABLE
        ATOT=ATOT/FF
        IF(IOPT.EQ.1)SUMARE(1)=ATOT
        IF(IOPT.EQ.1)SUMARE(2)=CONLIM
        IF(IOPT.EQ.2)SUMTOT(NCLASS,1)=ATOT
        IF(IOPT.EQ.2)SUMTOT(NCLASS,2)=CONLIM
        IF(IOPT.EQ.3)SUMNO(NCLASS,1)=ATOT
        IF(IOPT.EQ.3)SUMNO(NCLASS,2)=CONLIM
        IF(IFM.NE.3.OR.IOPT.NE.3)GO TO 106
        AVBND=(SUMBND(NCLASS,1)+SUMBND(NCLASS,2))/2
        SUMTOT(NCLASS,1)=SUMNO(NCLASS,1)*AVBND
        SUMTOT(NCLASS,2)=SUMNO(NCLASS,2)*AVBND
106     IF(ISWTCH(NANALY,4).EQ.0)GO TO 105
        GO TO 301
 300    CALL PAGE
        IF(IOPT.EQ.2)GOTO 61
```

```
      IF(IOPT.GE.3)GOTO 71
      WRITE(6,1)NANALY,NSTD
    1 FORMAT(/1X,'ANALYSIS',I2,':',5X,'* * * * *  ESTIMATION OF ROOT ',
     1'ROT AREA USING THE INTERSECT LENGTH METHOD * * * * *'//
     21X,'STAND NO. ',I3)
      GOTO 152
   61 WRITE(6,11)NANALY
      IF(NCTPS(NANALY).GT.0)WRITE(6,111)NCLASS,NSTD
      IF(NCTPS(NANALY).EQ.0)WRITE(6,112)NSTD
   11 FORMAT(/1X,'ANALYSIS',I2,':',5X,'* * * * *  ESTIMATION OF ',
     1'ROOT ROT AREA BY CENTER SIZE CLASS USING AREA/',
     2'PROBABILITY OF OCCURRENCE * * * * *')
  111 FORMAT(' '//1X,'SIZE CLASS,',I3, ' STAND NO.',I3)
  112 FORMAT(' * * * * *'//1X,'ONE SIZE CLASS ONLY    STAND NO.',I3)
      GO TO 151
   71 WRITE(6,21)NANALY
      IF(NCTPS(NANALY).GT.0)WRITE(6,111)NCLASS,NSTD
      IF(NCTPS(NANALY).EQ.0)WRITE(6,112)NSTD
   21 FORMAT(/ ' ANALYSIS',I2,':',5X,'* * * * *  ESTIMATION OF TOTAL ',
     1'NUMBER OF CENTERS BY SIZE CLASS USING PROBABILITY OF OCCURRENCE')
      IF(NCLASS.GT.NCTPS(NANALY))WRITE(6,14)BOUND1,SUNITS
   14 FORMAT('+',T40,'SIZE CLASS = ',F10.2,' +  ',A8)
  151  IF(NCLASS.LE.NCTPS(NANALY))WRITE(6,15)BOUND1,BOUND2,SUNITS
  152  IF(ATOT.GT.0.0)GO TO 52
      WRITE(6,110)
   15 FORMAT('+',T40,'SIZE CLASS = ',F10.2,' -',F5.2,' ',A8)
  110 FORMAT(//T40,'NO CENTERS IN THIS SIZE CLASS')
      GO TO 105
   52 IF(IOPT.EQ.2)GOTO 63
      IF(IOPT.GE.3)GOTO 73
      WRITE(6,2)MUNITS
    2 FORMAT(//' TABLE 1:',10X,'ESTIMATE OF MEAN INTERSECTION LENGTH',
     1' IN SAMPLE (',A8,')')
      GOTO 53
   63 WRITE(6,13)MUNITS
   13 FORMAT(//' TABLE 1:',10X,
     1'ESTIMATE OF MEAN CENTER AREA IN SAMPLE (SQUARE',A8,')')
      GOTO 53
   73 WRITE(6,23)
   23 FORMAT(//' TABLE 1:',10X,
     1'ESTIMATE OF MEAN NUMBER OF CENTERS IN SAMPLE')
   53 CALL TABLE1(OMAT,MGRD,MLIN,SUMX,SUMY,SX)
      IF(MLIN.LE.15)GOTO 40
      WRITE(6,3)
    3 FORMAT(//  11X,'* * * * *  CONTINUED ON NEXT PAGE  * * * * *')
      CALL PAGE
C  PRINT ESTIMATES OF VARIATION
   40 WRITE(6,5)
    5 FORMAT(///' TABLE 2:',10X,'ESTIMATE OF VARIATION FOR TOTAL STAND')
      GO TO 99
  301  IF(MLIN.LE.5)GOTO 54
      WRITE(6,3)
      CALL PAGE
   54 IF (IOPT.EQ.1)WRITE(6,6)SUNITS
      IF(IOPT.EQ.2)WRITE(6,7)SUNITS
      IF(IOPT.EQ.3)WRITE (6,8)
    6 FORMAT(///1X,'TABLE 3:',10X,
     1' ESTIMATES OF MEAN ROOT ROT AREA IN TOTAL STAND (',A8,')')
    7 FORMAT(///1X,'TABLE 3:',10X,'ESTIMATE OF MEAN CENTER AREA IN ',
     1'TOTAL STAND (',A8,')')
    8 FORMAT(//1X,'TABLE 3:',10X,'ESTIMATE OF MEAN NUMBER OF CENTERS',
     1' IN TOTAL STAND')
      CALL TABLE1(OMAT,MGRD,MLIN,SUMX,SUMY,SX)
       IF(IOPT.GE.3)GOTO 72
      IF(IOPT.EQ.2)GOTO 62
      WRITE(6,4)ATOT,CONLIM,SUNITS
    4 FORMAT(// ' TOTAL ESTIMATED ROOT ROT AREA FOR STAND',F10.2,
     1' + OR -',F10.2,1X,A8)
      GOTO 105
   62 IF(NCTPS(NANALY).GT.0) WRITE(6,12)NCLASS,ATOT,CONLIM,SUNITS
      IF(NCTPS(NANALY).EQ.0)WRITE(6,121)ATOT,CONLIM,SUNITS
  121 FORMAT(//' TOTAL ESTIMATED ROOT ROT AREA (ONE SIZE CLASS ONLY): ',
     1F10.2,' + OR -',F10.2,1X,A8)
   12 FORMAT(// ' TOTAL ESTIMATED ROOT ROT AREA FOR SIZE CLASS',I3,
     1':',F10.2,' + OR -',F10.2,1X,A8)
      GOTO 105
   72 IF(NCTPS(NANALY).GT.0) WRITE(6,22)NCLASS, ATOT,CONLIM
      IF(NCTPS(NANALY).EQ.0)WRITE(6,221)ATOT,CONLIM
  221 FORMAT(//' TOTAL ESTIMATED NO. OF CENTERS (ONE SIZE CLASS ONLY): ',
     1F10.2,' + OR -',F10.2,1X,A8)
   22 FORMAT(// ' TOTAL ESTIMATED NO. OF CENTERS FOR SIZE CLASS',I3,':',
     1F5.1,' + OR -',F5.1)
  105 RETURN
      END
C*********************************************************************
      SUBROUTINE STATS(D,OMAT,IOPT)
C  THIS SUBROUTINE CALCULATES THE STATISTICS PRINTED BY STATSP.
C
C*********************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION OMAT(6,20)
      DIMENSION SUMX(10),SUMY(10)
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/EF/SUMX,SUMY,ATOT,VTOT,VTOTA,SX,CONLIM
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMND(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
C  INITIALIZE VARIABLES
      DO 10 I=1,MGRD
   10 SUMX(I)=0.
      DO 11 I=1,MLIN
   11 SUMY(I)=0.
      SSSX=0.
      SSX=0.
      SX=0.
C  ACCUMULATE ESTIMATES BY GRID AND LINE
      DO 20 I=1,MGRD
      DO 20 J=1,MLIN
C  ASSIGN ESTIMATE METHOD VALUES
      A=OMAT(I,J)
D      WRITE(6,1000)MGRD,MLIN,A
D1000     FORMAT(T10,'STATS 1000',2I5,F10.0)
C  ACCUMULATE SUMS AND SUMSQUARES
      SUMY(J)=SUMY(J)+A
      SUMX(I)=SUMX(I)+A
      SSX=SSX+A*A
      SX=SX+A
   20 CONTINUE
      DO 30 I=1,MGRD
   30 SSSX=SSSX+SUMX(I)**2
C  CALCULATE STATS.
      ATOT=D*SX/MGRD
      MN=MGRD*MLIN
C  CALCULATE MEANS AND VARIANCES
      IF(MGRD.EQ.1)GO TO 41
C  FOR GRIDS
      VTOT=D*D*(SSSX-(SX*SX)/MGRD)/MGRD*(MGRD-1)
      GO TO 40
   41 VTOT=10.E30
C  FOR LINES
   40 VTOTA=MLIN*D*D*(SSX-(SX*SX)/MN)/((MN-1)*MGRD)
      DO 50 I=1,MGRD
   50 SUMX(I)=SUMX(I)/MLIN
      DO 51 I=1,MLIN
   51  SUMY(I)=SUMY(I)/MGRD
      SX=SX/(MGRD*MLIN)
      RETURN
      END
C
C*********************************************************************
      SUBROUTINE TABLE1(XMAT,MGRD,MLIN,SUMX,SUMY,SX)
C  THIS SUBROUTINE PRINTS A TABLE WITH ROW AND COLUMN HEADINGS
C
C*********************************************************************
      DIMENSION XMAT(6,20),SUMX(10),SUMY(10)
      REAL DASH(30),FMT1(13),FMT2(12),FMT3(13),FMT4(7),DIGIT(10)
      DATA DASH/30*'----'/
      ISTOP=(MGRD+1)*4+3
C  FORMAT STATEMENTS FOR VARIABLE LENGTH HEADINGS
 150  FORMAT(/T5,'LINE',T11,'I',3X,1(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
 160  FORMAT(/T5,'LINE',T11,'I',3X,2(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
 170  FORMAT(/T5,'LINE',T11,'I',3X,3(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
 180  FORMAT(/T5,'LINE',T11,'I',3X,4(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
 190  FORMAT(/T5,'LINE',T11,'I',3X,5(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
 195  FORMAT(/T5,'LINE',T11,'I',3X,6(3X,'GRID',I3,2X,'I',3X),3X,'MEANS')
C  FORMAT STATEMENTS FOR RESULTS
 102  FORMAT(T5,I3,3X,'I',7(F11.2,4X,'I'))
 103  FORMAT(T5,'MEANS',T11,'I',7(F11.2,5X))
C  WRITE TABLE
      GO TO(50,60,70,80,90,95)MGRD
 50   WRITE(6,150)(I,I=1,MGRD)
      GO TO 100
 60   WRITE(6,160)(I,I=1,MGRD)
      GO TO 100
 70   WRITE(6,170)(I,I=1,MGRD)
      GO TO 100
 80   WRITE(6,180)(I,I=1,MGRD)
      GO TO 100
 90   WRITE(6,190)(I,I=1,MGRD)
      GO TO 100
 95   WRITE(6,195)(I,I=1,MGRD)
 100  WRITE(6,110)(DASH(I),I=1,ISTOP)
  110 FORMAT(' ',30A4)
      DO 30 NLIN=1,MLIN
      WRITE(6,102)NLIN,(XMAT(J,NLIN),J=1,MGRD),SUMY(NLIN)
 30   CONTINUE
      WRITE(6,110)(DASH(I),I=1,ISTOP)
      WRITE(6,103)(SUMX(I),I=1,MGRD),SX
      WRITE(6,110)(DASH(I),I=1,ISTOP)
C
      RETURN
C
      END
C
C*********************************************************************
      SUBROUTINE TABLE2(ATOT,VTOT,VTOTA,CONLIM,NCLASS,IOPT)
C  THIS SUBROUTINE CALCULATES STANDARD ERROR, COEFFICIENT OF VARIANCE
C  AND CONFIDENCE LIMITS FOR TOTAL STAND (TABLE 2)
C
C*********************************************************************
      DIMENSION TVAL(30)
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMND(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      DATA DASH/4H----/
      DATA TVAL/12.71,4.30,3.18,2.78,2.57,2.45,2.36,2.31,2.26,2.23,
     12.20,2.18,2.16,2.14,2.13,2.12,2.11,2.10,2.09,2.09,2.08,2.07,
     22.07,2.06,2.06,2.06,2.05,2.05,2.04,2.04/
      VTOTF=VTOT
C  CALCULATE VARIANCE FOR ESTIMATES BY GRIDS IN TOTAL STAND
      IF(IOPT.NE.3)VTOTF=VTOT/FACTOR**2
      SE=SQRT(VTOTF)
      N1=MGRD-1
      A=MLIN
      C1=TVAL(N1)*SE/SQRT(A)
D      WRITE(6,1000)N1,MLIN,SE,TVAL(N1)
D1000 FORMAT(T10,'TABLE2 1000',2I5,2F10.1)
      CV=SE/ATOT
      IF(IOPT.NE.3)CV=(SE/ATOT)*FACTOR
      IF(MGRD.EQ.1)CV=10.E30
      IF(MGRD.EQ.1)SE=10.E30
      N2=MGRD*MLIN-1
      IF(N2.GT.30)N2=30
      IF(ISWTCH(NANALY,4).NE.1)GO TO 10
```

```
      WRITE(6,1)
      WRITE(6,2)(DASH,I=1,30)
      WRITE(6,3)
      WRITE(6,4)VTOTF,N1,SE,CV,C1
      WRITE(6,3)
      WRITE(6,2)(DASH,I=1,30)
      WRITE(6,3)
C  CALCULATE VARIANCE FOR ESTIMATES BY INDIVIDUAL LINES IN
C  TOTAL STAND
 10   VTOTAF=VTOTA
      IF(IOPT.NE.3)VTOTAF=VTOTA/FACTOR**2
      SE=SQRT(VTOTAF)
      C2=TVAL(N2)*SE
      CV=SE/ATOT
      IF(IOPT.NE.3)CV=(SE/ATOT)*FACTOR
C      STORE CV'S FOR SUMMARY TABLE
      IF(IOPT.EQ.1)SUMARE(4)=CV*100
      IF(IOPT.EQ.2) SUMTOT(NCLASS,5)=CV*100
      IF(IOPT.EQ.3) SUMNO(NCLASS,5)=CV*100
      IF(ISWTCH(NANALY,4).NE.1)GO TO 20
      WRITE(6,5)VTOTAF,N2,SE,CV,C2
      WRITE(6,2)(DASH,I=1,30)
    1 FORMAT(11H0ESTIMATION,9X,1H],6X,8HVARIANCE,5X,1H],5X,10HDEGREES OF
     *,4X,1H],6X,8HSTANDARD,5X,1H]4X,11HCOEFFICIENT,4X,1H]
     34X,10HCONFIDENCE,5X,1H]/7H METHOD,13X,
     *1H],6X,8HESTIMATE,5X,1H],5X,7HFREEDOM,7X,1H],6X,
     49HDEVIATION,4X,1H],
     *2X,12HOF VARIATION,5X,1H],4X,12HLIMIT (+OR-),3X,1H])
    2 FORMAT(' ',30A4)
    3 FORMAT(' ',19X,1H],T121,1H])
    4 FORMAT(9H METHOD 1,11X,1H],F15.3,2X,I15,7X,F14.3,8X,
     1F14.3,F18.3,T121,1H]/1X,
     210H(BY GRIDS),9X,1H],T121,1H])
    5 FORMAT(1X,9H METHOD 2,10X,1H],F14.3,I18,F21.3,4X,F18.3,5X
     1,F14.3,T121,1H)/1X,
     210H(BY LINES),9X,1H],T121,1H])
 20   CONLIM=AMIN1(C1,C2)
      RETURN
      END
C
C*****************************************************************
      SUBROUTINE PRINT(LCOUNT)
C  THIS SUBROUTINE PRINTS THE LIST OF CENTER MEASUREMENTS
C
C*****************************************************************
      REAL*8 REFNO,MUNITS,SUNITS
      LOGICAL FLAG
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      REAL*8 HEAD1(5)
      REAL HEAD2(5)
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,DIAM,AREA,METH,PROJ,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBNO(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      LOGICAL*1 SYMBOL(30)
      DATA SYMBOL/'1','2','3','4','5','6','7','8','9','A',
     *            'B','C','D','E','F','G','H','I','J','K',
     *            'L','M','N','O','P','Q','R','S','T','U'/
      DATA HEAD1/8H  REGRES,8H  RADIAL,8H  LINEAR,8H  OTHER ,8H********/
      DATA HEAD2/4HSION,4H    ,4H    ,4H    ,4H****/
      DATA STARB/4H* * /,DASH/4H----/
      IFM=IFORM(NANALY)
      IF(IFM.GT.1.AND.IFM.LT.4)GOTO 600
C  DETERMINE WHICH TABLE TO PRINT
      IF(IANALY(NANALY,1).EQ.1)GOTO 600
      IF(IANALY(NANALY,3).EQ.1)GOTO 600
      IF(IANALY(NANALY,4).EQ.1)GOTO 600
C PRINT THE SHORT TABLE
      IF(LCOUNT.GT.25)GOTO 10
      IF(LCOUNT.NE.0)GOTO 20
C  WRITE TITLES ON NEW PAGE
   60 CALL PAGE
      WRITE(6,30)NANALY,(STARB,I=1,4),NSTD,(STARB,I=1,4),(DASH,I=1,6)
   30 FORMAT(//1X,'ANALYSIS',I2,'1',10X,4A4,'CALCULATIONS FOR STAND NO.',
     *I3,'  ',4A4//'0',10X,'CENTER ID.',T25,'INTERSECTION'/T25,'LENGTH'/
     *' ',11X,6A4)
C  WRITE THE RECORD AND INCREMENT LINE COUNT
   20 WRITE(6,40)NGRD,NLIN,NCEN,XLEN
   40 FORMAT('0',12X,3I3,F11.0)
      LCOUNT=LCOUNT+1
      GOTO 804
C  WRITE CONTINUED MESSAGE
   10 WRITE(6,50)
   50 FORMAT('0',10X,'( CONTINUED ON NEXT PAGE )')
      LCOUNT=0
      GOTO 60
600   IHEAD=METH
      IF(LCOUNT.GT.25)GOTO 501
      IF(LCOUNT.NE.0)GOTO 821
C  WRITE TITLE ON NEW PAGE
  500 CALL PAGE
      WRITE(6,800)NANALY,(STARB,I=1,4),NSTD,(STARB,I=1,4),MUNITS,
     1(DASH,I=1,24)
  800 FORMAT(//1X,'ANALYSIS',I2,'1',10X,4A4,' CALCULATIONS FOR STAND NO.
     *',I3,'  ',4A4/
     1T40,'MEASUREMENTS IN ',A8/
     *'0',' CENTER ID.  MAP   INTERSECTION AREA EST-  TOTAL          M
     *ID-     SIZE CLASS  SELECTION  ESTIMATION'/
     1T15,'SYMBOL  LENGTH',T36,'IMATE',T46,'PROJECTION',T59,'LENGTH',T80
     2,'PROBABILITY',T93,'METHOD'/' ',24A4,'-')
  821 IF(METH.EQ.8.OR.METH.EQ.9)GO TO 810
       WRITE(6,803)NGRD,NLIN,NCEN,SYMBOL(NCEN),XLEN,AREA,PROJ,DIAM,
     1NCLASS,PRB
  805 FORMAT('+',T95,A8,A4)
      IF(IFM.EQ.2)WRITE(6,806)
  806 FORMAT('+',T95,'AREA ESTIMATED')
      IF(IFM.EQ.3)WRITE(6,807)
  807 FORMAT('+',T95,'BY SIZE CLASS')
      IF(IFM.EQ.1)WRITE(6,805)HEAD1(IHEAD),HEAD2(IHEAD)
810   IF(METH.EQ.8.OR.METH.EQ.9)WRITE(6,822)NGRD,NLIN,NCEN,SYMBOL(NCEN),
     1XLEN,AREA,NGRD1,NLIN1,NCEN1
  822 FORMAT('0',2X,3I3,4X,A1,F11.0,F12.0,T100,'SAME AS CENTER',3I4)
  803 FORMAT('0',2X,3I3,4X,A1,F11.0,3F12.0,I10,F14.5)
      LCOUNT = LCOUNT + 1
      GOTO 804
C  WRITE CONTINUED MESSAGE
  501 WRITE(6,50)
      LCOUNT=0
      GOTO 500
  804 RETURN
      END
C
C*****************************************************************
      SUBROUTINE MAP(ANAL,IFM,IMAP)
C
C     THIS SUBROUTINE READS THE INPUT DATA AND WRITES A MAP OF THE
C     INFECTED AREAS ONTO A FILE. THE MAP IS THEN TRANSFERED FROM THIS
C     FILE ONTO THE OUTPUT FILE
C*****************************************************************
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBNO(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 SIDE,L,R,BLEND(10)
      INTEGER ANAL
      LOGICAL*1 LINPRT(396)
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,METH,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
      COMMON /MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *              DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
      DATA L/'L'/,R/'R'/,IBLNK/' '/
C
C     READ IN DATA AND STORE DATA FOR MAPPING OF TRANSECT AND BASE
C     LINES IN APPROPRIATE ARRAYS
C
C      DETERMINE ZERO END OF BASELINE
C
      IF(BLEND(ANAL).EQ.L)LN=1
      IF(BLEND(ANAL).EQ.R)LN=2
C
C      SET X AND Y SCALE FACTORS
C
      SCALX=SCALMP/10.0
      SCALY=SCALMP/6.0
      LINLTH(1,1) = 0
      IMPWTH = 60
      NRECS = 0
  2   READ(5,901,END=600)NSTD,NGRD,NLIN,NCEN,NTAPE,PT1,PT2,METH,YDST12,
     1YDST22,DTRDST,YDIST3,XDIST3,CNTANG,SIDE,PRJ1,PRJ2,DSTFOR,DSTBCK,
     *NLINES,   DIST,(DATA(1,J),J=1,10)
  901 FORMAT(3I1,I2,I3,2F3.0,I2,4F3.0,F2.0,F3.0,A1,4F3.0,I2,F2.0,10I3)
      IF(METH.EQ.8.OR.METH.EQ.9)GO TO 1
      TDIST=NTAPE*LTAPE
      PT1=PT1+TDIST
      PT2=PT2+TDIST
      YDST12=YDST12+TDIST
      YDST22=YDST22+TDIST
      YDIST3=YDIST3+TDIST
      DTRDST=DTRDST+TDIST
      IF(IFM.EQ.1.AND.NCEN.NE.0)CALL MAPERR(SIDE,CNTANG,PT2,FLAG,DATA,
     *NLINES,METH,NGRD,NLIN,NCEN,IMAP)
      IF(NRECS.EQ.0)NSTAND=NSTD
      IF(NSTD.NE.NSTAND)GO TO 600
      NRECS =NRECS + 1
      IF(NCEN.NE.0)GO TO 1
      IF(NLIN.EQ.1.AND.NGRD.EQ.1.AND.NRECS.EQ.1)STRREF=PT1
C
C      IDENTIFY LINE START AND END
C
      IF(PT2.EQ.0)STRPNT(NGRD,NLIN) = PT1
      IF(PT2.GT.0)LINLTH(NGRD,NLIN) = PT2
D      WRITE(6,5000)NSTD,NGRD,NLIN,NCEN,NTAPE,LTAPE,PT1,PT2,
D    1TDIST,STRPNT
 5000 FORMAT(' MAP F5000',6I5,4F5.0)
      IF(METH.NE.0)LINDIR(NGRD,NLIN) = METH
 1    GO TO 2
 600  IF(.NOT.FLAG)RETURN
      NOGRD = 0
 500  IF(GSP(ANAL,NOGRD+1).EQ.0.OR.NOGRD.EQ.6)GO TO 3
      NOGRD = NOGRD + 1
      GO TO 500
 3    CALL MAP1(NOGRD,ANAL,IMPWTH,LN,XSCLMX,YSCLMX)
C
C     SKIP MAPPING ROUTINE IF MAP SCALE TOO LARGE
C
      IF(XSCLMX.GT.0.OR.YSCLMX.GT.0)RETURN
C     DETERMINE METHOD OF MEASUREMENT USED AND BRANCH TO APPROPRIATE
C     SUBROUTINE TO PLACE MAP OF INFECTED AREA ON MAP FILE
C
      DO 100 I=1,NRECS
        BACKSPACE 5
 100  CONTINUE
      DO 5 I=1,NRECS
      READ(5,901,END=600)NSTD,NGRD,NLIN,NCEN,NTAPE,PT1,PT2,METH,YDST12,
     1YDST22,DTRDST,YDIST3,XDIST3,CNTANG,SIDE,PRJ1,PRJ2,DSTFOR,DSTBCK,
     *NLINES,   DIST,(DATA(1,J),J=1,10)
      IF(NCEN.EQ.0)GO TO 5
      TDIST=NTAPE*LTAPE
      PT1=PT1+TDIST
      PT2=PT2+TDIST
      YDST12=YDST12+TDIST
```

```
      YDST22=YDST22+TDIST
      YDIST3=YDIST3+TDIST
      DTRDST=DTRDST+TDIST
         IF(METH.NE.1.AND.IFM.EQ.1.AND.IMAP.GT.2)GO TO 6
         CALL MAP2(NGRD,NCEN,ANAL,IMPWTH,IFM,IMAP,LN)
         GO TO 5
 6       IF(METH.NE.2.OR.IFM.GT.1)GO TO 7
         CALL MAP3(NGRD,NCEN,ANAL,IMPWTH,LN)
         GO TO 5
 7       IF(METH.NE.3.OR.IFM.GT.1)GO TO 5
         YDIST4 = YDIST3
         XDIST4 = XDIST3
         CALL MAP4(NGRD,NCEN,ANAL,IMPWTH,LN)
 5    CONTINUE
C
C     WRITE MAP FILE ONTO OUTPUT FILE
C
 601  IWRTLP = (IMPLTH + 131) / 132
      JJ = 1
      DO 602 I=1,IWRTLP
         CALL PAGE
         WRITE(6,21)SCALMP,MUNITS,IBRNG(ANAL)
 21   FORMAT(' ','MAP OF INFECTION CENTRES','   SCALE : 1 INCH = ',F6.1,
     12X,A8,4X,'TRANSECT BEARING =',I4,'   SEE LIST FOR CENTER SYMBOLS',
     1////130('.'))
         JJJ = JJ + 131
         DO 9 II=1,IMPWTH
            READ(4'II)(LINPRT(J),J=1,JJJ)
            WRITE(6,20)(LINPRT(J),J=JJ,JJJ)
 20         FORMAT(' ',132A1)
 9       CONTINUE
         JJ = JJ + 132
 602  CONTINUE
      RETURN
      END
C
C*********************************************************************
       SUBROUTINE MAP1(NOGRD,ANAL,IMPWTH,LN,XSCLMX,YSCLMX)
C
C      THIS SUBROUTINE DRAWS A MAP OF THE TRANSECT LINES ON WHICH,
C      A MAP OF THE INFECTED AREAS CAN BE SUPERIMPOSED
C*********************************************************************
      LOGICAL FLAG
      REAL*8 REFNO,MUNITS,SUNITS
      DIMENSION IANALY(10,5),IREGR(10,10),IFORM(10),DBTWN(10),STRPW(10),
     *  CUTPTS(10,10),NCTPS(10),ISTD(10)
      COMMON/ALL/IANALY,IREGR,STRPW,DBTWN,IFORM,CUTPTS,IDATA,MANALY,
     * ISTD,NCTPS,MGRD,MLIN,NSTD,NANALY,B,MCLASS,FLAG,REFNO,MUNITS,
     1SUNITS,FACTOR,SIZE(10),CEPT,SUMBND(10,2),SUMTOT(10,5),SUMNO(10,5),
     1SUMARE(4),ISWTCH(10,5),NGRID(10),NTRNS(10)
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
       INTEGER OFBLRC(6,10),DFBL,DFBLMX,OFBL1,BASELN,DST,IBND2(6,10),
     1DNBL,OFBLMN,ONBLRC(6,10)
       LOGICAL*1 LINPRT(396),DIGIT(9),IDASH,IBLNK,ICOLON
       LOGICAL*1 SIDE
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,METH,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
       COMMON/MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *            DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
       INTEGER ANAL
       LOGICAL*1 IBASE(11),LINE,GRID,BLEND(10)
       DATA DIGIT/'1','2','3','4','5','6','7','8','9'/
       DATA IDASH/'-'/,IBLNK/' '/,ICOLON/':'/,LINE/'L'/,GRID/'G'/
       DATA IBASE/'(','B','A','S','E',' ','L','I','N','E',')'/
C
C      FILL IN MAP FILE
C
C      FIND TRANSECT LINE TERMINATION FARTHEST 'ABOVE' AND 'BELOW'
C      BASELINE TO DETERMINE UPPER AND LOWER BOUND OF MAP
C
       DFBLMX = 0
      XSCLMX =0.
      YSCLMX = 0.
       DO 10 I=1,NTRNS(ANAL)
          DO 41 IGRD=1,NOGRD
             IF(IGRD.NE.1.OR.I.NE.1)GO TO 30
             DFBL = STRREF + LINLTH(1,1)
             OFBLRC(1,1) = DFBL / SCALY+0.5
             IF(DFBL.GT.DFBLMX)DFBLMX=DFBL
             DNBL=STRREF
             IF(DNBL.LT.OFBLMN)OFBLMN=DNBL
             GO TO 41
 30          LGRD = IGRD - 1
             IF(LGRD.EQ.0)LGRD = NOGRD
             LTRNS = I
             IF(IGRD-1.EQ.0)LTRNS = I - 1
C
C      CALCULATE START AND END OF TRANSECT LINES
C
             IF(LINDIR(LGRD,LTRNS).EQ.2)GO TO 12
            IF(LINDIR(IGRD,I).EQ.1)GO TO 35
            DFBL = DFBL + STRPNT(IGRD,I)
            DNBL = DFBL - LINLTH(IGRD,I)
            GO TO 13
  35        DNBL = DNBL +STRPNT(IGRD,I)
            DFBL = DNBL + LINLTH(IGRD,I)
            GO TO 13
   12       IF(LINDIR(IGRD,I).EQ.2)GO TO 36
            DNBL = DNBL +STRPNT(IGRD,I)
            DFBL = DNBL +LINLTH(IGRD,I)
            GO TO 13
   36       DFBL = DFBL + STRPNT(IGRD,I)
            DNBL = DFBL - LINLTH(IGRD,I)
 13          IF(DFBL.GT.DFBLMX) DFBLMX=DFBL
             OFBLRC(IGRD,I) = DFBL / SCALY+0.5
              ONBLRC(IGRD,I)=DNBL/SCALY + 0.5
             IF(DNBL.LT.OFBLMN)OFBLMN=DNBL
 41       CONTINUE
 10    CONTINUE
C
C      DETERMINE MAP WIDTH
C
      MXNUM=DFBLMX-OFBLMN
      IMPWTH=MXNUM/SCALY+4
      IF(IMPWTH.LE.160)GO TO 20
C
C      COMPUTE MAXIMUM SCALE POSSIBLE FOR 3 X 3 PAGE MAP
C
      YSCLMX=(MXNUM*6.0)/160
      WRITE(6,100)YSCLMX
  100 FORMAT(' '//' MAP WIDTH GREATER THAN THREE PAGES. MINIMUM POSSIBLE
    1 SCALE  = ', F5.1,' UNITS PER INCH')
      RETURN
   20 CONTINUE
C
C      BLANK OUT MAP FILE
C
       DO 998 I=1,396
          LINPRT(I)=IBLNK
 998   CONTINUE
       DO 999 I=1,IMPWTH
          WRITE(4'I)(LINPRT(L),L=1,396)
 999   CONTINUE
C
C      PLACE TRANSECT LINES IN MAP FILE
C
       BASELN = DFBLMX / SCALY+0.5
       IMPLTH=(NTRNS(ANAL)+1)*DSTWN(ANAL)/SCALX+25
      IF(IMPLTH.LE.396)GO TO 25
       XSCLMX =(NTRNS(ANAL)+1)*DSTWN(ANAL)*10.0/370
      WRITE(6,101)XSCLMX
  101 FORMAT(' '//' MAP LENGTH GREATER THAN THREE PAGES. MINIMUM POSSIBLE
    1 SCALE  = ', F5.1,' UNITS PER INCH')
      RETURN
   25 CONTINUE
       DO 11 IGRD=1,NOGRD
          DO 4 I=1,NTRNS(ANAL)
             IBND1(IGRD,I) = BASELN - OFBLRC(IGRD,I)
             IBND2(IGRD,I) =BASELN-ONBLRC(IGRD,I)
 4        CONTINUE
          DO 3 J=1,IMPWTH
             READ(4'J)(LINPRT(L),L=1,IMPLTH)
             IF(LN.EQ.1)K = GSP(ANAL,IGRD)  / SCALX + 25
             IF(LN.EQ.2)K=(LBASE(ANAL)-GSP(ANAL,IGRD))/SCALX + 25
             DO 6 I=1,NTRNS(ANAL)
                IF(J.GE.IBND1(IGRD,I).AND.J.LE.IBND2(IGRD,I)) GO TO 5
                IF((IBND2(IGRD,I)+1).NE.J)GO TO 50
                LINPRT(K) = DIGIT(IGRD)
                LINPRT(K-1) = GRID
 50             IF((IBND2(IGRD,I)+2).NE.J)GO TO 51
                LINPRT(K) = DIGIT(I)
                LINPRT(K-1) = LINE
 51             IF(LN.EQ.1)K = K + DSTWN(ANAL)  / SCALX
             IF(LN.EQ.2)K=K-DSTWN(ANAL)/SCALX
                GO TO 6
 5              LINPRT(K) = ICOLON
                IF(LN.EQ.1)K = K + DSTWN(ANAL) / SCALX
                IF(LN.EQ.2)K=K-DSTWN(ANAL)/SCALX
 6           CONTINUE
             IF(J.GT.IMPWTH)GO TO 3
             WRITE(4'J)(LINPRT(L),L=1,IMPLTH)
 3        CONTINUE
 11    CONTINUE
C
C      WRITE BASELINE ONTO MAP FILE
C
        IBLTH=LBASE(ANAL)/SCALX +25
        READ(4'BASELN)(LINPRT(J),J=1,IBLTH)
       DO 7 K=21,IMPLTH
          IF(LINPRT(K).EQ.IBLNK)LINPRT(K) = IDASH
 7     CONTINUE
       DO 8 K=1,3
          J = (K - 1)*132 + 40
           IF((J+11).GT.IBLTH)GO TO 8
          DO 9 I=1,11
             LINPRT(J+I-1) = IBASE(I)
 9        CONTINUE
 8     CONTINUE
       IF(BASELN.GT.IMPWTH)RETURN
       WRITE(4'BASELN)(LINPRT(J),J=1,IBLTH)
C
C      PLACE SCALE AXIS ON MAP
C
      MXNUM=DFBLMX-DFBLMN
       CALL SCALE(IMPWTH,IMPLTH,SCALMP,SCALX,SCALY,MXNUM)
       RETURN
       END
C
C*********************************************************************
      SUBROUTINE MAP2(IGRD,NUM,ANAL,IMPWTH,IFM,IMAP,LN)
C
C     THIS SUBROUTINE SUPERIMPOSES INFECTED AREAS MEASURED RECTANGULARLY
C     ONTO THE MAP OF TRANSECT LINES AND BASELINE
C*********************************************************************
      LOGICAL*1 SIDE
      COMMON/CK/NGRD,NLIN,NCEN,XLEN,METH,NCLASS,PRB,
     1NSTD1,NGRD1,NLIN1,NCEN1
      COMMON /MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
     *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
     *            DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
     1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
      INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
     *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
      LOGICAL*1 LINPRT(3910),NUMBER(30),BLEND(6)
      INTEGER ANAL
      DATA NUMBER/'1','2','3','4','5','6','7','8','9','A',
     *            'B','C','D','E','F','G','H','I','J','K',
     *            'L','M','N','O','P','Q','R','S','T','U'/
C
```

```
C       DETERMINE INFECTED AREAS POSITION WITH RESPECT TO TRANSECT LINE
C       AN BRANCH TO APPROPRIATE SECTION
C
C       DETERMINE DIRECTION OF TRAVEL ALONG TRANSECT LINE AND ADJUST
C       VARIABLES TO ACCOUNT FOR DIRECTION
        IF(LINDIR(IGRD,NLIN).EQ.2)GO TO 150
        PT1 = LINLTH(IGRD,NLIN) - PT1
        PT2 = LINLTH(IGRD,NLIN) - PT2
        YDST12 = LINLTH(IGRD,NLIN) - YDST12
        YDST22 = LINLTH(IGRD,NLIN) - YDST22
        DTRDST = LINLTH(IGRD,NLIN) - DTRDST
        DSTFOR=-DSTFOR
        DSTBCK=-DSTBCK
        PRJ1=-PRJ1
        GO TO 300
150     PRJ2=-PRJ2
 300    CONTINUE
        IF(LN.EQ.1)
       1ITRPNT=(NLIN - 1)*(DSTWN(ANAL)/SCALX)+GSP(ANAL,IGRD) / SCALX + 25
        IF(LN.EQ.2)ITRPNT=
       1(LBASE(ANAL)-(NLIN-1)*DSTWN(ANAL))/SCALX-GSP(ANAL,IGRD)/SCALX+25
C       CALCULATE BOUNDARY POINTS
C
        DO 1 I=1,6
           GO TO (10,11,12,13,14,15) I
   10     IF(PT1.EQ.PT2)GO TO 1
           II = ITRPNT
           III = PT1 / SCALY + IBND1(IGRD,NLIN)+0.5
           GO TO 20
   11     IF(PT2.EQ.PT1)GO TO 1
           III = PT2 / SCALY + IBND1(IGRD,NLIN)+0.5
           GO TO 20
   12    IF(IFM.GT.1.OR.IMAP.EQ.1.OR.METH.GE.8)GO TO 1
           II = ITRPNT + ((PRJ1 + PRJ2) / 2) / SCALX+0.5
           III = (DTRDST +DSTFOR)/ SCALY + IBND1(IGRD,NLIN)+0.5
           GO TO 20
   13 IF(IFM.GT.1.OR.IMAP.EQ.1.OR.METH.GE.8)GO TO 1
           III =  (DTRDST-DSTBCK) / SCALY+IBND1(IGRD,NLIN)+0.5
           GO TO 20
   14     IF(IFM.EQ.4.OR.IMAP.EQ.1.OR.METH.GE.8)GO TO 1
           II = ITRPNT + PRJ1 / SCALX+0.5
           III = YDST12 / SCALY + IBND1(IGRD,NLIN)+0.5
           GO TO 20
   15 IF(IFM.EQ.4.OR.IMAP.EQ.1.OR.METH.GE.8)GO TO 1
           II = ITRPNT + PRJ2 / SCALX+0.5
           III = YDST22 / SCALY + IBND1(IGRD,NLIN)+0.5
           GO TO 20
C
C       ADD CALCULATED POINTS TO RECORD IN MAP FILE
C
 20     IF(III.GT.IMPWTH)GO TO 1
        IF(III.EQ.0)III=1
         IF(II.GT.IMPLTH.OR.II.LE.0)GO TO 1
        READ(4'III)(LINPRT(J),J=1,IMPLTH)
        LINPRT(II) = NUMBER(NUM)
        WRITE(4'III)(LINPRT(K),K=1,IMPLTH)
    1  CONTINUE
        RETURN
        END
C
C*******************************************************************
        SUBROUTINE MAP3(IGRD,NUM,ANAL,IMPWTH,LN)
C
C       THIS SUBROUTINE SUPERIMPOSES INFECTED AREAS MEASURED BY THE RADIAL
C       METHOD ONTO THE MAP OF TRANSECT LINES AND BASELINE
C*******************************************************************
        LOGICAL*1 SIDE,L,R,BLEND(10)
        COMMON/CK/NGRD,NLIN,NCEN,XLEN,METH,NCLASS,PRB,
       1NSTD1,NGRD1,NLIN1,NCEN1
        COMMON/MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
       *   DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
       *            DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
       1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
        INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
       *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
        INTEGER ANAL
        LOGICAL*1 IBLNK,LINPRT(396),LEFT,RIGHT,NUMBER(30)
        DATA RIGHT/'R'/
        DATA IBLNK/' '/,LEFT/'L'/,NUMBER/'1','2','3','4','5','6','7',
       *                                 '8','9','A','B','C','D','E',
       *                                 'F','G','H','I','J','K','L',
       *                                 'M','N','O','P','Q','R','S',
       *                                 'T','U'/
C
C       ADJUST VARIABLES TO ACCOUNT FOR DIRECTION OF TRAVERSAL OVER
C       TRANSECT LINE AND WHICH SIDE OF THE TRANSECT LINE THE AREA
C       IS ON
C
        IF (LINDIR(IGRD,NLIN).EQ.1)YDIST3=LINLTH(IGRD,NLIN)-YDIST3
        IF (SIDE.EQ.LEFT.AND.LINDIR(IGRD,NLIN).EQ.1)XDIST3=-XDIST3
        IF (SIDE.EQ.RIGHT.AND.LINDIR(IGRD,NLIN).EQ.2)XDIST3=-XDIST3
C
C       CALCULATE COORDINATES OF CENTRE POINT
C
        IF(LN.EQ.1)
       1ICNTPX = (NLIN - 1) * (DSTWN(ANAL) / SCALX) + XDIST3 / SCALX
       *           + GSP(ANAL,IGRD) / SCALX + 25
        IF(LN.EQ.2)
       1ICNTPX=(LBASE(ANAL)-(NLIN-1)*DSTWN(ANAL))/SCALX+XDIST3/SCALX
       2-GSP(ANAL,IGRD)/SCALX+25
        ICNTPY = IBND1(IGRD,NLIN) + YDIST3 / SCALY
        SUMANG = 0
        DIST = DIST / 57.295780
C
C       CALCULATE POSTIONS OF BOUNDARY POINTS ABOUNT AREA CENTRE
C
        DO 1 I=1,NLINES
           IXOFF = (SIN(SUMANG) * DATA(1,I)) / SCALX
           IYOFF = (COS(SUMANG) * DATA(1,I)) / SCALY
           II = ICNTPX + IXOFF
            IF(II.GT.IMPLTH)GO TO 2
           III = ICNTPY - IYOFF
C
C          ADD CALCULATED POINTS TO APPROPRIATE RECORD IN MAP FILE
C
        IF(III.GT.IMPWTH)GO TO 2
        IF(III.EQ.0)III=1
         IF(II.GT.IMPLTH.OR.II.LE.0)GO TO 2
           READ(4'III)(LINPRT(J),J=1,IMPLTH)
           LINPRT(II) = NUMBER(NUM)
           WRITE(4'III)(LINPRT(K),K=1,IMPLTH)
 2         SUMANG = SUMANG + DIST
 1      CONTINUE
        RETURN
        END
C
C*******************************************************************
         SUBROUTINE MAP4(IGRD,NUM,ANAL,IMPWTH,LN)
C
C        THIS SUBROUTINE SUPERIMPOSES INFECTED AREAS MEASURED
C        BY THE LINEAR METHOD ON THE TRANSECT LINE
C*******************************************************************
        LOGICAL*1 SIDE,BLEND(10)
        COMMON/CK/NGRD,NLIN,NCEN,XLEN,METH,NCLASS,PRB,
       1NSTD1,NGRD1,NLIN1,NCEN1
         COMMON/MAP/GSP,STRREF,STRPNT,LINLTH,IBND1,DTRDST,PRJ1,PRJ2,
       *    DST12,YDST22,XDIST2,PT1,PT2,YDIST3,XDIST3,DIST,
       *             DATA,NLINES,LINDIR,XDIST4,YDIST4,CNTANG,IMPLTH,DSTWN
       1  ,DSTFOR,DSTBCK,SCALMP,SCALX,SCALY,IBRNG,LTAPE,LBASE,SIDE,BLEND
        INTEGER GSP(10,6),STRREF,STRPNT(6,10),LINLTH(6,10)
       *,IBND1(6,10),DATA(5,10),LINDIR(6,10),DSTWN(10),LBASE(10),IBRNG(10)
         INTEGER ANAL
         LOGICAL*1 LINPRT(396),LEFT,RIGHT,NUMBER(30)
         DATA RIGHT/'R'/
         DATA LEFT/'L'/,NUMBER/'1','2','3','4','5','6','7','8','9','A',
       *                       'B','C','D','E','F','G','H','I','J','K',
       *                       'L','M','N','O','P','Q','R','S','T','U'/
C
C        ADJUST VARIABLE FOR DIRECTION OF TRAVEL ALONG TRSANSECT LINE
C        AND SIDE OF TRANSECT LINE AREA IS ON
C
         IF (LINDIR(IGRD,NLIN).EQ.1)YDIST4=LINLTH(IGRD,NLIN)-YDIST4
         IF(SIDE.EQ.LEFT.AND.LINDIR(IGRD,NLIN).EQ.1)XDIST4=-XDIST4
         IF(SIDE.EQ.RIGHT.AND.LINDIR(IGRD,NLIN).EQ.2)XDIST4=-XDIST4
C
C        CONVERT EXTERNAL ANGLES TO INTERNAL ANGLES
C
        IF(CNTANG.GT.180)CNTANG=CNTANG-180
C
C       DETERMINE AZIMUTH OF CENTER LINE
C
        IF(LINDIR(IGRD,NLIN).EQ.2)GO TO 20
        IF(PRJ1.GE.0.AND.SIDE.EQ.RIGHT)CNTANG=CNTANG+180
        IF(PRJ1.LT.0.AND.XDIST3.GT.ABS(PRJ1))CNTANG=CNTANG+180
        GO TO 30
   20 IF(PRJ2.GE.0.AND.SIDE.EQ.LEFT)CNTANG=CNTANG+180
        IF(PRJ2.LT.0.AND.XDIST3.GT.ABS(PRJ2))CNTANG=CNTANG+180
   30    IF(CNTANG.GT.360)CNTANG=CNTANG-360.
         CNTANG = CNTANG / 57.295780
C
C        CALCULATE START POINT AND END POINT OF CENTRE LINE
C
         IF(LN.EQ.1)
       1ISTPTX = (NLIN - 1) * (DSTWN(ANAL) / SCALX) + XDIST4 / SCALX
       *            + GSP(ANAL,IGRD) / SCALX + 25
         IF(LN.EQ.2)
       1ISTPTX=(LBASE(ANAL)-(NLIN-1)*DSTWN(ANAL))/SCALX +XDIST4/SCALX
       2-GSP(ANAL,IGRC)/SCALX +25
         ISTPTY = IBND1(IGRD,NLIN) + YDIST4 / SCALY
         IEDPTX = ISTPTX + (DIST * (NLINES + 1) * SIN(CNTANG)) / SCALX
         IEDPTY = ISTPTY - (DIST * (NLINES + 1) * COS(CNTANG)) / SCALY
         KK = 1
         DO 1 I=1,2
            GO TO (3,4) I
 3          II = ISTPTX
            III = ISTPTY
            GO TO 5
 4          II = IEDPTX
            III = IEDPTY
            GO TO 5
 1       CONTINUE
         KK = 2
         J = 1
         NSEG = NLINES * 2
C
C        CALCULATE POSITION OF BOUNDARY POINTS ON EITHER SIDE OF
C        CENTRE LINE
C
         DO 6 I=1,NSEG-1,2
            II = ISTPTX + (J * DIST * SIN(CNTANG)) / SCALX -
       *        (SIN(CNTANG-1.57) * DATA(1,I) / SCALX)
            III = ISTPTY - (J * DIST * COS(CNTANG)) / SCALY +
       *         (COS(CNTANG-1.57) * DATA(1,I) / SCALY)
            J = J + 1
            GO TO 5
 6       CONTINUE
         KK = 3
         J = 1
         DO 7 I=2,NSEG,2
            II = ISTPTX + (J * DIST * SIN(CNTANG)) / SCALX +
       *          (SIN(CNTANG-1.57) * DATA(1,I) / SCALX)
            III = ISTPTY - (J * DIST * COS(CNTANG)) / SCALY -
       *          (COS(CNTANG-1.57) * DATA(1,I) / SCALY)
            J = J + 1
            GO TO 5
 7       CONTINUE
         GO TO 100
C
C        PLACE POINTS ON APPROPRIATE RECORD IN MAP FILE
C
 5       IF(III.GT.IMPWTH)GO TO 8
         IF(III.EQ.0)III=1
```

```
      IF(II.GT.IMPLTH.OR.II.LE.0)GO TO 8
      READ(4'III)(LINPRT(L),L=1,IMPLTH)
      LINPRT(II) = NUMBER(NUM)
      WRITE(4'III)(LINPRT(K),K=1,IMPLTH)
 8    GO TO (1,6,7)KK
 100  RETURN
      END
C
C*******************************************************************
      SUBROUTINE MAPERR(SIDE,CNTANG,PT2,FLAG,DATA,NLINES,METH,NGRO,NLIN,
     *NCEN,IMAP)
C
C     THIS SUBBROUTINE CHECKS FOR ERRORS IN MAP RELATED VARIABLES
C*******************************************************************
      INTEGER DATA(5,10)
      LOGICAL FLAG
      LOGICAL*1 SIDE,L,R
      DATA L/'L'/,R/'R'/
      IF(IMAP.LE.2.OR.SIDE.EQ.L.OR.SIDE.EQ.R)GO TO 400
      FLAG = .FALSE.
      WRITE(6,1111)FLAG
 1111 FORMAT(' ',L4)
      WRITE(6,450)
 450  FORMAT('0','ERROR MESSAGE FOR MAPPING ROUTINE')
      WRITE(6,401)NGRO,NLIN,NCEN
 401  FORMAT(' ','CENTRE ID',3I3,' INVALID CHARACTER ENTERED FOR SIDE')
 400  IF(CNTANG.LE.360.AND.CNTANG.GE.0)GO TO 406
      FLAG = .FALSE.
      WRITE(6,450)
      WRITE(6,403)NGRO,NLIN,NCEN
 403  FORMAT(' ','CENTRE ID',3I3,' CENTRE LINE ANGLE BEYOND BOUNDS')
 406  J = NLINES
      IF(METH.EQ.3)J=NLINES*2
      IF(J.GE.10.OR.J.EQ.0)GO TO 102
      DO 100 I=1,J
         IF(DATA(1,I).NE.0)GO TO 100
         FLAG = .FALSE.
         WRITE(6,450)
         WRITE(6,408)NGRO,NLIN,NCEN
 408     FORMAT(' ','CENTRE ID',3I3,' NUMBER OF MEASUREMENTS AND MEASURE
     *MENTS LISTED DO NOT CORRESPOND')
 100  CONTINUE
      DO 101 I=J+1,10
         IF(DATA(1,I).EQ.0)GO TO 101
         FLAG = .FALSE.
         WRITE(6,450)
         WRITE(6,408)NGRO,NLIN,NCEN
 101  CONTINUE
 102  IF(NLINES.LE.5.OR.METH.NE.3)GO TO 105
      FLAG = .FALSE.
      WRITE(6,450)
      WRITE(6,409)NGRO,NLIN,NCEN
 409  FORMAT(' ','CENTRE ID',3I3,' TOO MANY MEASUREMENTS SPECIFIED FOR L
     *INEAR METHOD')
 105  IF(NLINES.LE.10)GO TO 103
      FLAG = .FALSE.
      WRITE(6,450)
      WRITE(6,410)NGRO,NLIN,NCEN
 410  FORMAT(' ','CENTRE ID',3I3,' TOO MANY MEASUREMENTS SPECIFIED FOR R
     *ADIAL METHOD')
 103  RETURN
      END
C
C*******************************************************************
      SUBROUTINE SCALE(IMPWTH,IMPLTH,SCALMP,SCALX,SCALY,MXNUM)
C
C     THIS SUBROUTINE PLACES A SCALE ALONG THE X AND Y AXIS OF THE
C     INFECTED AREAS MAP
C*******************************************************************
      LOGICAL*1 DOT,COMMA,LINPRT(396),IBLNK,DSTNCE(4),ZERO
      DATA DOT/'.'/,COMMA/','/,IBLNK/' '/,ZERO/'0'/
      NEXDIV=MXNUM/100
      NEXDIV=NEXDIV*100
      DO 2 I=1,IMPWTH
         READ(4'I)(LINPRT(L),L=1,IMPLTH)
         YNUM=MXNUM-(I-1)*SCALY
         B=NEXDIV
         IF(B.LT.100)GO TO 9
         REM=AMOD(YNUM,B)
         IF(REM.LT.SCALY)GO TO 3
   9     DO 1 J=1,3
            JJ = (J - 1) * 132 + 4
            LINPRT(JJ) = DOT
            IF(ABS(YNUM).LT.SCALY)LINPRT(JJ)=ZERO
 1       CONTINUE
         WRITE(4'I)(LINPRT(L),L=1,IMPLTH)
         GO TO 2
   3     NUM=YNUM/100
         NUM=NUM*100
         NEXDIV=NEXDIV-100
         ENCODE(4,998,DSTNCE)NUM
         DO 10 J=1,3
            JJ = (J - 1) * 132 + 1
            LINPRT(JJ) = DSTNCE(1)
            LINPRT(JJ+1) = DSTNCE(2)
            LINPRT(JJ+2) = DSTNCE(3)
            LINPRT(JJ+3) = DSTNCE(4)
 10      CONTINUE
         WRITE(4'I)(LINPRT(L),L=1,IMPLTH)
 2    CONTINUE
      DO 6 IJ=1,2
        NEXDIV=100
         DO 5 I=4,396
         XNUM=(I-1)*SCALX
         B=NEXDIV
          REM=AMOD(XNUM,B)
         IF(IJ.EQ.2)GO TO 7
        IF(REM.LT.SCALX)GO TO 20
  19        LINPRT(I) = DOT
            GO TO 5
 20         LINPRT(I) = COMMA
            GO TO 5
 7          IF(REM.GE.SCALX.OR.XNUM.LT.100)GO TO 5
            NUM=XNUM/100
            NUM=NUM*100
            NEXDIV=NEXDIV+100
            ENCODE(4,998,DSTNCE)NUM
 998        FORMAT(I4)
             IF(I.LT.3)GO TO 5
            LINPRT(I-2) = DSTNCE(1)
            LINPRT(I-1) = DSTNCE(2)
            LINPRT(I)   = DSTNCE(3)
            LINPRT(I+1) = DSTNCE(4)
 5       CONTINUE
         IF(IJ.EQ.1)GO TO 100
         WRITE(4'IMPWTH)(LINPRT(L),L=1,396)
         GO TO 6
 100     WRITE(4'IMPWTH-1)(LINPRT(L),L=1,396)
         DO 33 L=1,396
            LINPRT(L) = IBLNK
 33      CONTINUE
 6    CONTINUE
      RETURN
      END
```

ENVIRONMENT CANADA
CANADIAN FORESTRY SERVICE
PACIFIC FOREST RESEARCH CENTRE
VICTORIA, B.C., CANADA V8Z 1M5
BC - R - 4, OCTOBER, 1980