STEM ANALYSIS:

Sampling Techniques and Data Processing

By

© Johanna Kavanagh

A thesis submitted in partial fulfillment
for the requirement of the degree of
Master of Science in Forestry

Lakehead University
School of Forestry
Thunder Bay, Ontario

January 14, 1983

ABSTRACT

Stem analysis is a common forest mensurational technique used to gain individual tree information for various growth attributes. Interest in stem analysis has been renewed with the availability of computer technology, and an increased emphasis on forest growth and yield research.

This thesis deals with two main areas of concern. The first is the need for a new computer algorithm capable of processing stem analysis data produced by annual ring measurement equipment. The development and application of two new algorithms, DUFFNO and STEM, are discussed.

DUFFNO's main functions are; to aid in data verification, and to produce the Duff-Nolan sequences for the ring width data. STEM's main function is to calculate and produce tabular and graphical output of the growth attributes. able The second area of concern involves stem analysis sampling techniques. Nine trees were sectioned intensively to obtain true volume estimates, which were used as control values. These were compared statistically against volume estimates derived from sub-samples of the disc data. Reliable volume estimates, within 10 percent of control values at a confidence level of 95 percent, were obtained from three basic sampling methods. These were referred to as the "uniform section length" method, the "form class" method, and Romberg's method.

Recommendations for further research are offered.

ACKNOWLEDGEMENTS

I would like to extend a special thank you to Mr. H.G. Murchison, and to Mr. F.R. Clarke, for their willing help with fieldwork, their valuable advice, the time they gave to read the preceding drafts, and most of all for the encouragement which brightened the bad times.

A thank you goes to Dr. W.H. Carmean for his advice and criticism directed to previous drafts of this thesis.

Without the financial aid of Lakehead University and the National Science and Engineering Research Council, this thesis would not have been possible.

A special thanks goes to my family, Robert and Kelly, and to my parents, for their encouragement and cooperation during the last two years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

INTRODUCTION

Stem analysis is a commonly used mensurational technique capable of showing how a tree grows in height, in diameter, and in form (Husch, Miller and Beers 1972). Past development of tree height, diameter, form and volume, can be determined by annual ring counts and by measuring the increase in diameter on each cut surface of a felled and sectioned tree (Spurr 1952). Stem analysis is applicable to coniferous and broadleaved trees, and may also be used for multiple stemmed trees (Carron 1968).

Stem analysis is a labour-intensive, but valuable technique. Therefore, any improvements in the sampling techniques, or in data processing, will be beneficial to those intending to use it as part of their investigative methodology.

OBJECTIVES

This thesis has two objectives. One concerns the processing of stem analysis data. The other objective involves the area of stem analysis sampling techniques.

The first part of the thesis considers the development of a computer algorithm general enough to process and analyze stem analysis data, whether it is produced by traditional methods, or mechanically by the Holman Digimicrometer, one of several machines capable of measuring ring widths.

The second part of this thesis deals with the problem of determining how many sample discs and what bolt lengths are required per tree. Such information is needed in order to obtain reliable information from stem analysis, while incurring a minimum amount of sampling.

LITERATURE REVIEW

Various forest mensuration textbooks have been written by such authors as Loetsch, Zohrer and Haller (1973), Husch, Miller and Beers (1972), Carron (1968), Avery (1967), Spurr (1952), Chapman and Meyer (1949), Bruce and Schumacher (1942), and Graves (1907). They describe the general techniques for conducting a stem analysis study, and discuss various applications for the resultant information.

Stem analysis has been used extensively in growth studies, especially in the development of yield tables (Spurr 1952). When information is obtained at a number of positions along the stem, the technique is called a complete stem analysis (Carron 1968). Information obtained from only one position on the stem is called partial stem analysis data. Due to the high cost of sampling, it is often confined to trees already destined for felling.

APPLICATIONS OF STEM ANALYSIS

Stem analysis studies vary in purpose, and may include some or all of the measurements required to compute the growth in diameter, basal area, height and volume (Graves 1907). Stem analysis may be used to investigate one or more of the following problems.

1 - To determine at what age a given species under given conditions, will become merchantable (Graves 1907).

2 - To compare rates of height, diameter, basal area

and volume growth of two species, or the same species under different conditions (Graves 1907). The survival and dominance of species in mixed stands may also be of interest (Chapman and Meyer 1949).

3 - To illustrate the results of some type of silvicultural treatment, such as a thinning, or the initial spacing of a plantation (Chapman and Meyer 1949).

4 - To serve as an intermediate step in the determination of volume growth (Graves 1907).

5 - To study the effects of spacing at different ages, on the diameter and height growth of trees, and on their form and quality (Chapman and Meyer 1949).

6 - To study the ability of trees to recover after suppression (Chapman and Meyer 1949).

7 - To determine height growth patterns leading to the development of polymorphic site index curves (Carmean 1972).

ADVANTAGES OF STEM ANALYSIS

Well maintained permanent sample plots could provide similar height and diameter growth information, however, such plots require repeated measurements over a long time period. Accordingly, there are several distinct advantages in using stem analysis, rather than permanent sample plots.

Even for species with a wide geographical distribution, such as white spruce (*Picea glauca* (Moench) Voss), researchers have trouble locating equal numbers of stands for all ages over a range of growing sites. Researchers can overcome this problem by using overmature stands and stem analysis to gain information on young tree growth (Herman and DeMars 1970). Stem analysis is also very efficient in terms of the number of trees required to provide sufficient data (Herman and DeMars 1970).

DISADVANTAGES OF STEM ANALYSIS

One potential disadvantage can occur during the selection of sample trees for site index curve development. The dominant tree chosen for stem analysis may not always have been dominant throughout the life of the stand (Curtis 1964). This problem can be minimized by choosing the sample trees carefully.

A second disadvantage of stem analysis data, is the inherent dependence among successive measurements on the same sample tree (Herman and DeMars 1970).

SAMPLING TECHNIQUES FOR STEM ANALYSIS

It is possible to obtain measurements needed for stem analysis by climbing and boring trees. However, the usual procedure requires the sample trees to be felled and sectioned. The exact method followed in making a complete study, including points of stem measurement, and intervals between sections, varies according to tree form and desired precision (Avery 1967).

The purpose of the study often dictates which trees may be sampled. For site index studies, the tallest dominant or codominant tree of the desired species at each location is usually chosen (Curtis, DeMars and Herman 1974). Trees should be free of visible breaks, large forks, or other growth interruptions (Herman and DeMars 1970). Depending on the purpose of the study, trees may need to be selected from even-aged stands where trees are similar in chronological age, or in physiological age. It may also be necessary to chose similar sites, according to field evaluations of understory vegetation, soils, moisture conditions, slope and aspect (Herman and DeMars 1970). Some studies require trees of an average basal area. Often trees are selected from specific diameter classes or crown classes.

A complete stem analysis should include the length and diameter growth information for each section, total tree age, diameter at breast height (dbh), total height, and the clear and merchantable lengths of the bole. A full description of the tree, including tree class, a sketch of crown form, live crown length and width, bole form and state of health should also be included (Graves 1907). Shea and Armson (1972) also measured the positions of all the whorls on the main stem before sectioning the bole.

Text book descriptions of the general procedure for stem analysis tend to be similar and follow the general order below.

1 - The selected tree is felled and limbed. Broken tops should be reassembled before sectioning (Herman and DeMars 1970).

2 - The tree's species, dbh, total height, an estimate of years to attain stump height, and total age are recorded.

3 - Section lengths are measured and cut. The bole may be cut into regular merchantable bolt lengths for merchantable trees (Graves 1907). Uniform section lengths are not necessary or always desirable (Whyte 1971). Shea and Armson (1972) used section lengths of 30 centimetres (cm) for the first 14 metres (m) of tree height, and 15 cm intervals after that.

4 - If the actual analysis work will be done indoors, discs should be taken from the top of each section (Avery 1967). Herman and DeMars (1970) only removed the rectangular block with the chosen representative

radius for analysis to minimize the amount of wood taken to the lab.

5 - Measure and record the average inside-bark diameter at the top of each section. The average diameter is derived by arithmetically averaging the maximum and minimum inside-bark diameters (Avery 1967). For odd shaped discs, the minimum inside-bark diameter may not intersect the pith (Graves 1907).

6 - Locate and mark the average inside-bark radius, which is taken to be one-half the average inside-bark diameter, on each of the cross sections. The average radius must intersect the pith. Radii crossing rot, pitch pockets and aberrant annual ring configurations, should be avoided (Herman and DeMars 1970).

7 - Count the rings from the cambium, inward along the average radius, marking the end of every periodic interval. Periods of ten years are commonly used. Record the total number of rings, and the number remaining, if a fractional periodic interval exists near the pith.

8 - Measure the radius for each period, from the pith outward, and record the cumulative radius, or cumulative diameter, if preferred.

9 - Plot the height versus age relationship for the tree based on the age at each section, and the height of each section. The sections rarely occur on the bole at points coinciding with the position of terminal buds marking annual tree height, thus height-age curves will underestimate tree height growth. Carmean (1972) and Lenhart (1972) present alternative methods for correcting height underestimates caused by assuming that the height at the point of sectioning represents actual height.

1Ø - Plot the series of diameter measurements against height, connecting the points representing the same year, to produce taper curves. The terminal position along the height axis for each taper curve, is linearly interpolated from the height/age curve.

11 - Calculate the total volume under each taper curve. Smalian's formula may be used to calculate the volume for each section, and the tip volume may be calculated

as a conoid (Avery 1967). A polar planimeter may also be used to measure the area under each taper curve (Carron 1968), if height is plotted against diameter squared. This method is more time consuming than using a formula if many measurements were taken.

SELECTING SAMPLE DISC LOCATIONS

The volume of trees affected by butt swell may be overestimated, unless shorter sections are used for that portion of the bole (Carron 1968). Bell, Marshall and Johnson (1981) treated the sections of the first 2Ø percent of tree height as neilloids, using the two-end conic formula to calculate section volumes. The remaining sections were treated as paraboloids, using Smalian's formula. Smalian's, Newton's and Huber's formulae are the three commonly used equations for calculating section volumes. Most of the text books cited note the formulae and give examples to illustrate their use.

Whyte (1971) compared volume estimates for hypothetical diameters of geometric solids. The sections were considered as paraboloids, conoids and neiloids in turn. Comparisons showed that very small differences existed between the estimates, based on the three shapes. Differences were minimized by restricting the end diameter values for each section to remain within $\pm$ 25 millimetres (mm) of each other. No statistical analysis techniques were used to compare the results, possibly because the volume estimates were created with a simulation routine. Whyte (1971) did recommend the conic formula, the intermediate one, as the best choice. In practical terms, the implications of Whyte's findings would require the researcher to use shorter intervals for sectioning. The form of the tree would dictate how many discs would be required. The better the form (less taper), the fewer extra sections required. As the form becomes poorer (increased taper), a greater number of sections would be required in order to maintain the difference between the two end diameters to within the allowable $\pm$ 25 mm range.

Whyte (1971) recommended that sampling points should occur midway between internodes. Sampling midway between internodes works very well for young trees with recognizable internodes. However, internodes are often impossible to distinguish on old trees, and on trees displaying lammas growth characteristics. By choosing representative measurement points, different operators can take a set of sectional measurements on any one tree, which will not alter the estimate of total stem volume. Volume estimates from tree to tree will also be more consistent, and generally lower than obtained by traditional methods. The lower volume estimates are offset by the reduction in tree volume variance.

Goulding (1971) reported that the volume for a bole sectioned into equal lengths can be calculated more accurately by employing Romberg's method than by using Smalian's formula exclusively. The example given in the paper cited, shows that Romberg's method for 8 sections, 15.7 feet in length, gives a more accurate estimate of the true total volume, than Smalian's formula for 9, 10, or 11 equal sections. Goulding also suggests that researchers may not be sampling intensively enough throughout the length of the bole.

Shea and Armson (1972) used sections as short as 15 cm in the upper portion of the crown in order to sample as close as possible to the midpoint between the internodes.

STEM ANALYSIS DATA PROCESSING

Interest in stem analysis has been renewed with the development of computer technology, advanced statistical techniques, and an emphasis on research on forest growth and yield (Brace and Magar 1968) and biomass studies. The development of the Addo-X tree ring measuring device (Eklund 1949) had a pronounced impact on the interest in stem analysis reflected by its use in traditional areas such as cull studies, forest-productivity rating, volume table construction, product determination (Brace and Magar 1968), and forest growth simulation studies (Newnham 1964; Wilson 1964; Mitchell 1967). Stem analysis was also used for detailed individual tree growth studies (Duff and Nolan 1953, 1957, 1958; Tryon, Cantrell and Carvell 1957; Heger 1965a, 1965b), and studies of tree form, product potential, and tree-volume yields (Fries and Matern 1965).

Brace and Magar (1968) conducted a study which illustrated methods of improving the efficiency and flexibility of stem analysis using computer technology. Their methods produced volume summaries by section and for 5-year growth periods. Taper curves for individual trees were also plotted by computer. The purpose of the study was to compare manual methods of plotting stem profiles against producing volume summaries and stem profiles by computer.

Griffin and Yeatman (197Ø) implemented the automated procedure briefly described by Brace and Magar (1968), for stem analysis of 5Ø jack pine (Pinus banksiana Lamb.), using the Addo-X. They indicated that the main advantages for using this method were :

1 - Accuracy to the nearest ± Ø.Ø1 inches for radius measurements.

2 - The elimination of transcription errors, as data were automatically punched onto paper tape.

3 - A minimal amount of training was required for operators.

AUTOMATION OF DISC MEASUREMENT TECHNIQUES

One of several recent advances in modern technology applicable to stem analysis has been the development of the Holman Digimicrometer. This tree ring measuring device operates on principles similar to those of the Addo-X. The Holman Digimicrometer has a microprocessor which controls the measurement of rings, stores the information temporarily before transferring it to cassette tape, and arranges subsequent transmission to a main computer. All of these operations are controlled by computer logic.

The Holman Digimicrometer was designed and produced by Holman Electronics of Fredericton, New Brunswick, Canada. It is designed to measure bark thickness and ring widths, from the cambium inward to the pith, along any chosen radius. The ring width data for each disc are stored in the microprocessor memory, along with any information entered in the disc header, until the disc is completely measured. The disc header is an extra line of data entered by the operator. It usually contains information such as tree number, species code, disc height, disc number, and any other pertinent disc information. When measurement is complete the operator transfers all the information for the disc to cassette tape. At the end of a measuring session, the taped information is transmitted to a main computer for storage and editing prior to analysis. Unfortunately, the ring count on each disc is not automatically recorded by the digimicrometer after the disc is measured. It must be entered in the disc header during editing.

The Addo-X and the Holman Digimicrometer both measure annual ring widths. Griffin and Yeatman (197Ø) state that the Addo-X is accurate within ± Ø.Ø1 inches. Holman Electronics claim that the Holman Digimicrometer is accurate within ± Ø.Ø1 mm. The Addo-X measuring system suffers from internal mechanical slack, which becomes evident when backing up to measure a missed ring width. The measuring table must be reversed well beyond the missed ring, to tighten up the slack. Otherwise the hand operated system introduces an error in the ring width measurements. The Holman Digimicrometer measuring table is moved by a motor-driven, threaded screw, and is not affected by internal slack.

The Addo-X data were originally hand-transcribed. Griffin and Yeatman (197Ø) described a method which used an electronic device for punching data to paper tape, which could be keypunched mechanically at a later time. The

Holman information is transferred to cassette tape for intermediate storage, before being transmitted to the main computer. A direct link between the Holman Digimicrometer and a computer, eliminating the cassette player, has been tested and used successfully. A link to an Apple II microprocessor was also tested successfully at Lakehead University. The Ontario Ministry of Natural Resources at Maple, has replaced the cassette recorder with an Apple II, which stores data on floppy disks, until it can be transferred to the main computer. Special software has also been developed for the Apple II, by the Ontario Ministry of Natural Resources to aid in monitoring the ring width data as it is being measured.

OTHER MEASURING TECHNIQUES AND DEVICES

Renton, Lanasa and Tryon (1974) report the use of X-radiology to aid in identifying annual rings of slow-growing understory sugar maple (*Acer saccharum* Marsh.) for radius measurements.

Behman (1982) photographed small discs along with a millimetre ruler to provide scale. The negatives were projected through an enlarger onto a flat surface, so that a Numonics 237 Graphic Calculator, functioning as a free standing electronic planimeter, could be used to measure the outside and inside area of the first ring, to obtain the area of the last year's growth.

The Measu-Chron, also known as the Digital Positiometer in Europe, is now available through Micro-Measurement Technology, Bangor, Maine. The Measu-Chron was developed in 1979 by K. Johann in Vienna, Austria. It has application in the field of forestry for the measurement of annual rings. The Measu-Chron may also be used in the fields of zoology, biology, industry, and quality control.

Beek and Maessen (1981) describe the "Dorschkamp" equipment used for measuring annual growth rings. The system includes a measuring table with an object stage, a microscope, the "Sony Magnescale" SR 801 electronic ruler, the "Sony Magnescale" LF 100-12 electronic digital counter, the "Sharp Compet" 626 electronic calculator with a printer, and an interface built by "Dorschkamp" Research Institute for Forestry and Landscape Planning in Wageningen, the Netherlands. The interface converts the signals of the electronic ruler and counter into signals used by the calculator and printer.

STEM ANALYSIS ALGORITHMS

The capabilities of the Holman Digimicrometer and other measuring devices dramatically increase the amount of data that can be gathered and processed in a given period of time. Since data are automatically recorded, an operator can measure a greater number of discs per hour and obtain more information per disc with increased accuracy. An experienced operator can process a larger sample of trees per week, than was ever possible using traditional methods, providing that the device continues to function correctly.

With the improved capability of sampling a greater number of trees, or an increased number of discs per tree, the problem of processing and analyzing large data files becomes apparent.

Herman, DeMars and Woollard (1975) report the existence of two published programs, for computing and graphing tree growth from stem analysis data. The first was developed by Brace and Magar (1968). The other was a Fortran IV algorithm by Pluth and Cameron (1971). More recent algorithms include one published by Herman, DeMars and Woollard (1975), as well as two unpublished programs; one by Wang (1976) of Lakehead University and a second by Chapeskie and Fleet (1981) of the Ontario Ministry of Natural Resources, Brockville, Ontario.

A comparison of the available algorithms illustrates the advantages and disadvantages of each program, based on digimicrometer data processing, the type of information produced, and the ease of data handling.

All of the programs produce similar output. Each algorithm calculates individual tree height for a range of ages, and plots the uncorrected height/age curve. Taper curves are also produced by each of the programs. Visual inspection of the taper curves can alert the researcher to errors in the data, such as intersecting taper curves. Areas of inadequate sampling, especially in the stump section near the butt swell, may become evident on the plotted profile (Herman, DeMars and Woollard 1975).

Of the available programs, only the one by Chapeskie and Fleet (1981) was specifically intended to handle Holman Digimicrometer data. It is also the only algorithm which can process more than one tree at a time. The program's greatest drawback is that it is not general enough to process a large volume of ring width data collected from an old tree. The program tests each line of data for digits in the first two columns. If 19 is found in the first two columns, the program assumes that the remainder of the line is ring width data. Two blank columns denote a line containing a disc header. However, the algorithm is unable to deal with data lines beginning with anything other than 19 or two blank columns, and is therefore incapable of handling a tree dating back to the 1800's or earlier.

A second disadvantage is that the algorithm does not allow variable section lengths. Lengths of 1.0 metres have been incorporated into the program. The other available algorithms allow the option of variable section lengths, as specified by the user.

The algorithm by Herman, DeMars and Woollard (1975) was written specifically for site index research. Therefore, it is the only program which does not calculate individual tree diameters at breast height, basal area and volume over time.

On the other hand, it is the only program which processes more than one average radius per disc.

The Chapeskie and Fleet (1981) algorithm computes estimates of dbh, height, area and volume, at the time of cutting and for the previous one- and five-year growth periods.

Algorithms by Wang (1976), and Pluth and Cameron (1971) computed estimates for tree height, dbh, basal area and volume. Both programs calculate the periodic annual increment (PAI) and the mean annual increment (MAI) for each attribute, and reproduce the information in tabular and graphical form. The intervals for time periods are flexible and dictated by the user in both programs. Pluth and Cameron (1971) also incorporate optional models for volume calculations, including models based on stem form parameters. Wang's algorithm (1976) is only available for metric units. Pluth and Cameron (1971) originally used Imperial measuring units, but have an updated version for metric units only.

None of the available algorithms are capable of dealing with data for multiple stemmed trees.

LITERATURE REVIEW SUMMARY

As stated, this thesis deals with two main objectives, the first being the analysis and processing of stem analysis data and the second the sample disc locations along the bole.

With the improved capability of sampling a greater number of trees, or an increased number of discs per hour, the problem of processing large data files becomes evident. Therefore an important aspect of stem analysis is the availability of a suitable computer algorithm to process and analyze increased volumes of data. Existing algorithms are useful, but have several limiting features. Therefore there is a need for a new algorithm capable of handling data produced by the Holman Digimicrometer. At the same time, the algorithm should be general enough to process data from any other sources, such as traditional methods and other ring-width measuring devices.

A new algorithm could greatly facilitate the investigations of the other main objective, by reducing the time required to analyze stem analysis data. Therefore, the development of a computer algorithm is discussed first.

The literature review discussed two papers (Goulding 1971; Whyte 1971) which described the number of discs required for a reliable sample. A second part of the thesis will attempt to establish the number of sample discs required to obtain volume estimates, which are within ± 10 percent of the true volume, at a specified level of confidence.

DEVELOPING THE STEM ANALYSIS ALGORITHMS

Lakehead University purchased a Holman Digimicrometer in the fall of 1980. It was immediately used to obtain growth and yield information for jack pine and black spruce (*Picea mariana* (Mill.) B.S.P.) from stem analysis. Part of the contract agreement for the project required the production of graphical and tabular output of height, dbh, basal area and volume growth estimates for individual trees. The tabular output of the Duff-Nolan sequences (Duff and Nolan 1953), was also desired.

During the initial stages of project field work, it became obvious that the proposed algorithm should be able to deal with any number of trees on a per plot basis.

Flexibility in the matter of section lengths, and the number of allowable discs per tree, was also essential. Black spruce height growth was considerably slower in the lower bole than that for jack pine. Therefore, shorter section lengths in the lower bole were required to obtain reliable height growth information for black spruce. This change in methodology indicated the need for flexibility in section lengths and number of allowable discs per tree in the proposed algorithm. Field work also established a need for some way of processing trees with multiple tops, as they constituted a significant portion of the tree populations under study.

Many text books use examples of stem analysis, based on 10-year periods. The Holman Digimicrometer measures every annual ring, therefore it seemed beneficial to incorporate the ability of altering the time period used in calculating mean annual increments (MAI's) and periodic annual increments (PAI's).

In summary, desired algorithm characteristics include the ability to process Holman Digimicrometer data, or stem analysis data from any other source, as well as the following features :

1 - It must be able to process many trees at one time, preferably on a per plot basis.

2 - The algorithm must calculate individual tree height, dbh, basal area and volume estimates over time. The MAI and PAI values for each growth attribute should also be produced in tabular and graphical form for any desired period length.

3 - The program should be capable of dealing with multiple stem or multiple top data.

4 - The algorithm must be able to cope with variable section lengths, a flexible time length for periodic growth information, a large number of discs per tree, and any tree age (some of the oldest trees sampled were approximately 3ØØ years old).

5 - The program must be able to work with files stored on diskettes.

6 - Ideally, an algorithm would produce a series of taper curves, and a corrected height/age curve for each tree, following one of the alternative methods of

Carmean (1972) or Lenhart (1972), as an aid in the inspection of each set of tree data. However, the problem of crossing taper curves is minimized with the Holman Digimicrometer, because all ring widths are measured, and the data are automatically recorded. Therefore this feature was not as critical for a new algorithm.

During the initial weeks of disc measuring, it became evident that the digimicrometer data frequently contained recording errors. These should not be confused with operator mistakes made while entering the disc header. Loss of partial lines of data also caused problems and prompted the development of a smaller algorithm called DUFFNO. The main purposes of DUFFNO are to check the prepared data files before they are used in the stem analysis program, and to produce tabular output of the Duff-Nolan sequences (Duff and Nolan 1953) for each tree.

THE DUFFNO ALGORITHM

Appendix A (page 89), contains a flowchart, example output, and a complete listing for DUFFNO, including all subroutines called by the main program.

All the digimicrometer data for one plot are stored in a file called PLOT.DAT (Table 1, page 3Ø). Trees are arranged sequentially, starting with tree # 1 on the plot. Discs for each tree are also arranged sequentially, starting with the disc closest to ground level. Data for multiple tops, follows the data for the main stem. Disc data for the leaders are arranged in sequential order, by leader number, starting with the disc closest to the fork. The indented lines are disc headers, containing the plot number, tree number, disc number, leader number, the year cut, and the total average radius and single bark thickness in millimetres. The ring width data, also measured in millimetres, appear in the following lines in groups of ten, except for the fractional remainder near the pith.

Most of the discs can be measured in the correct sequence. Those which must be remeasured at a later time, can be placed in the correct order during editing.

TABLE 1 PLOT.DAT file. The indented lines are disc headers containing information for each disc in the following order: tree number, species code, disc number, disc height, ring count, year cut, year measured to, total radius, and single bark thickness. The remaining lines contain ring width data in mm.

```
 01  12  0001  0000 37  1981  0000  0171.07  03.85  HOGARTH
1981  04.94  04.73  03.46  06.81  04.39  04.90  05.83  05.44  04.59  03.45
1971  03.58  04.08  02.89  05.17  05.36  04.27  07.28  07.28  03.92  05.88
1961  05.82  08.67  06.70  05.88  05.86  04.85  04.36  06.05  04.51  03.46
1951  04.84  03.90  01.80  01.07  01.18  02.19  01.68
 01  12  0100  0100 32  1981  0000  0131.17  05.86
1981  04.88  02.90  02.09  02.42  01.40  01.93  04.01  02.59  02.93  03.49
1971  03.89  03.28  03.29  04.21  02.58  04.33  06.03  04.56  05.90  08.02
1961  05.47  06.89  04.82  05.63  04.95  04.24  04.49  04.89  03.87  04.13
1951  04.52  02.54
 01  12  0130  0130 31  1981  0000  0127.10  04.55
1981  04.13  02.39  02.37  02.39  00.89  02.05  03.68  03.43  03.05  03.90
1971  03.22  02.99  04.09  04.21  02.87  03.67  06.45  03.98  05.15  06.23
1961  04.95  05.83  05.07  05.31  05.17  04.99  04.78  04.71  04.57  04.63
1951  05.95
 01  12  0200  0200 30  1981  0000  0120.94  05.31
1981  03.63  02.29  02.44  02.16  01.41  01.91  04.20  03.03  03.31  04.24
1971  03.35  02.62  03.42  04.33  02.55  04.27  05.90  03.55  06.71  07.76
1961  05.72  06.88  05.92  05.99  05.56  04.18  04.56  03.44  02.58  03.03
 01  12  0300  0300 28  1981  0000  0116.26  06.37
1981  03.41  02.41  02.80  02.20  01.36  02.05  03.61  03.28  04.24  03.62
1971  03.90  03.39  03.86  04.86  03.18  04.61  06.62  04.71  05.77  06.28
1961  05.80  06.67  05.68  05.45  04.54  04.37  05.12  02.47
 01  12  0400  0400 26  1981  0000  0108.14  04.26
1981  02.89  02.22  02.39  01.97  01.60  01.80  03.36  03.23  03.78  04.95
1971  05.24  03.91  04.66  05.51  02.96  04.86  05.18  05.73  06.18  07.09
1961  06.40  06.58  04.90  04.21  03.46  03.08
 01  12  0500  0500 24  1981  0000  0103.78  04.31
1981  03.30  02.62  02.93  02.37  01.74  02.00  03.70  03.96  04.93  05.91
1971  04.97  04.10  03.25  05.35  03.44  05.62  05.74  05.45  07.31  06.98
1961  05.76  04.88  03.92  03.55
 01  12  0600  0600 22  1981  0000  0095.32  03.82
1981  04.38  02.95  03.60  03.15  02.33  02.02  04.22  03.89  05.35  06.49
1971  04.68  04.57  03.71  04.66  03.33  04.50  04.66  04.95  06.91  06.59
1961  04.27  04.11
 01  12  0701  0700 20  1981  0000  0093.19  03.38
1981  05.28  02.65  02.55  03.45  02.25  02.52  05.44  05.87  05.97  07.92
1971  05.82  05.75  04.09  07.64  05.35  04.67  03.73  05.28  02.91  04.05
 01  12  0800  0800 19  1981  0000  0081.21  03.13
1981  04.26  02.73  02.71  04.46  03.25  02.39  04.64  04.10  05.43  06.95
1971  06.82  05.94  04.75  03.65  02.68  05.41  04.43  02.41  04.20
 01  12  0900  0900 16  1981  0000  0070.90  03.38
1981  05.26  03.19  05.17  05.27  04.11  03.14  05.28  04.70  04.29  05.96
1971  05.94  04.78  03.92  03.24  01.50  05.15
 01  12  1001  1000 13  1981  0000  0060.75  03.02
1981  05.34  04.62  05.16  06.11  03.19  03.33  04.61  04.11  05.26  05.14
1971  04.51  03.74  05.63
 01  12  1100  1100 10  1981  0000  0045.89  02.77
1981  05.14  05.24  05.60  05.95  04.17  03.96  04.31  04.54  04.68  02.30
 01  12  1200  1200  8  1981  0000  0030.04  02.93
1981  04.16  03.68  04.59  04.67  04.27  03.47  03.31  01.89
 01  12  1300  1300  4  1981  0000  0012.77  01.81
1981  03.91  03.21  04.16  01.49
 01  12  1400  1400  2  1981  0000  0004.91  01.96
1981  03.60  01.31
```

A smaller data file called TREE.DAT (Table 2, page 32), containing information for individual trees is also required. This data file is required for processing the digimicrometer data through DUFFNO and STEM (the stem analysis program). TREE.DAT contains the plot number, tree number, species code, total age, total tree height, main bole height, number of leaders, total number of discs for the tree, the number of discs in the main bole, the number of discs in each leader, and the length of each leader, for each tree in the plot. The data in TREE.DAT are entered in the same order as the tree data appear in PLOT.DAT.

DUFFNO and STEM use an infinite do-loop, to allow any number of trees to be processed during a run. The run ends when the end of the TREE.DAT file is found. Only the data for one tree are retained in the program memory at any given time. The data are processed and any information which is to be saved is stored in the output file. The memory cells for each variable are then purged, before the data for the next tree are entered.

TABLE 2 TREE.DAT file. Each line of data contains all the information required to process a tree. This includes plot #, tree #, species code, total age, year cut, total height, bole height, # leaders, total # discs, # bole discs, # discs per leader, and leader lengths.

```
Ø1   112 37 1981 146Ø 146Ø 1142142        Ø5Ø
Ø1   212 37 1981 1352 1352 1131131        Ø52
Ø1   312 38 1981 142Ø 142Ø 1141141        Ø2Ø
Ø2   112 28 1981 11Ø6 11Ø6 11Ø71Ø7        Ø46
Ø2   212 28 1981  973  973 1 93 93        Ø53
Ø2   312 3Ø 1981 1Ø39 1Ø39 11Ø11Ø1        Ø39
Ø3   112 52 1981 1471 1471 1145145        Ø31
Ø3   212 82 1981 1488 1488 1148148        Ø18
Ø3   312 79 1981 1437 1437 1142142        Ø27
Ø4   112 37 1981 146Ø 146Ø 1142142        Ø5Ø
Ø4   212 37 1981 1352 1352 1131131        Ø52
Ø4   312 38 1981 142Ø 142Ø 1141141        Ø2Ø
Ø5   112 28 1981 11Ø6 11Ø6 11Ø71Ø7        Ø46
Ø5   212 28 1981  973  973 1 93 93        Ø53
Ø5   312 3Ø 1981 1Ø39 1Ø39 11Ø11Ø1        Ø39
Ø6   112 52 1981 1471 1471 1145145        Ø31
Ø6   212 82 1981 1488 1488 1148148        Ø18
Ø6   312 79 1981 1437 1437 1142142        Ø27
Ø7   112 37 1981 146Ø 146Ø 1142142        Ø5Ø
Ø7   212 37 1981 1352 1352 1131131        Ø52
Ø7   312 38 1981 142Ø 142Ø 1141141        Ø2Ø
Ø8   112 28 1981 11Ø6 11Ø6 11Ø71Ø7        Ø46
Ø8   212 28 1981  973  973 1 93 93        Ø53
Ø8   312 3Ø 1981 1Ø39 1Ø39 11Ø11Ø1        Ø39
Ø9   112 52 1981 1471 1471 1145145        Ø31
Ø9   212 82 1981 1488 1488 1148148        Ø18
Ø9   312 79 1981 1437 1437 1142142        Ø27
```

DUFFNO EXECUTION PROCEDURES

During execution of DUFFNO (see flowchart; Appendix A-1, page 90), the first line of data are taken from TREE.DAT. The necessary information includes the plot number, tree number, species code, total age, total tree height, bole height, the number of leaders, the total number of discs, the number of discs in the bole, the number of discs in each leader, and the leader lengths. A maximum of five leaders, including the main leader, are allowed. For single stem trees, the bole height is the same as the total tree height, and the number of discs in the bole are equal to the total number of discs in the tree. For single leader trees, the number of leader discs are declared to be zero.

For multiple top trees, the leader lengths are taken to be the distance from the fork to the tip of each leader. For a single stem tree, leader length is considered to be the tip length remaining beyond the last disc.

Total age, number of leaders, discs per tree, discs per bole, and discs per leader all function as integer values for ending do-loops.

The variable indicating the number of leaders, taken from the first line in TREE.DAT, determines whether the algorithm should proceed along path A for single leader trees, or path B for multiple leader trees (see flowchart; Appendix A-1, page 90). The algorithm then reads the disc data from PLOT.DAT, reading a disc header, and then the actual ring width data for the disc. The disc header repeats the plot number, tree number, and species code, which are followed by disc number, leader number, disc height, ring count, year cut, year measured to, total average radius and single bark thickness. Everything with the exception of the ring count, is automatically recorded by the digimicrometer. Ring counts are included during the editing of the raw data files. Ring count is used as an integer value to end the do-loop which reads the ring width data for the disc.

When all the data for one tree have been entered, the ring width data are converted from millimetres to centimetres and organized into tabular form. The file DUFF.DAT (Table 3, page 35), the file created by DUFFNO, contains ring width data, sorted and presented in the Duff-Nolan sequences.

TABLE 3 DUFF.DAT file. The Duff-Nolan sequences are displayed for Tree # 1, a white spruce.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

DISC NUMBER	0	10	13	20	30	40	50	60	70	80	90	100	110	120	130	140	0
BOLT LENGTH (M)	0.00	1.00	0.30	0.70	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.60
DISC HEIGHT (M)	0.00	1.00	1.30	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	14.60
RING COUNT	37	32	31	30	28	26	24	22	20	19	16	13	10	8	4	2	0
AGE (YEARS)	0.	5.	6.	7.	9.	11.	13.	15.	17.	18.	21.	24.	27.	29.	33.	35.	37.
SBT (CM)	0.385	0.586	0.455	0.531	0.637	0.426	0.431	0.382	0.338	0.313	0.338	0.302	0.277	0.293	0.181	0.196	0.000
AVE RADIUS (CM)	17.107	13.117	12.710	12.094	11.626	10.814	10.378	9.532	9.319	8.121	7.090	6.075	4.589	3.004	1.277	0.491	0.000
RING WIDTH (CM)																	
	0.494	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.473	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.346	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.681	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.439	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.490	0.488	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.583	0.290	0.413	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.544	0.209	0.239	0.363	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.459	0.242	0.237	0.229	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.345	0.140	0.239	0.244	0.341	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.358	0.193	0.089	0.216	0.241	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.408	0.401	0.205	0.141	0.280	0.289	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.289	0.259	0.368	0.191	0.220	0.222	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.517	0.293	0.343	0.420	0.136	0.239	0.330	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.536	0.349	0.305	0.303	0.205	0.197	0.262	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.427	0.389	0.390	0.331	0.361	0.160	0.293	0.438	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.728	0.328	0.322	0.424	0.328	0.180	0.237	0.295	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.728	0.329	0.299	0.335	0.424	0.336	0.174	0.360	0.528	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.392	0.421	0.409	0.262	0.362	0.323	0.200	0.315	0.265	0.426	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.588	0.258	0.421	0.342	0.390	0.378	0.370	0.233	0.255	0.273	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.582	0.433	0.287	0.433	0.339	0.495	0.396	0.202	0.345	0.271	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.867	0.603	0.367	0.255	0.386	0.524	0.493	0.422	0.225	0.446	0.526	0.000	0.000	0.000	0.000	0.000	0.000
	0.670	0.456	0.645	0.427	0.486	0.391	0.591	0.389	0.252	0.325	0.319	0.000	0.000	0.000	0.000	0.000	0.000
	0.588	0.590	0.398	0.590	0.318	0.466	0.497	0.535	0.544	0.239	0.517	0.000	0.000	0.000	0.000	0.000	0.000
	0.586	0.802	0.515	0.355	0.461	0.551	0.410	0.649	0.587	0.464	0.527	0.534	0.000	0.000	0.000	0.000	0.000
	0.485	0.547	0.623	0.671	0.662	0.296	0.325	0.468	0.597	0.410	0.411	0.462	0.000	0.000	0.000	0.000	0.000
	0.436	0.689	0.495	0.776	0.471	0.486	0.535	0.457	0.792	0.543	0.314	0.516	0.000	0.000	0.000	0.000	0.000
	0.605	0.482	0.583	0.572	0.577	0.518	0.344	0.371	0.582	0.695	0.528	0.611	0.514	0.000	0.000	0.000	0.000
	0.451	0.563	0.507	0.688	0.628	0.573	0.562	0.466	0.575	0.682	0.470	0.319	0.524	0.000	0.000	0.000	0.000
	0.346	0.495	0.531	0.592	0.580	0.618	0.574	0.333	0.409	0.594	0.429	0.333	0.560	0.416	0.000	0.000	0.000
	0.484	0.424	0.517	0.599	0.667	0.709	0.545	0.450	0.764	0.475	0.596	0.461	0.595	0.368	0.000	0.000	0.000
	0.390	0.449	0.499	0.556	0.568	0.640	0.731	0.466	0.535	0.365	0.594	0.411	0.417	0.459	0.000	0.000	0.000
	0.180	0.489	0.478	0.418	0.545	0.658	0.698	0.495	0.467	0.268	0.478	0.526	0.396	0.467	0.000	0.000	0.000
	0.107	0.387	0.471	0.456	0.454	0.490	0.576	0.691	0.373	0.541	0.392	0.514	0.431	0.427	0.391	0.000	0.000
	0.118	0.413	0.457	0.344	0.437	0.421	0.488	0.659	0.528	0.443	0.324	0.451	0.454	0.347	0.321	0.000	0.000
	0.219	0.452	0.463	0.258	0.512	0.346	0.392	0.427	0.291	0.241	0.150	0.374	0.468	0.331	0.416	0.360	0.000
	0.168	0.254	0.595	0.303	0.247	0.308	0.355	0.411	0.405	0.420	0.515	0.563	0.230	0.189	0.149	0.131	0.000

DUFFNO ERROR DETECTION

As DUFFNO reads the disc data, the ring width data for each disc are accumulated and compared to the total average radius given in the disc header. A warning message is issued if the two values do not agree. The warning will not prevent the algorithm from continuing through the file. If one observation is missing, due to digimicrometer transmission problems, it is still possible to estimate the missing value, by assuming that the total average radius given in the header is correct. The procedure is based on the assumption that all ring widths are without error. However, if there is unconscious bias on the part of the operator in reading ring widths, the accumulated bias for all rings will then be included in the estimate of the missing ring width. Computer software on the VAX 11/780 also issues warnings for incorrect characters, such as alphanumerics within the data set, and indicates problems with a file missing discs or trees.

A software error message may be issued if an incorrect ring count is encountered. If the given ring count is less than the actual value, an entire line of data could be missed. The program is then unable to continue, because it assumes that the next line of data is a disc header, when in fact it is ring width data. The error message issued is caused by the mismatch between the data and the format statement required to read it. If the missing ring width does not cause a line of data to be missed, DUFFNO will issue the warning message given for the total average radius not equal to the accumulated ring widths.

If the total number of discs per tree is entered incorrectly in TREE.DAT, one of two errors may occur. Too few discs will cause the last discs to be included at the beginning of the next tree. This may leave extra discs at the end of the file. This will not activate any warnings, unless a format statement is unable to read a line of data. If too many discs are reported, the first few discs of the next tree will be included at the end of the first tree. This will cause a shortage of discs at the end of the file, causing the computer to abort the run and to issue an error message indicating that the file has ended too soon.

No error messages are issued if the discs are not in consecutive order. It is the user's responsibility to check the tabular output of DUFF.DAT for discs out of sequence and incorrect disc heights leading to erroneous bolt lengths.

Several runs of DUFFNO may be required to eliminate all the errors in the data sets. When all the errors have been corrected, DUFFNO should be executed once more, so that the tabular output can be double-checked. The corrected data may now be used successfully in STEM.

THE STEM ALGORITHM

A flowchart, example output, and a complete listing for STEM and all its subroutines, is given in Appendix B, page 1Ø8.

The raw data files PLOT.DAT and TREE.DAT are also used by the algorithm STEM. This program analyzes these files and creates a third file, called STEM.DAT. The file contains tabular output of the disc data, as well as tabular and graphical output of height, dbh, basal area, and volume growth. It is important to remember that STEM does not check the raw data files for errors. If DUFFNO is used as a debugging algorithm, only one run of STEM will be required. This saves execution time and paper, because STEM consumes greater amounts of both in the course of producing the tables and graphs for individual trees.

STEM's operational procedures are similar to those of DUFFNO. After reading the tree header, the program follows one of two major paths, depending on the number of leaders the tree has (see flowchart; Appendix B-1, page 1Ø9). The disc data are subsequently entered and processed. At the end of each calculation section, pertinent information is printed in the output file, STEM.DAT. This eliminates the need for saving all the information in the program. The raw data stored in STEM.DAT, (Table 4, page 4Ø), display cumulative radius values for each year as measured along the average radius of each disc sampled from the tree.

The main program calculates the values for height, dbh, basal area, and volume for all the taper curves. A general subroutine calculates MAI and PAI for each of the growth attributes. The subroutine also graphs the values, and outputs the information to STEM.DAT.

Values for height are calculated using straight-line interpolations. Values for dbh are calculated in the same manner, unless the dbh disc is measured. Basal area is based on the dbh values. Smalian's formula is used to calculate section volumes, including the tip. The stump is assumed to be cylindrical for the purposes of volume

TABLE 4 STEM.DAT file. Radius (inside bark) measurements are displayed for Tree # 1, a white spruce.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

DISC NUMBER	0	10	13	20	30	40	50	60	70	80	90	100	110	120	130	140	0
BOLT LENGTH (M)	0.00	1.00	0.30	0.70	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.60
DISC HEIGHT (M)	0.00	1.00	1.30	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	14.60
RING COUNT	37	32	31	30	28	26	24	22	20	19	16	13	10	8	4	2	0
AGE (YEARS)	0.	5.	6.	7.	9.	11.	13.	15.	17.	18.	21.	24.	27.	29.	33.	35.	37.
SBT (CM)	0.385	0.586	0.455	0.531	0.637	0.426	0.431	0.382	0.338	0.313	0.338	0.302	0.277	0.293	0.181	0.196	0.000
RIB (CM)																	
1981	17.107	13.117	12.710	12.094	11.626	10.814	10.378	9.532	9.319	P 121	7.090	6.075	4.589	3.004	1.277	0.491	0.000
1980	16.613	12.629	12.297	11.731	11.285	10.525	10.048	9.094	8.791	7.695	6.564	5.541	4.075	2.588	0.886	0.131	0.000
1979	16.140	12.339	12.058	11.502	11.044	10.303	9.786	8.799	8.526	7.422	6.245	5.079	3.551	2.220	0.565	0.000	0.000
1978	15.794	12.130	11.821	11.258	10.764	10.064	9.493	8.439	8.271	7.151	5.728	4.563	2.991	1.761	0.149	0.000	0.000
1977	15.113	11.888	11.582	11.042	10.544	9.867	9.256	8.124	7.926	6.705	5.201	3.952	2.396	1.294	0.000	0.000	0.000
1976	14.674	11.748	11.493	10.901	10.408	9.707	9.082	7.891	7.701	6.380	4.790	3.633	1.979	0.867	0.000	0.000	0.000
1975	14.184	11.555	11.288	10.710	10.203	9.527	8.882	7.689	7.449	6.141	4.476	3.300	1.583	0.520	0.000	0.000	0.000
1974	13.601	11.154	10.920	10.290	9.842	9.191	8.512	7.267	6.905	5.677	3.948	2.839	1.152	0.189	0.000	0.000	0.000
1973	13.057	10.895	10.577	9.987	9.514	8.868	8.116	6.878	6.318	5.267	3.478	2.428	0.698	0.000	0.000	0.000	0.000
1972	12.598	10.602	10.272	9.656	9.090	8.490	7.623	6.343	5.721	4.724	3.049	1.902	0.230	0.000	0.000	0.000	0.000
1971	12.253	10.253	9.882	9.232	8.728	7.995	7.032	5.694	4.929	4.029	2.453	1.388	0.000	0.000	0.000	0.000	0.000
1970	11.895	9.864	9.560	8.897	8.338	7.471	6.535	5.226	4.347	3.347	1.859	0.937	0.000	0.000	0.000	0.000	0.000
1969	11.487	9.536	9.261	8.635	7.999	7.080	6.125	4.769	3.772	2.753	1.381	0.563	0.000	0.000	0.000	0.000	0.000
1968	11.198	9.207	8.852	8.293	7.613	6.614	5.800	4.398	3.363	2.278	0.989	0.000	0.000	0.000	0.000	0.000	0.000
1967	10.681	8.786	8.431	7.860	7.127	6.063	5.265	3.932	2.599	1.913	0.665	0.000	0.000	0.000	0.000	0.000	0.000
1966	10.145	8.528	8.144	7.605	6.809	5.767	4.921	3.599	2.064	1.645	0.515	0.000	0.000	0.000	0.000	0.000	0.000
1965	9.718	8.095	7.777	7.178	6.348	5.281	4.359	3.149	1.597	1.104	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1964	8.990	7.492	7.132	6.588	5.686	4.763	3.785	2.683	1.224	0.661	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1963	8.262	7.036	6.734	6.233	5.215	4.190	3.240	2.188	0.696	0.420	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1962	7.870	6.446	6.219	5.562	4.638	3.572	2.509	1.497	0.405	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1961	7.282	5.644	5.596	4.786	4.010	2.863	1.811	0.838	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1960	6.700	5.097	5.101	4.214	3.430	2.223	1.235	0.411	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1959	5.833	4.408	4.518	3.526	2.763	1.565	0.747	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1958	5.163	3.926	4.011	2.934	2.195	1.075	0.355	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1957	4.575	3.363	3.480	2.335	1.650	0.654	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1956	3.989	2.868	2.963	1.779	1.196	0.308	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1955	3.504	2.444	2.464	1.361	0.759	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1954	3.068	1.995	1.986	0.905	0.247	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1953	2.463	1.506	1.515	0.561	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1952	2.012	1.119	1.058	0.303	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1951	1.666	0.706	0.595	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1950	1.182	0.254	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1949	0.792	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1948	0.612	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1947	0.505	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1946	0.387	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1945	0.168	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

calculation. Stump flare is best avoided by sampling more intensively at the base of the tree.

SUMMARY AND RECOMMENDATIONS FOR THE ALGORITHMS

The algorithms DUFFNO and STEM were both written in 1977 ANSIFOR standard Fortran, for use on a Digital Equipment Corporation VAX 11/780 computer using the VMS operating system. Neither program uses any external subroutines which might be unique to a given system. Therefore it should be possible to use the algorithms on most other systems equipped with a Fortran compiler.

The programs were written to be used together. DUFFNO's greatest contribution is the data check for errors which eliminates the need for costly erroneous runs of STEM. However, DUFFNO is not designed to find operator errors. It is the user's responsibility to ensure that all data are arranged in the correct sequence.

DUFFNO will tolerate information for trees with rotten centres, and produce the Duff-Nolan sequences for all the discs. However, the resultant information should be considered suspect. STEM is incapable of dealing with these trees.

Although STEM and DUFFNO were written to accomodate data produced by the Holman Digimicrometer, data from other sources, whether mechanical or otherwise, could easily be adapted for use in these algorithms. Future considerations for the algorithms could include the development of a subroutine to output data from other sources in the format required by DUFFNO and STEM.

Presently, the height/age data produced by STEM is not corrected for bias. Another subroutine could be developed to correct height data, according to the methods of Carmean (1972) and Lenhart (1972).

At this point in time, STEM is incapable of producing a height/age curve and taper curves for individual trees. This is due to the lack of suitable plotting facilities on the VAX 11/780 computer at Lakehead University. When a graphics package becomes available to the system, a subroutine could easily be adapted to the algorithm, to produce the graphs. The main advantage of not having this capability, is the relative ease in which the algorithm may be transferred to another system. The use of special incremental plotting facilities could complicate such a transfer.

METHODOLOGY FOR DEVELOPING STEM ANALYSIS TECHNIQUES

This section considers the question of determining how many discs per tree are required, in order to obtain an accurate estimate of the true total volume. It should be possible to estimate the volume for the present, or past time periods, within ± 10 percent of the true volume mean, at a 95 percent confidence level.

THE STUDY AREAS

This study is limited to white spruce within the Thunder Bay District of the Ontario Ministry of Natural Resources. Three sample trees were taken from each of the three areas. The Hogarth Plantation, a privately owned woodlot, (89 degrees, 22 minutes west longitude, 48 degrees, 21 minutes north latitude) is located near the Thunder Bay Forest Station (Figure 1, page 44). The second area is a

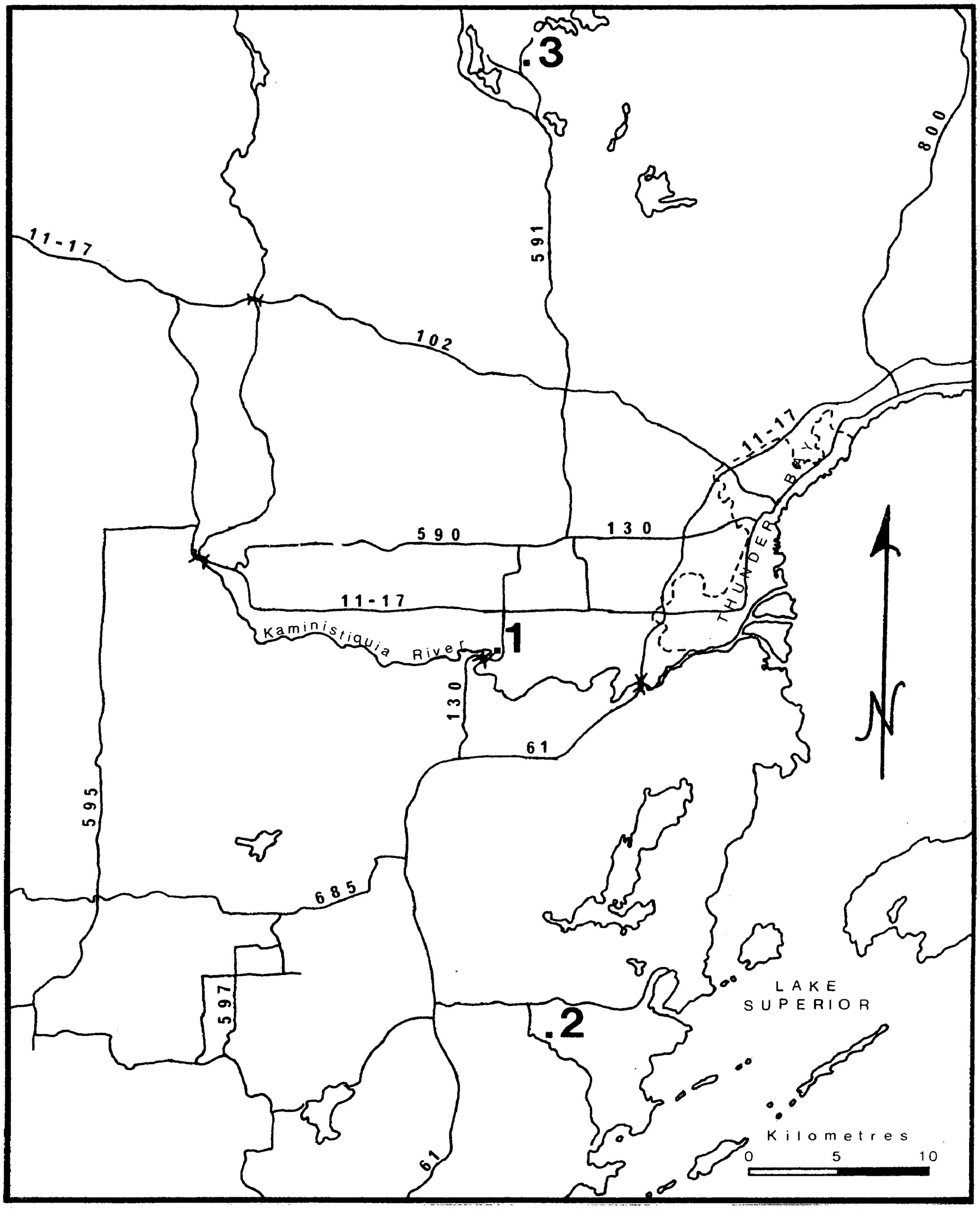


FIGURE 1 Sample areas. 1 - Hogarth Plantation; 2 - Prince and Jarvis; 3 - Lakehead University Woodlot.

crown land plantation in the Prince and Jarvis area located approximately 45 kilometres south of Thunder Bay (89 degrees, 23 minutes west longitude, 48 degrees, 10 minutes north latitude). The remaining trees were obtained from the Lakehead University Woodlot, Jacques Township, about 29 kilometres northwest of Thunder Bay (89 degrees, 22 minutes west longitude, 48 degrees, 38 minutes north latitude).

SAMPLING METHODOLOGY

Sample trees were selected on the basis of reasonable form with no multiple tops. Trees with obvious deformities or growth interruptions due to leader losses were avoided. Access to each area to enable the easy removal of discs was also considered. Trees could be selected from any crown class. It was assumed that there would be no difference in the affect of the stem analysis sample on true volume for dominants, codominants or suppressed trees. The trees were not chosen randomly, thus for the purpose of analysis of variance, the experiment can be blocked on trees.

Intervals of 10 cm for section lengths were used to obtain the true volume. Only a volumetric measurement could provide a more accurate estimate of true total volume. A closer interval, such as 5 cm, would have necessitated extracting the entire tree, assuming a disc thickness of approximately 5 cm. With a 10 cm interval between discs, only half the tree was transported back to the laboratory.

Before felling a tree, the first 50 cm of height were marked at 10 cm intervals, starting as close as possible to the ground. The location of the ground level disc depended on chain saw cutting safety. For trees growing on a slope, the lowest possible mark was on the high side of the tree. The intervals were marked with a carpenter's saw. The felling cut was made between the ground mark at 0.0 m and the 0.1 m mark. Special care was taken to preserve the discs at both marks. The ground disc was taken after felling, and labelled with a waterproof felt marker, on the reverse of the side to be measured. The disc was labelled with the tree number, and a disc number of zero, corresponding to the height of o.o metres. Disc numbers for the remaining discs also corresponded to disc height. For example, the disc at 1.3 metres became disc # 13.

After felling, the tree was limbed and total height was measured. Then the remainder of the tree was marked at 10 cm intervals, using a carpenter's saw. The intervals for the last 3 metres of the tree, were marked with a felt marker, up to the point were only one year's growth existed. The last 3 metres were left intact and transported to the laboratory, for sectioning on the band saw. This ensured that none of the smaller discs were misplaced in the field. The band saw produced a smoother surface on the smaller discs, which was impossible to achieve with a chain saw.

After the bole was marked for cutting, discs were taken in sequence, starting with the disc closest to the base of the bole. Discs were labelled with the tree number and disc number, as they were being cut, and then stacked in order on the ground, until the cutting was complete. The 3 metre tip was also labelled with the tree number, to avoid confusion in the lab. The stacks of discs were carried out and bagged at roadside before transporting. At the laboratory, discs were stored outside in a frozen state, prior to measurement.

An average of 2 hours were required for a 2-man crew to fell, mark, cut, label and bag the discs for an average tree of 14 metres in height. A total of 36 man-hours of fieldwork were required to obtain the discs for the 9 sample trees.

DIGIMICROMETER MEASUREMENTS

The discs had to be specially prepared before they could be measured on the Holman Digimicrometer. The minimum preparation, required for all discs, was to calculate, locate and mark an average radius. The minimum and maximum radius from pith to inside-bark was measured with a set of dividers and a scale. The arithmetic average was marked on the disc with an ebony pencil. At times it was necessary to chose an alternate radius, as close as possible to the average radius, when the average radii crossed over rot, pitch pockets or aberrant annual ring configurations. Discs located close to the nodes caused most of the problems.

Smoothly surfaced, wide-ringed discs required no further preparation. Narrow-ringed discs with latewood pushed over the springwood by the chain saw generally required extra preparation with a knife or razor blade. The disturbed surface along the average radius, was re-cut to eliminate the disturbed wood. Ring borders are difficult to distinguish on roughly surfaced discs, especially under a microscope. A total of 61 man-hours were required to prepare all the discs for the sample trees. The slower growth trees from the Woodlot required more preparation time, than those from the Hogarth Plantation.

Discs for each tree were measured in sequential order, from the ground level disc up, on the Holman Digimicrometer. Trees were also measured sequentially, to minimize the amount of editing required to produce the final PLOT.DAT file for each plot.

The individual disc header included the tree number, species code, disc number, a code to identify whether the disc occurred at a node, the disc height, and the year cut. The ring count was added to the header during editing. The total average radius and single bark thickness were included by the digimicrometer before the data were transferred to cassette tape.

To measure a disc, the digimicrometer moves the disc from right to left, through the microscope's field of vision, starting at the outside-bark, and moving inward to the pith. The operator controls the speed of the disc moving through the viewing field, and presses the reset button when the crosshairs in the microscope eye piece are at the edge of an annual ring. The ring width measurements are stored in the microprocessor memory until measuring is complete. Then the machine is programmed to transfer the ring width data and disc header for the disc, to cassette tape. At the end of a measuring session, the data are transferred to a file in a computer.

During editing, incorrect disc records are eliminated, and later replaced with a correct remeasurement of the same disc. Discs with missing data were remeasured, unless only one ring width was missing. DUFFNO was used to aid in calculating the missing value.

The total measuring time required for the 9 trees, with an average of 14Ø discs each, was 73.5 hours. Editing time required for the full data set was another 1Ø hours.

ANALYSIS AND RESULTS

All discs for each of the 9 trees were measured and stored in three seperate PLOT.DAT files, one for each of the areas. DUFFNO was then used to produce the Duff-Nolan sequences, and to detect errors in the data. The data were then submitted to STEM to calculate the stem analysis results. The volume estimates based on the 10 cm intervals, were used as the true volume (control volume) for the trees, in any subsequent statistical analysis.

Subsets of disc data used for the analysis, were created on the VAX 11/780, using available editing software.

THE STATISTICAL TREATMENTS

A practical approach, was to try a range of methods, generally used in the field, to determine whether any of the commonly used methods are adequate. For this study, 26 different treatments were tested.

1 - The control (true volume); 10 cm intervals starting at ground level, to the first disc with only one year's growth.

2 - Section lengths of 0.5 m; with an additional disc at dbh and at ground level.

3 - Section lengths of 0.5 m; with an additional disc

at ground level only.

4 - Section lengths of 1.Ø m; with an additional disc at dbh and at ground level.

5 - Section lengths of 1.Ø m; with an additional disc at ground level only.

6 - Section lengths of 2.Ø m; with an additional disc at dbh and at ground level.

7 - Section lengths of 2.Ø m; with an additional disc at ground level only.

8 - Section lengths of 4.Ø m; with an additional disc at dbh and at ground level.

9 - Section lengths of 4.Ø m; with an additional disc at ground level only.

1Ø - Discs taken as dictated by the Girard form class (Avery 1967); at dbh and at 5.3 m, with an additional disc at ground level.

11 - Discs taken as dictated by the Girard form class; at dbh and at 5.3 m.

12 - Discs taken as dictated by the absolute form quotient (Avery 1967); at dbh and at half the height above dbh, including the ground level disc.

13 - Discs taken as dictated by the absolute form quotient; at dbh and at half the height above dbh.

X 14 - Discs taken as dictated by the normal form quotient (Avery 1967); at dbh and at half the total height, including the ground level disc.

15 - Discs taken as dictated by the normal form quotient; at dbh and at half the total height.

16 - Discs taken at ground level, Ø.2 metres, dbh and at one third the total height of the tree.

17 - Discs taken at ground level, dbh and at one third the total height of the tree.

18 - Discs taken at dbh and at one third the total

height of the tree.

19 - Discs taken at ground level and at dbh.

2Ø - Only the dbh disc.

21 - Only the ground level disc.

22 - Romberg's method (Goulding 1971) for 1 section using the first and last discs sampled.

23 - Romberg's method for 2 equal sections, using the first and last and disc sampled, with one at half the distance of the bole.

24 - Romberg's method for 4 equal sections, using the first and last discs sampled, with 3 others at equal distances.

25 - Romberg's method for 8 equal sections, using the first and last discs sampled, with 7 others at equal distance.

26 - Romberg's method for 16 equal sections, using the

first and last discs sampled, with 15 others at equal distance.

For all treatments, the disc closest to the actual height required was used. Because the full sample used intervals of 10 cm, a sub-sample disc could miss the actual height required, by up to 5 cm in either direction. This discrepancy has been ignored. It is assumed that the original section intervals were initially measured correctly. Any resultant error is considered insignificant.

The PLOT.DAT and TREE.DAT file for each of the 26 treatments, were created by editing the PLOT.DAT and TREE.DAT file for the full set of discs for each tree. An extra data file was also created to save the volume estimates for past and present, for each tree, based on the treatments. These volume files were saved for future analysis.

STATISTICAL METHODS

Rather than comparing only the present volume for each tree, it seemed essential to also compare volumes for past time periods. Therefore, for all trees, analyses were carried out on the volume estimates of the present, and for five, ten and fifteen years ago.

Analysis of variance was used to test for differences between the volume estimates based on the 26 treatments, including the control. The experiment was blocked on trees, because they were not chosen randomly, and were known to be from different stands. Therefore any interaction between treatments and blocks has been included in the sum of squares for the error.

Analysis of variance was carried out on four sets of volume data. The first test was performed on the estimates for the present volume of the trees. Table 5 (page 57) describes the Analysis of Variance (ANOVA) table for the experiment. The F value for the blocks (trees) is significant but that is not a valid test due to lack of randomness. The F value for the treatments is significant at the 95 percent level. Therefore, there is a significant difference between the volume estimates produced by each

TABLE 5 ANOVA for the present volume estimates.

ANOVA

Treatment	Sum of Squares	Df	Mean Square	F
Block (Trees)	2681670.3248	8	335208.7906	578.4
Treatment	313240.2131	25	12529.6085	21.6*
Error	115902.3206	200	579.5116	
Total	3110812.8585	233		

* significant at the 95 % level.

TABLE 6 ANOVA for the volume estimates of 5 years ago.

ANOVA

Treatment	Sum of Squares	Df	Mean Square	F
Block (Trees)	1555641.2837	8	194455.1605	918.5
Treatment	151392.2753	25	6055.6910	28.6*
Error	42340.1911	200	211.7009	
Total	1749373.7501	233		

* significant at the 95 % level.

treatment. Tests for five, ten, and fifteen years ago are shown in Tables 6, 7 and 8 (pages 57, 59). Note that they indicated a significant difference between the volume estimates produced by each treatment. The volume estimates produced by the various treatments show consistent results in the ANOVA tables over time. Therefore the procedure appears consistent.

Analysis of variance, used to test the four sets of data, indicates that there are significant differences between treatments. Analysis of variance for orthogonal experiments is capable of determining which treatments are significantly different from the control treatment, the true volume. Due to the size of the experiment, specifically the number of treatments, it was not feasible to analyze the data with orthogonal treatment contrasts. Therefore alternative methods were chosen to determine which treatments could predict the true volumes, within an allowable error of $\pm$ 10 percent of the control mean volume, at a 95 percent level of confidence. Several tests are available to compare pairs of means. In this instance, one of the means in a given pair, will always be the control mean.

TABLE 7 ANOVA for the volume estimates of 10 years ago.

ANOVA

Treatment	Sum of Squares	Df	Mean Square	F
Block (Trees)	746178.5976	8	93272.3247	1194.8
Treatment	59931.5283	25	2397.2611	30.7*
Error	15612.8213	200	78.0641	
Total	821722.9472	233		

* significant at the 95 % level.

TABLE 8 ANOVA for the volume estimates of 15 years ago.

ANOVA

Treatment	Sum of Squares	Df	Mean Square	F
Block (Trees)	372401.2183	8	46550.1523	1341.0
Treatment	18681.2014	25	747.2481	21.5*
Error	6942.6417	200	34.7132	
Total	398025.0614	233		

* significant at the 95 % level.

Tukey's test or the Honestly Significant Difference (HSD) procedure is used for experiments where many comparisons are to be made (Steel and Torrie 1960). The method is not limited to prechosen comparisons, therefore it may be used as a multiple range test. It may also be used to compare a control mean against all other means. Tukey's test uses a single value for judging observed differences. The number of experiments being tested is the unit used for stating the significance level.

Dunnett's test was specifically designed to locate treatments which are different from the standard or control mean (Steel and Torrie 1960). The procedure requires a single difference for judging the significance of the observed differences. The error rate of the Type I error, is on an experiment-wise basis, rather than on a comparison-wise basis.

A third test, is Bonferonni's Significant Difference, which also controls the experiment-wise error, like Tukey's and Dunnett's tests. For any single test, the Type I error rate is equal to αi, when αi is equal to α/n, n being the number of means being compared. The probability of at least one Type I error can be no greater than α (Hinkley 1978). Bonferroni's significant difference is a single value based on the number of treatments being tested, and the error degrees of freedom from the ANOVA table (Snedecor and Cochrane 1980). In this case the test comparisons are prechosen, or _a priori_. For Tukey's test, the comparisons may be _a priori_, as in the case of comparing the control mean against all others. When Tukey's test is used as a multiple range test, the tests are not prechosen.

Of the three tests being used, Bonferroni's is the most powerful test (Weisburg 1980). This procedure requires very large differences to occur, before it finds a significant difference. Dunnett's test is the least powerful test.

Table 9 (page 62) presents the results for Dunnett's two-tailed test, Tukey's test and Bonferonni's test for the present volume means. In each test, the control treatment (1), is compared to all the other means. The means are listed in ranking order, from lowest to highest value. In this instance the control mean is the lowest mean.

Dunnett's test groups the first 16 means. Means within this group are not significantly different from one another. Tukey's test and Bonferonni's test, both include the next 3 means. Interestingly enough, Dunnett's test only includes those means which fall within 10 percent of the control

TABLE 9 Dunnett's, Tukey's and Bonfferoni's tests for the present volume means.

Mean Volume (dm3)	Dunnett's	Tukey's	Bonfferoni's
180.970 (1)	│	│	│
181.950 (2)	│	│	│
181.951 (3)	│	│	│
183.643 (13)	│	│	│
184.270 (15)	│	│	│
185.167 (26)	│	│	│
186.772 (4)	│	│	│
187.091 (5)	│	│	│
189.392 (6)	│	│	│
189.735 (11)	│	│	│
191.052 (25)	│	│	│
191.057 (8)	│	│	│
191.391 (18)	│	│	│
195.519 (7)	│	│	│
197.581 (12)	│	│	│
198.207 (14)	│	│	│
199.979 (16)		│	│
201.018 (20)		│	│
203.672 (10)		│	│
205.329 (17)			│
208.675 (24)			│
214.955 (19)			│
215.128 (9)			│
247.407 (23)			
318.852 (22)			
327.484 (21)			

The │ line represents homogenous groups of means defined by the different tests, using pairwise comparisons with the control mean (Treatment 1) as one of the pair of means. Treatment number is given in parenthesis.

mean. The 3 means included by Tukey's and Bonferonni's tests, occur outside the allowable error level previously chosen. Figure 2 (page 64) gives a graphical representation of the treatment means, as differences from the control mean. The standard deviations from the treatment means, are also shown. The graph readily emphasizes those treatments which are obviously not suitable for estimating true volume. Most of the treatments overestimate the control volume. This is probably due to the parabolic form assumed by Smalian's formula, which was used to calculate the volume estimates for each treatment.

Table 10 (page 65) provides the results of the three pairwise tests for the volume means of five years ago. In this case Dunnett's and Tukey's tests agree that the first 17 of the ranked means are not significantly different from one another. Unlike the previous case, the control mean is not the lowest ranking treatment mean. Treatment 15 has moved from fifth place (Table 9, page 62), to the lowest ranking mean. Bonferonni's test has included 4 more means in the homogeneous group. However, as in the last instance these 4 means are beyond the $\pm$ 10 percent allowable error. Dunnett's and Tukey's test have chosen means within the $\pm$ 10 percent allowable error. Figure 3 (page 66) graphically

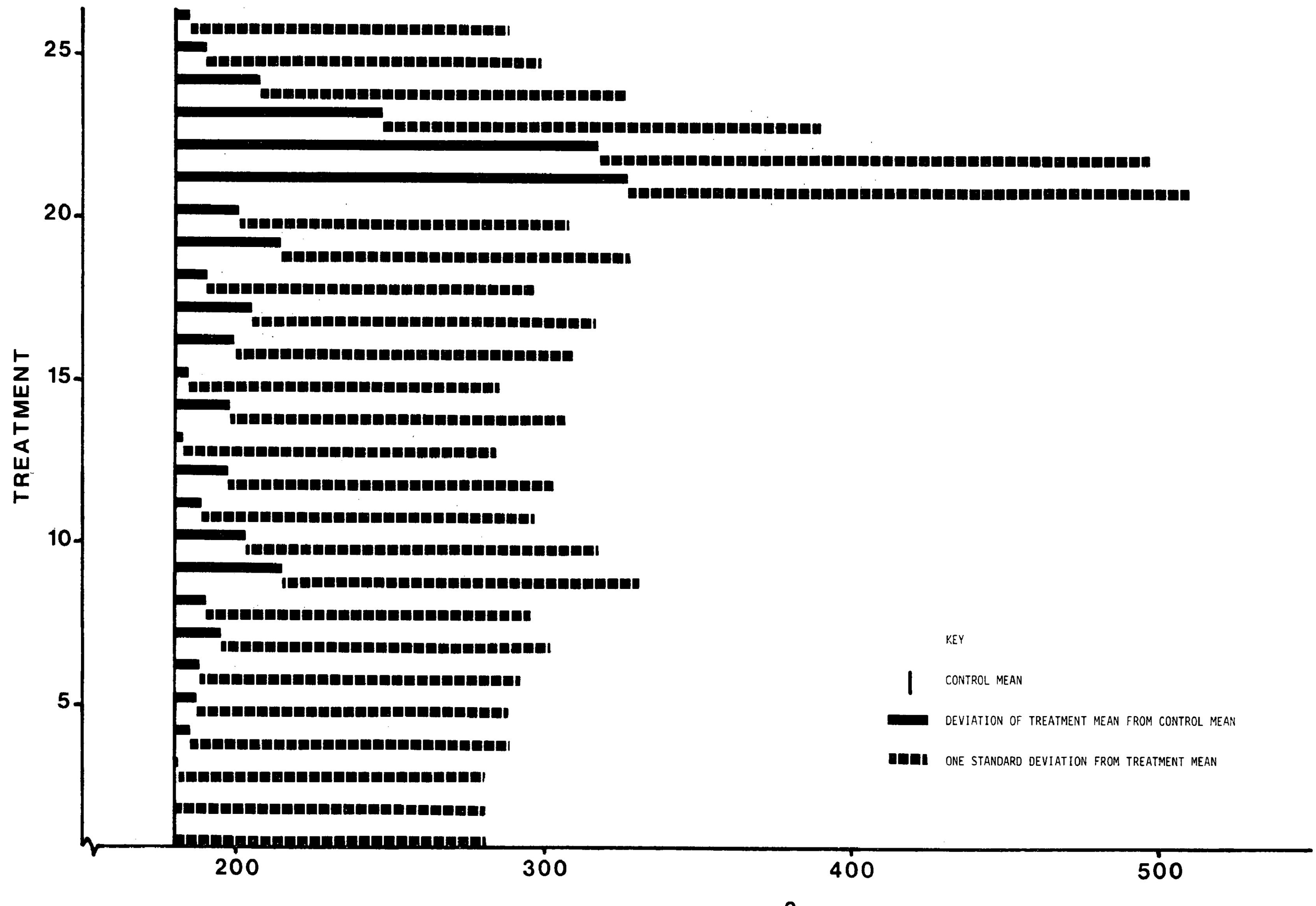


Figure 2 Individual treatment means and standard deviations for present volume estimates.

TABLE 10 Dunnett's, Tukey's, and Bonfferoni's tests for the volume means of five years ago.

Mean Volume (dm3)	Dunnett's	Tukey's	Bonfferoni's
123.937 (15)	|	|	|
124.401 (1)	|	|	|
124.792 (13)	|	|	|
125.025 (3)	|	|	|
125.077 (2)	|	|	|
128.468 (26)	|	|	|
128.595 (4)	|	|	|
128.744 (5)	|	|	|
128.784 (18)	|	|	|
129.756 (11)	|	|	|
131.116 (6)	|	|	|
133.318 (25)	|	|	|
134.240 (8)	|	|	|
134.328 (14)	|	|	|
135.184 (12)	|	|	|
135.288 (7)	|	|	|
135.664 (16)	|	|	|
139.175 (17)			|
139.434 (20)			|
140.147 (10)			|
145.550 (24)			|
149.825 (19)			
151.092 (9)			
171.458 (23)			
222.250 (21)			
222.778 (22)			

The | line represents homogenous groups of means defined by the different tests, using pairwise comparisons with the control mean (Treatment 1) as one of the pair of means. Treatment number is given in parenthesis.

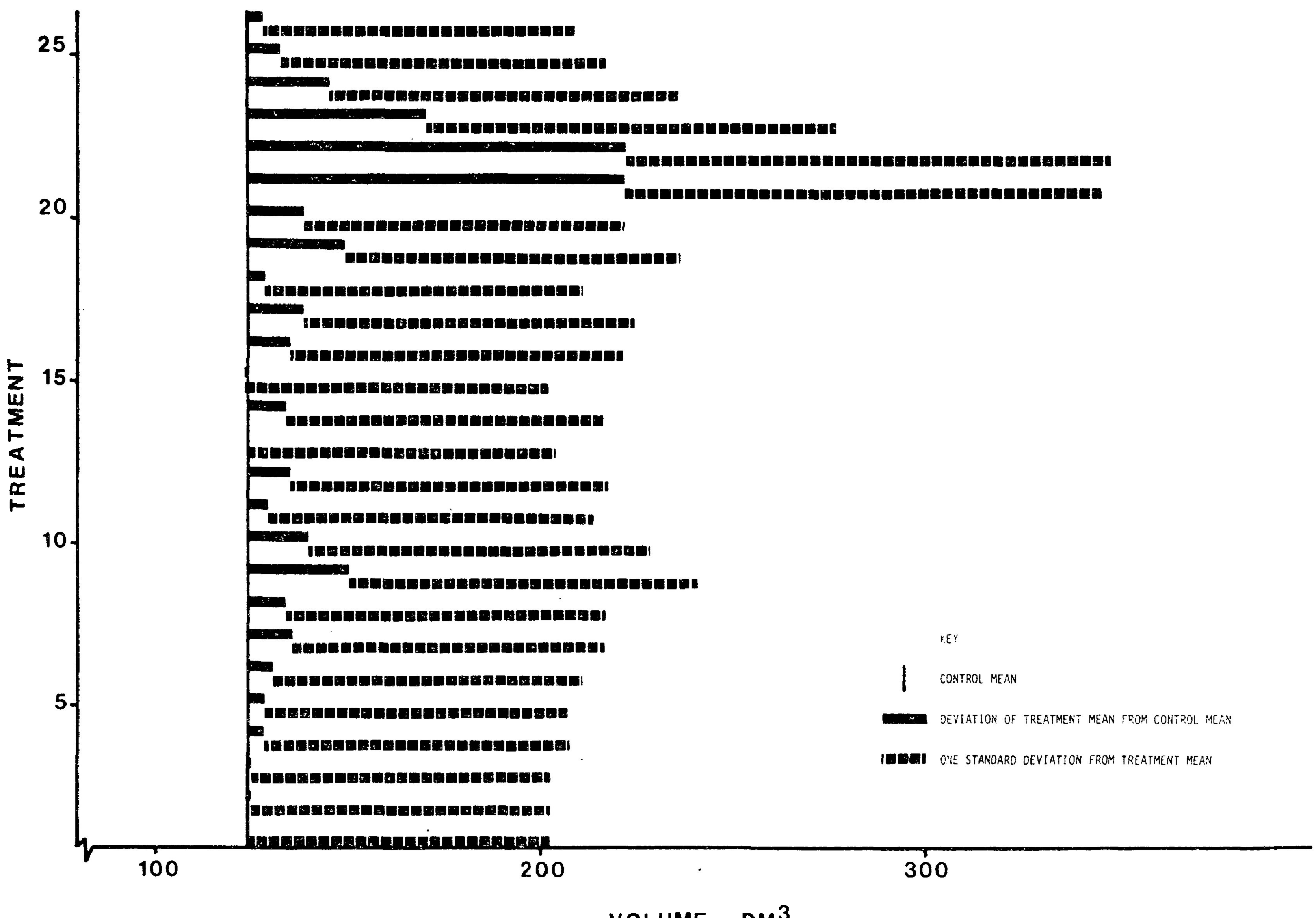


Figure 3 Individual treatment means and standard deviations for volume estimates of 5 years ago.

represents the treatment means and standard deviations for the volume estimates of five years ago.

The volume means of 10 and 15 years ago were also tested with the three pairwise comparison tests. Table 11 (page 68) shows the results for volume means of 10 years ago. As before, the control is not the lowest ranking mean. Dunnett's test groups the first 16 means. This time it misses treatment 12, which is still within ± 10 percent of the control mean. Tukey's test groups the first 18 means, and includes one mean which is beyond the ± 10 percent allowable error. Bonferonni's test groups the first 20 treatments, again exceeding the ± 10 percent allowable error. Figure 4 (page 69) gives the graphical representation of treatment means and standard deviations for volumes of 10 years ago.

For the volume means of 15 years ago (Table 12 page 70), the results are relatively similar. Dunnett's test groups the first 17 means, all within ± 10 percent of the control mean. The control mean has now dropped to fourth place. Tukey's test includes the first 19 means in a homogeneous group. The last two means included, exceed the ± 10 percent allowable error. Bonferonni's test groups the first 21 means, which includes 4 means that exceed the ± 10

TABLE 11 Dunnett's, Tukey's and Bonfferoni's tests for the volume means of ten years ago.

Mean Volume (dm3)	Dunnett's	Tukey's	Bonfferoni's
78.230 (15)	│	│	│
79.740 (1)	│	│	│
80.099 (13)	│	│	│
80.161 (3)	│	│	│
80.195 (2)	│	│	│
80.341 (18)	│	│	│
81.452 (11)	│	│	│
81.911 (4)	│	│	│
82.086 (5)	│	│	│
82.164 (26)	│	│	│
83.509 (6)	│	│	│
85.643 (25)	│	│	│
85.649 (14)	│	│	│
85.841 (16)	│	│	│
86.147 (7)	│	│	│
86.397 (8)	│	│	│
87.518 (12)		│	│
87.761 (17)		│	│
88.872 (10)			│
88.913 (20)			│
93.827 (24)			
96.333 (19)			
97.696 (9)			
108.890 (23)			
140.649 (21)			
140.972 (22)			

The │ line represents homogenous groups of means defined by the different tests, using pairwise comparisons with the control mean (Treatment 1) as one of the pair of means. Treatment number is given in parenthesis.

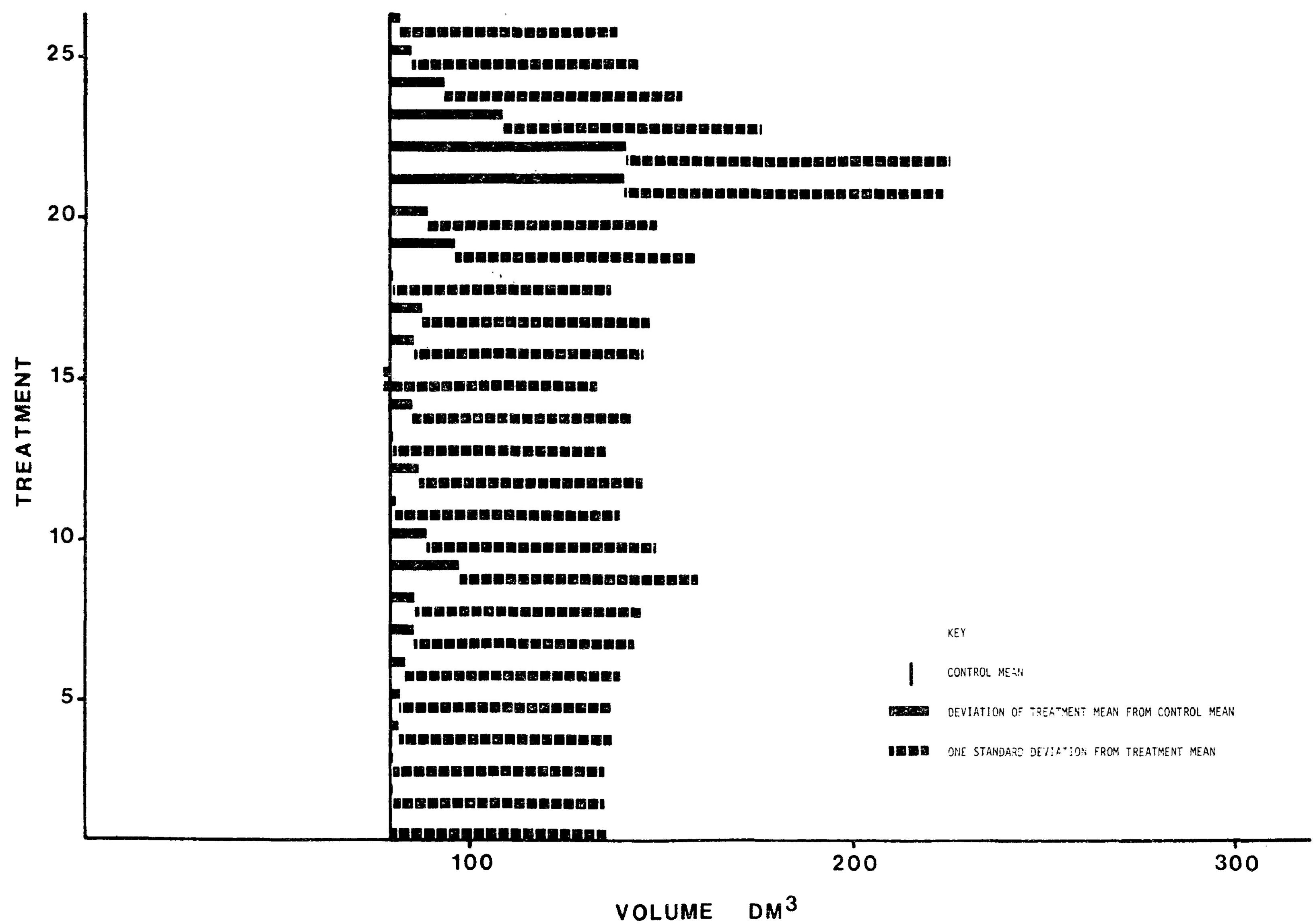


Figure 4 Individual treatment means and standard deviations for volume estimates of 10 years ago

TABLE 12 Dunnett's, Tukey's and Bonfferoni's tests for the volume means of fifteen years ago.

Mean Volume (dm3)	Dunnett's	Tukey's	Bonfferoni's
46.338 (18)	│	│	│
47.836 (11)	│	│	│
48.105 (3)	│	│	│
48.142 (1)	│	│	│
48.162 (2)	│	│	│
48.474 (15)	│	│	│
48.689 (4)	│	│	│
48.803 (5)	│	│	│
49.590 (26)	│	│	│
49.843 (6)	│	│	│
50.275 (16)	│	│	│
50.292 (13)	│	│	│
50.974 (17)	│	│	│
51.312 (7)	│	│	│
51.425 (15)	│	│	│
51.536 (8)	│	│	│
52.471 (10)	│	│	│
53.010 (20)		│	│
53.110 (14)		│	│
54.928 (12)			│
55.909 (24)			│
57.646 (19)			
57.736 (9)			
65.538 (23)			
81.562 (21)			
81.713 (22)			

The | line represents homogenous groups of means defined by the different tests, using pairwise comparisons with the control mean (Treatment 1) as one of the pair of means. Treatment number is given in parenthesis.

percent allowable error. Treatment means and standard deviations for volumes of 15 years ago, are shown graphically in Figure 5 (page 72).

Common sense suggests that the treatments, which hold interest for us are those which are consistently within 1Ø percent of the control mean and which are consistently grouped by Dunnett's test. For the four sets of data, treatments 2, 3, 4, 5, 6, 7, 8, 11, 13, 15, 18, 25, and 26 are consistently in the homogeneous group, based on the control mean. The results suggest several options for conducting stem analysis, with a potential for saving valuable time and money for researchers. The first method of sectioning tested, was that of "uniform section lengths". Section lengths of Ø.5, 1.Ø, 2.Ø and 4.Ø metres were tested. The disc at dbh was included for four of the eight treatments. According to the groupings of all three tests, any of the tested uniform lengths estimate the volume, to within 1Ø percent of the control or "true" volume. The only combination which failed to estimate the volume reasonably, was treatment 9, which used 4.Ø metre sections without the dbh disc.

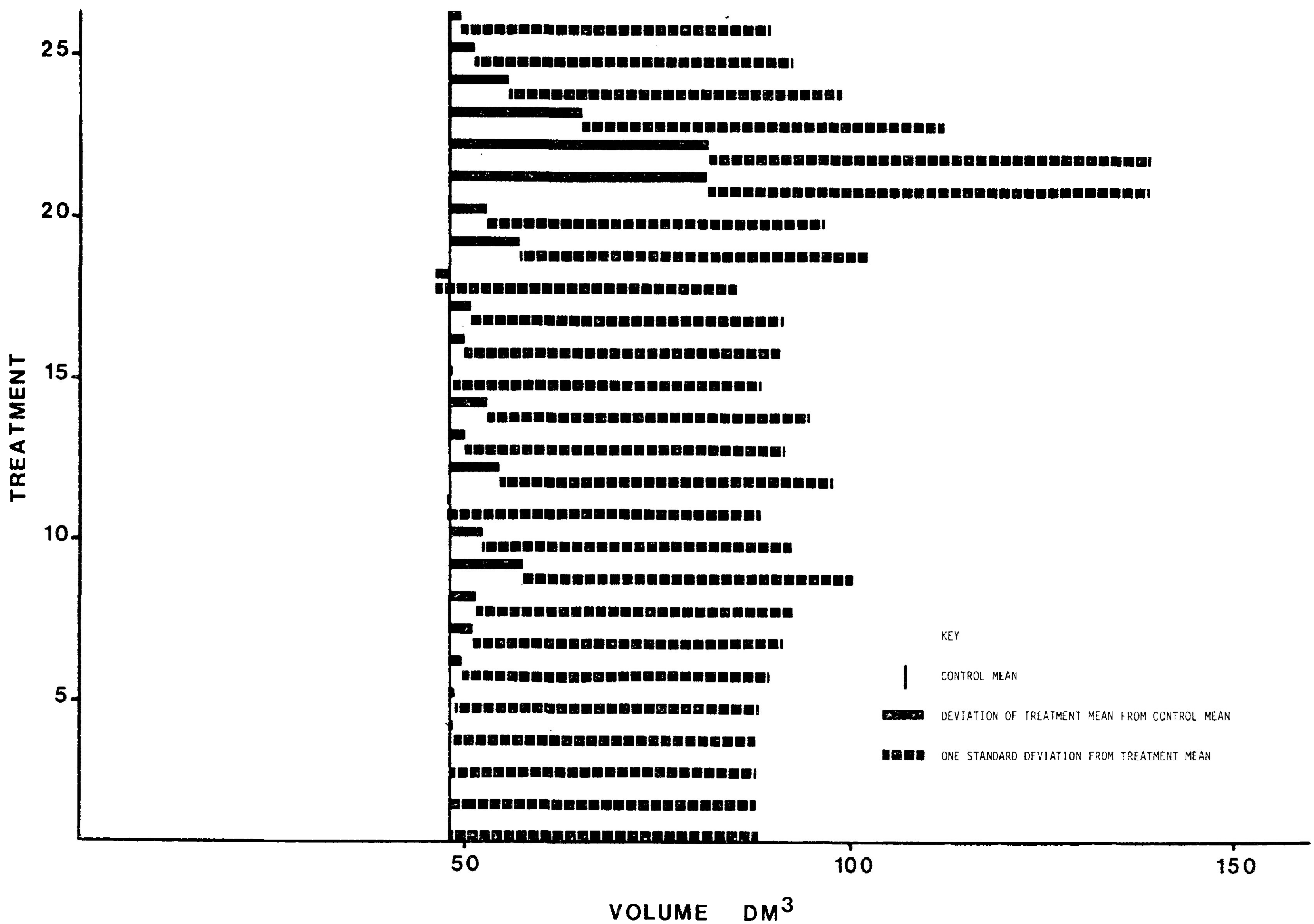


Figure 5 Individual treatment means and standard deviations for volume estimates of 15 years ago.

A second method of selecting discs, were the methods based on form quotients. Treatments 11, 13, 15, and 18 were consistently grouped with the control mean, by all the tests. Treatment 11 is based on Girard's form class, and used only the discs at dbh and at 5.3 metres. Treatment 13 is based on the absolute form quotient. Again only two discs were sampled, as is the case for treatment 15, based on the normal form quotient. Treatment 18 was based on the dbh disc and the disc at one third of the total height of the tree.

The third method of selecting discs, was based on Romberg's method (Goulding 1971). The three tests consistently placed treatments 25 and 26 in the homogeneous group, based on the control mean. Treatments 25 and 26, were based on 8 and 16 equal section lengths, respectively.

An alternative method for comparing treatment means has been suggested by Freese (1960). His example involves two comparisons of treatments against a control. For this project it was necessary to make 25 different pair-wise comparisons. Therefore it was necessary to control experiment-wise error simultaneously with the comparison-wise error.

DISCUSSION

Bonferonni's Significant Difference test, Tukey's test and Dunnett's test, were used to evaluate the control mean for the four sets of volume data against all the other treatment means. For the four sets of volume data, Bonferonni's test consistently groups more means together as a homogeneous group.

The experiments were to be carried out at the 95 percent level of confidence and the mean volume estimates had to be within ± 1Ø percent of the control volume mean. Bonferonni's test consistently groups means into the homogeneous group which exceed the ± 1Ø percent allowable error limit, at the 95 percent level of confidence. Conversely, Dunnett's test consistently groups treatment means within the ± 1Ø percent allowable error. Considering the two restrictions, Dunnett's test appears to be the most reliable test. Tukey's test is less consistent than Dunnett's test, as it often includes treatment means outside the ± 1Ø percent allowable error limit.

Dunnett's test consistently included most of the "uniform section length" treatments with the control mean. The trials with section lengths of Ø.5, 1.Ø, 2.Ø and 4.Ø metres with the dbh disc, worked well for the present and past volumes. Of the trials without the dbh disc, only the 4.0 metre section treatment fell outside the group delineated by Dunnett's test. Therefore, it appears that researchers who wish to use stem analysis for estimating past and present volumes, can use the "uniform section length" method, with good success. However, it is important to realize that the experiments are only representative of a limited sample of nine white spruce between the ages of 30 and 80 years, in the Thunder Bay District.

The experiments do not indicate any relationship between section lengths and height growth. No testing was done to compare the control height growth information against the treatment height growth estimates. The height/age curves for the nine white spruce were not corrected for bias in height measurement. The control treatment sections were 10 cm long, therefore, the actual height at the top of each section could only be underestimated by a maximum of 5 cm. Future studies of this nature could investigate the effects of stem analysis sampling techniques, on the reliability of height growth information.

The "form class" method of choosing discs for measurement, also appears to be useful to researchers. Dunnett's test groups the 2 - disc "form class" methods consistently with the control mean. Apparently, the volume estimates for the past or present, based on the dbh disc, and one other disc dictated by the form quotient, are within 1Ø percent of the control mean, 95 percent of the time. Therefore, researchers who are only interested in volume estimates for the past and present, may be able to obtain the desired information from a minimum of 2 discs. The "form class" method would also seem unreliable as far as height growth is concerned. Again, the reliability of height growth estimates were not investigated.

Romberg's method of selecting discs for stem analysis gave some interesting results. Dunnett's test consistently included Romberg's 8 - section, and 16 - section treatments with the control mean. Therefore, for the tested data set, this method requires a minimum of 9 discs (8 sections), to give an estimate of volume within ± 1Ø percent of the true volume. If this remains true for very tall trees, such as many of the western tree species, then it may be possible to use a small number of discs to obtain the desired volume information. Because the section lengths are equal and more numerous, information concerning height growth should also be better, than that obtained with the "form class" methods. For shorter trees, such as the white spruce tested, this method does not seem to be more advantageous than the "uniform section length" method. In fact, for 14 metre trees using the 4.0 m length sections, including a disc at dbh, only 5 discs are required compared to the nine discs required for Romberg's method.

SUMMARY AND RECOMMENDATIONS FOR SAMPLING TECHNIQUES

The experiments have indicated that there are three basic sampling methods which can be used to select discs for obtaining volume estimates. These could be used to reduce sampling costs for stem analysis.

The "uniform section length" method, that is, lengths of 0.5, 1.0, 2.0 and 4.0 metres, works well for shorter trees. The experiments were based on trees ranging in height from 10 to 14 metres. A minimum of 5 discs were required for an accurate estimation of the true volume under the two experimental restrictions. This method may provide reliable height growth information, although this was not investigated. It is important to recognize that an old, slow-growing tree may require more sections to describe volume and height growth, than a young, fast-growing tree of the same height.

The "form class" method only required 2 discs to estimate the true volume within 10 percent of the volume, at a 95 percent level of confidence. However, it is unlikely that the height growth information derived from this method, would be reliable enough for research leading to the computation of site index curves.

Romberg's method also gives favourable results for volume estimates. The advantage of this method may become noticeable for taller west coast species. This method, like the "uniform section length" method, should give reasonable height growth information. The sample trees for this study were all white spruce, of similar height and form class. Therefore further testing is required to evaluate the three sampling methods for other tree species, other height classes, different form classes, and various sites across Canada. Expanded research could also include testing of the height growth information provided by the various sampling techniques. This is especially beneficial to site index research.

The effects of tree form class on the number of discs required to sample a tree, may also be an important area requiring further research. Generally, trees growing in the same stand tend to have similar form. It is possible that a detailed sampling of a few trees could be used to obtain volume estimates of other sample trees if the sheath volumes put on, are similar. For example, if trees of different height and dbh classes have the same form, could the sheath volume information from the larger tree be used to aid in predicting the volume of the smaller tree? Research in this area would be interesting and useful.

SUMMARY OF RESEARCH

The thesis has investigated two main areas of concern; stem analysis data processing, and stem analysis sampling techniques.

The first part of the thesis demonstrated the need for a new computer algorithm capable of processing data produced by mechanical and traditional methods. The development of programs DUFFNO and STEM was described, and directions for their use was given.

Future considerations for DUFFNO and STEM could include any or all of the following.

1 - A matrix-generator subroutine, which would accept data from sources other than the Holman Digimicrometer, and output the information in the format required for DUFFNO and STEM.

2 - A subroutine to correct height underestimates caused by assuming that the height at the point of sectioning represents actual height.

3 - A subroutine to plot height/age and taper curves for individual trees.

This thesis has contributed to improving stem analysis data processing, by providing two algorithms capable of handling stem analysis data from any source. When DUFFNO and STEM are used properly, the user can save valuable computer time and processing costs.

The second part of the thesis investigated several stem disc sampling methods for estimating true volume. A total of 26 methods including the precise volumes, were compared.

The treatments that consistently grouped with the control treatment, were considered to give reliable estimates of the true volume. These sampling methods can be used to obtain reliable volume estimates, while saving sampling time and money, compared to the cost and effort involved in sampling the control treatment. The three basic sampling methods developed included, the "uniform section length" method, the "form class" method, and Romberg's method. The "form class" method required the least number of discs, but yielded the most unreliable height growth information.

This thesis has also contributed to improving stem analysis sampling techniques by providing several alternatives for selecting sample discs. A reduction in the number of discs selected per tree, leads to reduced field time and costs. Associated laboratory measuring time and costs are also reduced.

Future research into the sampling techniques should include the effect of other tree species, other height classes, different form classes, and site quality. The reliability of the resultant height growth information should also be investigated.

REFERENCES

Avery, T. Eugene. 1967. Forest measurements. McGraw-Hill Book Co. Inc., New York. 290 pp.

Beek, J. van der, and P.P.Th.M. Maessen. 1981. The "Dorschkamp" equipment for measuring width of annual growth rings. Nederlands Bosbouw Tijdschrift 53(6):158-164.

Behman, Philip James. 1982. Indices of branch contribution to bole growth in *Pinus resinosa* Ait. Master of Science Thesis, School of Graduate Studies, Lakehead University, Thunder Bay, Ontario, 148 pp.

Bell, J.F., D.D. Marshall and G.P. Johnson. 1981. Tarif tables for mountain hemlock developed from an equation of total stem cubic-foot volume. For. Res. Lab. Oregon State Univ., Corvallis, Res. Bull. # 35, 46 pp.

Bowen, Murray G. 1974. Selected metric (SI) units and conversion factors for Canadian Forestry. Environment Canada, Forestry Service. 6pp.

Brace, L.G. and K.M. Magar. 1968. Automated computation and plotting of stem analysis data. Can. Dept. For. Rur. Dev., For. Br. Pub. # 1209, 8pp.

Bruce, D. and F.X. Schumacher. 1942. Forest mensuration. McGraw-Hill Book Co. Inc., New York. 425 pp.

Carmean, Willard H. 1972. Site index curves for upland oaks in the Central States. Forest Science 18(2):109-120.

Carron, L.T. 1968. An outline of mensuration: with special reference to Australia. Australian National University Press, Canberra. 224 pp.

CBE Style Manual Committee. 1978. Council of Biology Editors style manual: a guide for authors, editors, and publishers in the biological sciences. 4th ed. Council of Biology Editors. 265 pp.

Chapeskie, D. and R. Fleet. 1981. Stem analysis program. Ontario Ministry of Natural Resources, Brockville. (unpublished).

Chapman, H.H. and W.H. Meyer. 1949. Forest mensuration. McGraw-Hill Book Co. Inc., New York. 522 pp.

Curtis, R.O. 1964. A stem-analysis approach to site-index curves. Forest Science 10(2):241-256.

Curtis, R.O., D.J. DeMars and F.R. Herman. 1974. Which dependent variable in site index - height - age regressions? Forest Science 20:74-87.

Duff, G.H. and N.J. Nolan. 1953. Growth and morphogenesis in the Canadian forest species, I. The controls of cambial and apical activity in *Pinus resinosa* Ait. Can. Jour. Botany 31:471-513.

Duff, G.H. and N.J. Nolan. 1957. Growth and morphogenesis in the Canadian forest species, II. Specific increments and their relation to the quality and activity of growth in *Pinus resinosa* Ait. Can. Jour. Botany 35:527-572.

Duff, G.H. and N.J. Nolan. 1958. Growth and morphogenesis in the Canadian forest species, III. The time scale of morphogenesis at the stem apex of *Pinus resinosa* Ait. Can. Jour. Botany 36:687-706.

Eklund, B. 1949. Skogsforskninginst. Avsringmatning-maskiner Medd. fran Skogsforskn. Inst. Stockholm, Sweden. 38:1-77.

Fries, J. and B. Matern. 1965. On the use of multivariate methods for the construction of tree taper curves. Paper # 9, Advisory Group of Forest Statisticians of I.U.F.R.O., Sec. 25, Conf. in Stockholm, Sweden.

Freese, F. 1960. Testing accuracy. Forest Science 6:139-149.

Goulding, C.J. 1971. Reducing the error in the calculation of the volume of sectioned logs. Can. Jour. For. Res. 1:267-268.

Graves, H.S. 1907. Forest mensuration. John Wiley and Sons, New York. 458 pp.

Griffin, H. and C.W. Yeatman. 1970. An Addo-X tape-punch programme for automated tree stem analysis. Petawawa For. Expt. Station, PS-X-14, 5pp.

Heger, L. 1965a. Morphogenesis of stems of Douglas fir (Pseudotsuga menziessi (Mirb.) Franco). Doctoral Dissertation, Faculty of Forestry, Univ. British Columbia, 176 pp. plus appendices.

Heger, L. 1965b. A trial of Hohenadl's method of stem form and stem volume estimation. For. Chron. 41(4):466-475.

Herman, F.R. and D.J. DeMars. 1970. Techniques and problems of stem analysis of old-growth conifers in the Oregon - Washington Cascade Range. In: J.H.G. Smith and J. Worrall (eds.), Tree-ring analysis with special reference to Northwest America, pp. 74-77. Univ. B. C. Fac. For. Bull. # 7, Vancouver.

Herman, F.R., D.J. DeMars and R.F. Woollard. 1975. Field and computer techniques for stem analysis of coniferous forest trees. USDA For. Serv. Res. Pap. PNW-194, 51 pp. Pacific Northwest For. and Range Expt. Station, Portland, Oregon.

Hinkley, D.V. 1978. Comparing means in one-way Analysis of Variance. Class handout for Statistics 5Ø21, The University of Minnesota.

Hosie, R.C. 1973. Native trees of Canada. Canadian Forestry Service, Department of the Environment. 38Ø pp.

Husch, B., C.I. Miller and T.W. Beers. 1972. Forest mensuration. The Ronald Press Co., New York. 41Ø pp.

Lenhart, J. David. 1972. An alternative procedure for improving height/age data from stem analysis. Forest Science 18:332.

Loetsch, F. and K.E. Haller. 1964. Forest Inventory. BLV Verlagsgeselleschaft, Munich. Vol. 1. 436 pp.

Loetsch, F., F. Zohrer and K.E. Haller. 1973. Forest Inventory. BLV Verlagsgeselleschaft, Munich. Vol. 2. 452 pp.

Mitchell, K.J. 1967. Simulation of the growth of even-aged stands of white spruce. Doctoral Dissertation, Graduate School, Yale Univ. 124pp.

Newnham, R.M. 1964. The development of a stand model for Douglas fir. Doctoral Dissertation, Faculty of Forestry, Univ. British Columbia, 219 pp.

Pluth, D.J. and D.R. Cameron. 1971. Announcing Fortran IV program for computing and graphing tree growth parameters from stem analysis. Forest Science 17:1Ø2.

Renton, J.J., M.J. Lanasa and E.H. Tryon. 1974. Radiography for observing wood features. Jour. Forestry 72(5):292-293.

Shea, S.R. and K.A. Armson. 1972. Stem analysis of jack pine (Pinus banksiana, Lamb.): Techniques and concepts. Can. Jour. For. Res. 2:392-4Ø6.

Snedecor, G.W. and W.G. Cochran. 198Ø. Statistical methods. The Iowa State University Press. 5Ø7 pp.

Spurr, S.H. 1952. Forest inventory. The Ronald Press Co., New York. 476 pp.

Steel, R.G.D. and J.H. Torrie. 196Ø. Principles and procedures of statisics. McGraw-Hill Book Co., New York. 481 pp.

Tryon, E.H., J.O. Cantrell, and K.L. Carvell. 1957. Effects of precipitation and temperature on increment of young poplar. For. Sci. 3(1):32-44.

Wang, E. 1976. Stem analysis program. Lakehead University, Thunder Bay. (unpublished).

Weisberg, Sanford. 198Ø. Applied linear regression. John Wiley and Sons, New York. 283 pp.

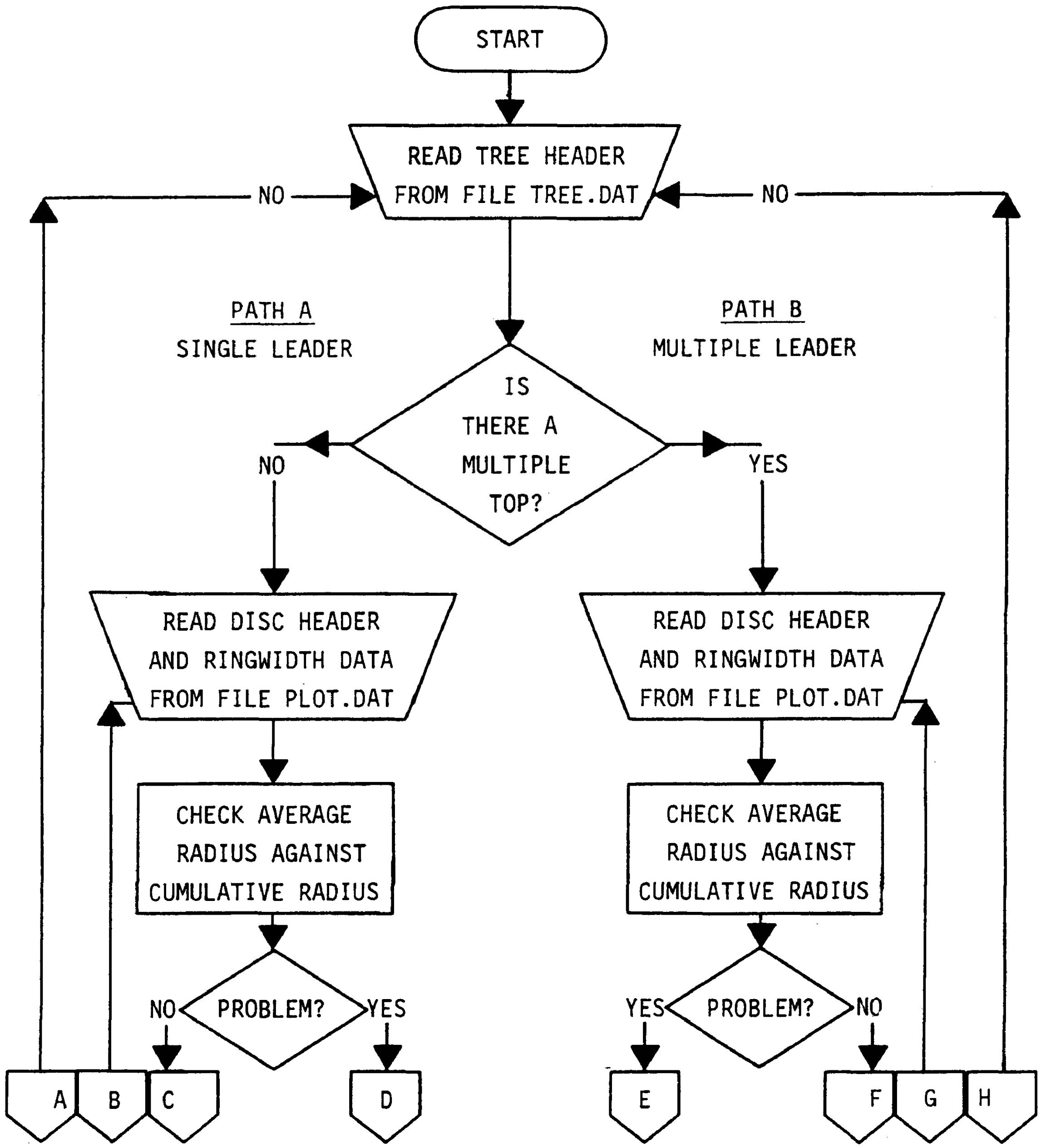
Whyte, A.G.D. 1971. Sectional measurement of trees: a rationalized method. New Zealand Jour. For. Sci. 1(1):74-79.

Wilson, R.W. 1964. An approach to silvicultural control of growth in eastern white pine trees, Pinus strobus L. Doctoral Dissertation, Faculty of Forestry, Graduate School, Yale Univ. 115 pp.

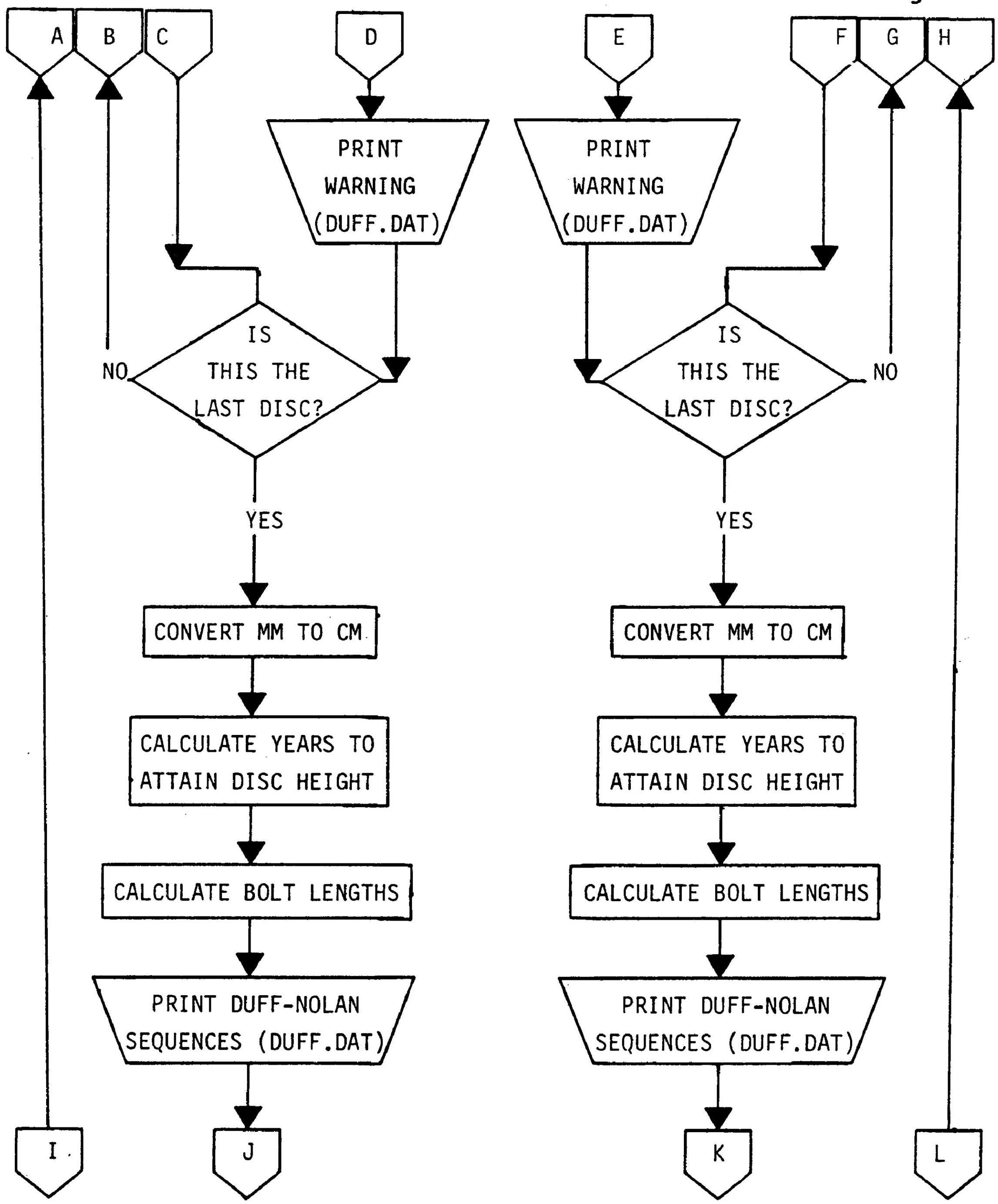
APPENDIX A

Program DUFFNO.FOR

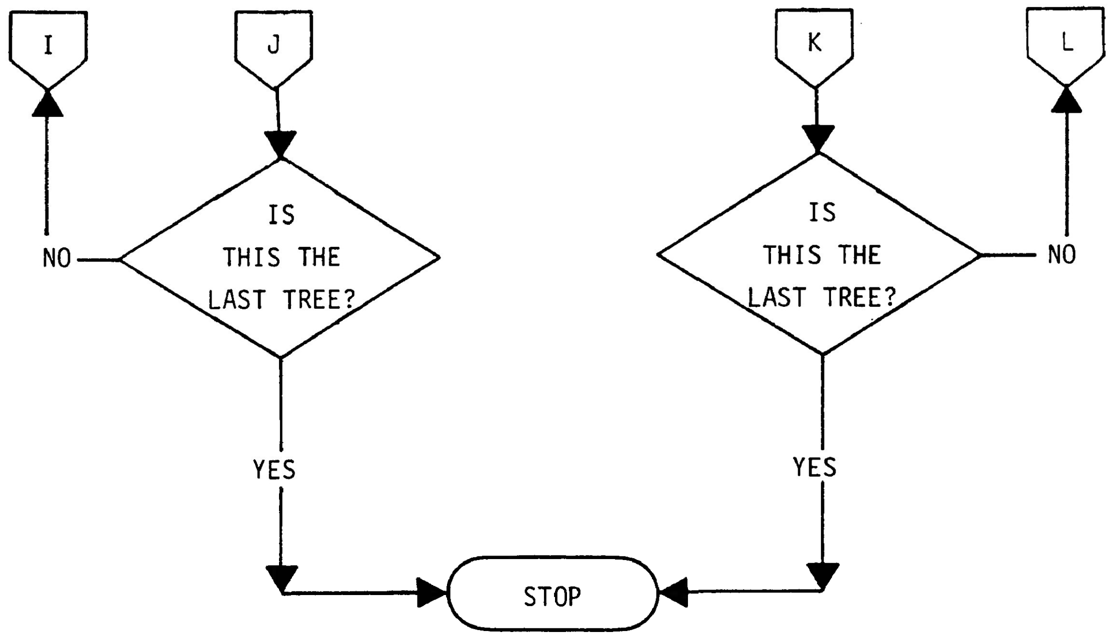
DUFFNO



APPENDIX A-1 Flowchart for algorithm DUFFNO.



APPENDIX A-1 Flowchart for algorithm DUFFNO.



APPENDIX A-1 Flowchart for algorithm DUFFNO.

APPENDIX A-2 DUFF.DAT file. The Duff-Nolan sequences are displayed for Tree # 1, a white spruce.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

DISC NUMBER	0	10	13	20	30	40	50	60	70	80	90	100	110	120	130	140	0
BOLT LENGTH (M)	0.00	1.00	0.30	0.70	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.60
DISC HEIGHT (M)	0.00	1.00	1.30	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	14.60
RING COUNT	37	32	31	30	28	26	24	22	20	19	16	13	10	8	4	2	0
AGE (YEARS)	0.	5.	6.	7.	9.	11.	13.	15.	17.	18.	21.	24.	27.	29.	33.	35.	37.
SBT (CM)	0.385	0.586	0.455	0.531	0.637	0.426	0.431	0.382	0.338	0.313	0.338	0.302	0.277	0.293	0.181	0.196	0.000
AVE RADIUS (CM)	17.107	13.117	12.710	12.094	11.626	10.814	10.378	9.532	9.319	8.121	7.090	6.075	4.589	3.004	1.277	0.491	0.000
RING WIDTH (CM)																	
	0.494	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.473	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.346	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.681	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.439	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.490	0.488	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.583	0.290	0.413	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.544	0.209	0.239	0.363	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.459	0.242	0.237	0.229	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.345	0.140	0.239	0.244	0.341	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.358	0.193	0.089	0.216	0.241	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.408	0.401	0.205	0.141	0.280	0.289	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.289	0.259	0.368	0.191	0.220	0.222	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.517	0.293	0.343	0.420	0.136	0.239	0.330	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.536	0.349	0.305	0.303	0.205	0.197	0.262	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.427	0.389	0.390	0.331	0.361	0.160	0.293	0.438	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.728	0.328	0.322	0.424	0.328	0.180	0.237	0.295	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.728	0.329	0.299	0.335	0.424	0.336	0.174	0.360	0.528	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.392	0.421	0.409	0.262	0.362	0.323	0.200	0.315	0.265	0.426	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.588	0.258	0.421	0.342	0.390	0.378	0.370	0.233	0.255	0.273	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.582	0.433	0.287	0.433	0.339	0.495	0.396	0.202	0.345	0.271	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	0.867	0.603	0.367	0.255	0.386	0.524	0.493	0.422	0.225	0.446	0.526	0.000	0.000	0.000	0.000	0.000	0.000
	0.670	0.456	0.645	0.427	0.486	0.391	0.591	0.389	0.252	0.325	0.319	0.000	0.000	0.000	0.000	0.000	0.000
	0.588	0.590	0.398	0.590	0.318	0.466	0.497	0.535	0.544	0.239	0.517	0.000	0.000	0.000	0.000	0.000	0.000
	0.586	0.802	0.515	0.355	0.461	0.551	0.410	0.649	0.587	0.464	0.527	0.534	0.000	0.000	0.000	0.000	0.000
	0.485	0.547	0.623	0.671	0.662	0.296	0.325	0.468	0.597	0.410	0.411	0.462	0.000	0.000	0.000	0.000	0.000
	0.436	0.689	0.495	0.776	0.471	0.486	0.535	0.457	0.792	0.543	0.314	0.516	0.000	0.000	0.000	0.000	0.000
	0.605	0.482	0.583	0.572	0.577	0.518	0.344	0.371	0.582	0.695	0.528	0.611	0.514	0.000	0.000	0.000	0.000
	0.451	0.563	0.507	0.688	0.628	0.573	0.562	0.466	0.575	0.682	0.470	0.319	0.524	0.000	0.000	0.000	0.000
	0.346	0.495	0.531	0.592	0.580	0.618	0.574	0.333	0.409	0.594	0.429	0.333	0.560	0.416	0.000	0.000	0.000
	0.484	0.424	0.517	0.599	0.667	0.709	0.545	0.450	0.764	0.475	0.596	0.461	0.595	0.368	0.000	0.000	0.000
	0.390	0.449	0.499	0.556	0.568	0.640	0.731	0.466	0.535	0.365	0.594	0.411	0.417	0.459	0.000	0.000	0.000
	0.180	0.489	0.478	0.418	0.545	0.658	0.698	0.495	0.467	0.268	0.478	0.526	0.396	0.467	0.000	0.000	0.000
	0.107	0.387	0.471	0.456	0.454	0.490	0.576	0.691	0.373	0.541	0.392	0.514	0.431	0.427	0.391	0.000	0.000
	0.118	0.413	0.457	0.344	0.437	0.421	0.488	0.659	0.528	0.443	0.324	0.451	0.454	0.347	0.321	0.000	0.000
	0.219	0.452	0.463	0.258	0.512	0.346	0.392	0.427	0.291	0.241	0.150	0.374	0.468	0.331	0.416	0.360	0.000
	0.168	0.254	0.595	0.303	0.247	0.308	0.355	0.411	0.405	0.420	0.515	0.563	0.230	0.189	0.149	0.131	0.000

APPENDIX A-3

DUFFNO Program Listing

```
**********************************************************************
*                                                                    *
*                     PROGRAM DUFFNO.FOR                             *
*                                                                    *
*              DUFFNC.FOR CREATES THE DUFF-NOLAN SERIES FROM STEM    *
*       ANALYSIS PRODUCED BY THE HOLMAN DIGIMICROMETER.  ANY NUMBER  *
*       OF TREES MAY BE PROCESSED AT ONE TIME.  RAW DATA IS STORED   *
*       IN TWO FILES.  TREE.DAT CONTAINS PERTINENT INFORMATION       *
*       REQUIRED TO PROCESS EACH TREE.  PLOT.DAT CONTAINS ALL        *
*       RINGWIDTH DATA FOR ALL THE DISCS TAKEN FROM EACH TREE.  THE  *
*       ORDER OF TREES IN TREE.DAT MUST CORRESPOND WITH THE DISC     *
*       DATA IN PLOT.DAT.  DISC DATA MUST BE ORGANIZED WITH DISCS    *
*       IN CONSECUTIVE AND ASCENDING ORDER (IE BASAL DISC TO TIP DISC). *
*              DUFFNO.FOR SHOULD BE USED AS A TOOL TO COMPLETE THE   *
*       EDITING OF RAW DATA FILES BEFORE THEY ARE USED IN STEM.FOR,  *
*       THE STEM ANALYSIS ALGORITHM.                                 *
*              THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD    *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE   *
*       VMS OPERATING SYSTEM.                                        *
*              EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS   *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE   *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR   *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                       *
*                                                                    *
*       AUTHOR:           JOANNE KAVANAGH                            *
*                       SCHOOL OF FORESTRY                           *
*                       LAKEHEAD UNIVERSITY                          *
*                       THUNDER BAY, ONTARIO, P7B 5E1                *
*                           AUGUST, 1981                             *
*                                                                    *
*                       COPYRIGHT APPLIED FOR                        *
*                                                                    *
**********************************************************************
*                                                                    *
*                     VARIABLE IDENTIFICATION                        *
*                                                                    *
*       PN = PLOT NUMBER                                             *
*       TN = TREE NUMBER                                             *
*       SC = SPECIES CODE; ACCORDING TO OMNR CODES                   *
*       AGE = TOTAL TREE AGE IN YEARS                                *
*       YR = YEAR CUT                                                *
*       TOTHT = TOTAL TREE HEIGHT IN METRES                          *
*       BOLEHT = TOTAL BOLE HEIGHT IN METRES TO SPLIT                *
*       NLEAD = NUMBER OF LEADERS (5 MAXIMUM ALLOWED)                *
*       NDISCS = TOTAL NUMBER OF TREE DISCS                          *
*       ITD = NDISCS + 1                                             *
*       NBDISC = TOTAL NUMBER OF BOLE DISCS                          *
*       NDISCL(K) = TOTAL NUMBER OF DISCS PER LEADER                 *
*       ITD(K) = NDISCL(K) + 1                                       *
*       LLNGTH(K) = LEADER LENGTH IN METRES                          *
*       DN(J) = DISC NUMBER                                          *
*       RC(J) = RING COUNT ON EACH DISC                              *
*       DNL(JJ,K) = DISC NUMBER FOR MULTIPLE TOPS                    *
```

```
*       RCL(JJ,K) = RING COUNT ON EACH DISC FOR MULTIPLE TOPS               *
*       DHT(J) = DISC HEIGHT IN METRES                                      *
*       AVERAD(J) = AVERAGE RADIUS OF DISC                                  *
*       SBT(J) = SINGLE BARK THICKNESS OF DISC                              *
*       RW(I,J) = RING WIDTHS BY YEAR PER DISC                              *
*       DNS(I,J) = DUFF-NOLAN SERIES BY YEAR PER DISC                       *
*       TYRS(J) = TOTAL YEARS REQUIRED TO ATTAIN DISC HEIGHT                *
*       BOLT(J) = BOLT LENGTH BETWEEN CONSECUTIVE DISCS                     *
*       DHTL(JJ,K) = DISC HEIGHT OF DISCS ON MULTIPLE TOPS                  *
*       AVRADL(JJ,K) = AVERAGE RADIUS OF DISCS ON MULTIPLE TOPS             *
*       SBTL(JJ,K) = SINGLE BARK THICKNESS OF DISCS ON MULTIPLE TOPS        *
*       RWL(I,JJ,K) = RING WIDTHS BY YEAR PER DISC ON MULTIPLE TOPS         *
*       DNSL(I,JJ,K) = DUFF-NOLAN SERIES BY YEAR PER DISC FOR               *
*               MULTIPLE TOPS                                               *
*       TYRSL(JJ,K) = TOTAL YEARS REQUIRED TO ATTAIN DISC HEIGHT ON         *
*               MULTIPLE TOPS                                               *
*       BOLTL(JJ,K) = BOLT LENGTH BETWEEN CONSECUTIVE DISCS ON              *
*               MULTIPLE TOPS                                               *
*       I = 1 TO A MAXIMUM OF 300 YEARS                                     *
*       J = 1 TO A MAXIMUM OF 60 DISCS                                      *
*       JJ = 1 TO A MAXIMUM OF 25 DISCS IN EACH MULTIPLE TOP                *
*       K = 1 TO A MAXIMUM OF 5 LEADERS                                     *
*                                                                           *
*****************************************************************************
*                                                                           *
*                         STORAGE ALLOCATION                                *
*                                                                           *
        REAL AGE,TOTHT,BOLEHT,DHT(60),AVERAD(60),SBT(60),RW(300,60),
     1DNS(300,60),TYRS(60),BOLT(60),DHTL(25,5),AVRADL(25,5),SBTL(25,
     25),RWL(300,25,5),DNSL(300,25,5),TYRSL(25,5),BOLTL(25,5),
     3LLNGTH(5)
        INTEGER PN,TN,SC,YR,NLEAD,NDISCS,NBDISC,NDISCL(5),DN(60),RC
     1(60),ITD,DNL(25,5),RCL(25,5),ITDL(5)
        COMMON/OUTP/PN,TN,SC,AGE,NLEAD
        COMMON/OUTP1/DHT,BOLT,RC,TYRS,SBT,DNS,DN,AVERAD
        COMMON/OUTP2/DHTL,BOLTL,RCL,TYRSL,SBTL,DNSL,DNL,AVRADL
*                                                                           *
*****************************************************************************
*                                                                           *
*                         OPEN I/O UNITS                                    *
*                                                                           *
        OPEN(UNIT=20,FILE='TREE.DAT',STATUS='OLD')
        OPEN(UNIT=21,FILE='PLOT.DAT',STATUS='OLD')
        OPEN(UNIT=22,FILE='DUFF.DAT',STATUS='NEW')
*                                                                           *
*****************************************************************************
*                                                                           *
*       START LOOP FOR DUFF-NOLAN SERIES; ONE TREE AT A TIME                *
*                                                                           *
*****************************************************************************
*                                                                           *
*                         INITIALIZE VARIABLES                              *
*                                                                           *
10      CONTINUE
        PN=0
```

```
        TN=0
        SC=0
        AGE=0.0
        YR=0
        TOTHT=0.0
        BOLEHT=0.0
        NLEAD=0
        NDISCS=0
        NBDISC=0
        ITD=0
        DO 20 I=1,5
        NDISCL(I)=0
        LLNGTH(I)=0.0
        ITDL(I)=0
20      CONTINUE
        DO 30 J=1,60
        DHT(J)=0.0
        AVERAD(J)=0.0
        SBT(J)=0.0
        TYRS(J)=0.0
        BOLT(J)=0.0
        RC(J)=0
        DN(J)=0
        DO 40 I=1,300
        RW(I,J)=0.0
        DNS(I,J)=0.0
40      CONTINUE
30      CONTINUE
        DO 50 J=1,25
        DO 60 K=1,5
        DHTL(J,K)=0.0
        AVRADL(J,K)=0.0
        SBTL(J,K)=0.0
        TYRSL(J,K)=0.0
        BOLTL(J,K)=0.0
        DNL(J,K)=0
        RCL(J,K)=0
        DO 70 I=1,300
        RWL(I,J,K)=0.0
        DNSL(I,J,K)=0.0
70      CONTINUE
60      CONTINUE
50      CONTINUE
*                                                                        *
**************************************************************************
*                                                                        *
*       READ TREE HEADER FROM FILE TREE.DAT                              *
*       WHEN END OF FILE OCCURS, GO TO 90                                *
*                                                                        *
        READ(20,80,END=90)PN,TN,SC,AGE,YR,TOTHT,BOLEHT,NLEAD,NDISCS,
        1NBDISC,(NDISCL(I),I=1,5),(LLNGTH(I),I=1,5)
80      FORMAT(I2,I3,I2,F3.0,I5,2F5.2,I2,2I3,5I2,5F4.2)
*                                                                        *
*       TEST FOR A MULTIPLE LEADER TREE.  IF THERE IS A MULTIPLE TOP     *
*       GO TO 100                                                        *
```

```
*                                                                        *
        IF(NLEAD.GT.1) GO TO 100
*                                                                        *
*       READ DISC HEADERS WITH ACCOMPANYING RING DATA FOR SINGLE         *
*       LEADER TREES FROM FILE PLOT.DAT                                  *
*                                                                        *
        DO 110 J=1,NDISCS
        READ(21,120),DN(J),DHT(J),RC(J),AR,SBTT
120     FORMAT(12X,I4,4X,F4.2,I3,12X,F9.2,F7.2)
        AVERAD(J)=AR/10.0
        SBT(J)=SBTT/10.0
        READ(21,130)(RW(I,J),I=1,RC(J))
130     FORMAT(4X,10F7.2)
*                                                                        *
*       RESET CHECK TO ZERO                                              *
*                                                                        *
        CHECK=0.0
        DO 140 I=1,RC(J)
        CHECK=CHECK+RW(I,J)
140     CONTINUE
        DIFF=ABS(CHECK-AR)
*                                                                        *
*       IF THE GIVEN AVERAGE RADIUS DOES NOT EQUAL THE CALCULATED        *
*       AVERAGE RADIUS, PRINT A WARNING MESSAGE                          *
*                                                                        *
        IF(DIFF.LT.0.01) GO TO 150
        WRITE(22,160)PN,TN,DN(J),AR,CHECK
160     FORMAT('0',T10,'RING WIDTHS DO NOT EQUAL AVERAGE RADIUS FOR
        2THIS DISC',3I4,'AVERAD',F9.2,'CHECK',F9.2)
150     CONTINUE
110     CONTINUE
*                                                                        *
*       CREATE TIP DISC FOR THE TREE                                     *
*                                                                        *
        ITD=NDISCS+1
        DHT(ITD)=TOTHT
        BOLT(ITD)=TOTHT-DHT(NDISCS)
        RC(ITD)=0
        SBT(ITD)=0.0
        DNS(1,ITD)=0.0
*                                                                        *
*       CHANGE RING WIDTH MEASUREMENT UNITS FROM MILLIMETRES TO          *
*       CENTIMETRES                                                      *
*                                                                        *
        DO 170 J=1,NDISCS
        NN=RC(J)
        DO 180 I=1,NN
        IDIFF=RC(1)-RC(J)
        K=IDIFF+I
        DNS(K,J)=RW(I,J)/10.0
180     CONTINUE
170     CONTINUE
*                                                                        *
*       CALCULATE THE NUMBER OF YEARS REQUIRED TO ATTAIN DISC HEIGHT     *
*                                                                        *
```

```
      DO 190 J=1,ITD
      TYRS(J)=AGE-(RC(J)*1.0)
190   CONTINUE
*                                                                   *
*     CALCULATE BOLT LENGTHS BETWEEN CONSECUTIVE DISCS              *
*                                                                   *
      BOLT(1)=DHT(1)-0.0
      DO 200 J=2,NDISCS
      JJ=J-1
      BOLT(J)=DHT(J)-DHT(JJ)
200   CONTINUE
*                                                                   *
*     PRINT OUT ORIGINAL MEASUREMENTS IN TABULAR FORMAT             *
*                                                                   *
      CALL OUTPT1(ITD)
*                                                                   *
*     END OF DUFF-NOLAN SERIES LOOP FOR SINGLE LEADER TREES;        *
*     GO TO 210                                                     *
*                                                                   *
      GO TO 210
100   CONTINUE
*                                                                   *
*     READ DISC HEADERS WITH ACCOMPANYING RING DATA FOR MULTIPLE    *
*     LEADER TREES (BOLE ONLY) FROM FILE PLOT.DAT                   *
*                                                                   *
      DO 220 J=1,NBDISC
      READ(21,120)DN(J),DHT(J),RC(J),AR,SBTT
      AVERAD(J)=AR/10.0
      SBT(J)=SBTT/10.0
      READ(21,130)(RW(I,J),I=1,RC(J))
*                                                                   *
*     RESET CHECK TO ZERO                                           *
*                                                                   *
      CHECK=0.0
      DO 230 I=1,RC(J)
      CHECK=CHECK+RW(I,J)
230   CONTINUE
      DIFF=ABS(CHECK-AR)
*                                                                   *
*     IF THE GIVEN AVERAGE RADIUS DOES NOT EQUAL THE CALCULATED     *
*     AVERAGE RADIUS, PRINT A WARNING MESSAGE                       *
*                                                                   *
      IF(DIFF.LT.0.01) GO TO 240
      WRITE(22,160)PN,TN,DN(J),AR,CHECK
240   CONTINUE
220   CONTINUE
*                                                                   *
*     READ DISC HEADERS WITH ACCOMPANYING RING DATA FOR MULTIPLE    *
*     LEADER TREES (LEADERS ONLY) FROM FILE PLOT.DAT                *
*                                                                   *
      DO 250 K=1,NLEAD
      ND=NDISCL(K)
*                                                                   *
*     IF THERE ARE NO DISCS FOR THE LEADER, CONTINUE AFTER 260      *
*                                                                   *
```

```
      IF(ND.EQ.0) GO TO 260
      DO 270 J=1,ND
      READ(21,120)DNL(J,K),DHTL(J,K),RCL(J,K),AR,SBTT
      AVRADL(J,K)=AR/10.0
      SBTL(J,K)=SBTT/10.0
      READ(21,130)(RWL(I,J,K),I=1,RCL(J,K))
*                                                                  *
*     RESET CHECK TO ZERO                                          *
*                                                                  *
      CHECK=0.0
      DO 280 I=1,RCL(J,K)
      CHECK=CHECK+RWL(I,J,K)
280   CONTINUE
      DIFF=ABS(CHECK-AR)
*                                                                  *
*     IF THE GIVEN AVERAGE RADIUS DOES NOT EQUAL THE CALCULATED    *
*     AVERAGE RADIUS, PRINT A WARNING MESSAGE                      *
*                                                                  *
      IF(DIFF.LT.0.01) GO TO 290
      WRITE(22,160)PN,TN,DNL(J,K),AR,CHECK
290   CONTINUE
270   CONTINUE
*                                                                  *
*     CREATE TIP DISC FOR EACH LEADER                              *
*                                                                  *
260   CONTINUE
      ITDL(K)=ND+1
      J=ITDL(K)
      DHTL(J,K)=BOLEHT+LLNGTH(K)
      RCL(J,K)=0
      SBTL(J,K)=0.0
      DNSL(1,J,K)=0.0
      IF(ND.EQ.0) GO TO 300
      BOLTL(J,K)=DHTL(J,K)-DHTL(ND,K)
      GO TO 250
300   CONTINUE
      BOLTL(J,K)=LLNGTH(K)
250   CONTINUE
*                                                                  *
*     CHANGE RING WIDTH MEASUREMENT UNITS FROM MILLIMETRES TO      *
*     CENTIMETRES                                                  *
*                                                                  *
      DO 310 J=1,NBDISC
      NN=RC(J)
      DO 320 I=1,NN
      IDIFF=RC(1)-RC(J)
      K=IDIFF+I
      DNS(K,J)=RW(I,J)/10.0
320   CONTINUE
310   CONTINUE
      DO 330 K=1,NLEAD
      ND=NDISCL(K)
      DO 340 J=1,ND
      NN=RCL(J,K)
      DO 350 I=1,NN
```

```
        IDIFF=RCL(1,K)-RCL(J,K)
        KK=IDIFF+I
        DNSL(KK,J,K)=RWL(I,J,K)/10.0
350     CONTINUE
340     CONTINUE
330     CONTINUE
*                                                                   *
*       CALCULATE THE NUMBER OF YEARS REQUIRED TO ATTAIN DISC HEIGHT    *
*                                                                   *
        DO 360 J=1,NBDISC
        TYRS(J)=AGE-(RC(J)*1.0)
360     CONTINUE
        DO 370 K=1,NLEAD
        DO 380 J=1,ITDL(K)
        TYRSL(J,K)=AGE-(RCL(J,K)*1.0)
380     CONTINUE
370     CONTINUE
*                                                                   *
*       CALCULATE BOLT LENGTHS BETWEEN CONSECUTIVE DISCS            *
*                                                                   *
        BOLT(1)=DHT(1)-0.0
        DO 390 J=2,NBDISC
        JJ=J-1
        BOLT(J)=DHT(J)-DHT(JJ)
390     CONTINUE
        DO 400 K=1,NLEAD
        BOLTL(1,K)=DHTL(1,K)-DHT(NBDISC)
        ND=NDISCL(K)
        DO 410 J=2,ND
        JJ=J-1
        BOLTL(J,K)=DHTL(J,K)-DHTL(JJ,K)
410     CONTINUE
400     CONTINUE
*                                                                   *
*       PRINT OUT ORIGINAL MEASUREMENTS IN TABULAR FORMAT           *
*                                                                   *
        CALL OUTPT1(NBDISC)
        CALL OUTPT2(ITDL)
210     CONTINUE
*                                                                   *
*       END OF DUFF-NOLAN SERIES LOOP FOR MULTIPLE LEADER TREES;    *
*       GO TO 10 AND PROCESS THE NEXT TREE                          *
*                                                                   *
        GO TO 10
*                                                                   *
*       WHEN ALL TREES ARE PROCESSED, END RUN                       *
*                                                                   *
90      CONTINUE
        CLOSE(UNIT=20)
        CLOSE(UNIT=21)
        CLOSE(UNIT=22)
        STOP
        END
*                                                                   *
*********************************************************************
```

```
************************************************************************
*                                                                      *
      SUBROUTINE OUTPT1(ITD)
*                                                                      *
************************************************************************
*                                                                      *
*                      SUBROUTINE IDENTIFICATION                       *
*                                                                      *
*             THIS SUBROUTINE ORGANIZES THE INPUT DATA INTO TABULAR    *
*       OUTPUT FOR SINGLE LEADER TREES, OR THE MAIN BOLE OF MULTIPLE   *
*       TOP TREES.                                                     *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD       *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE     *
*       VMS OPERATING SYSTEM.                                          *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM    *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY   *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS     *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE     *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR     *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                         *
*                                                                      *
*       AUTHOR:         JOANNE KAVANAGH                                *
*                      SCHOOL OF FORESTRY                              *
*                      LAKEHEAD UNIVERSITY                             *
*                      THUNDER BAY, ONTARIO, P7B 5E1                   *
*                                                                      *
************************************************************************
*                                                                      *
*                      VARIABLE IDENTIFICATION                         *
*                                                                      *
*       ALL VARIABLES ARE COMMON TO THE MAIN PROGRAM                   *
*                                                                      *
      REAL AGE,DHT(60),SBT(60),DNS(300,60),TYRS(60),BOLT(60),
     2AVERAD(60)
      INTEGER ITD,CODE(75),PN,TN,SC,NLEAD,DN(60),RC(60)
      COMMON/OUTP/PN,TN,SC,AGE,NLEAD
      COMMON/OUTP1/DHT,BOLT,RC,TYRS,SBT,DNS,DN,AVERAD
*                                                                      *
************************************************************************
*                                                                      *
*                      DATA SPECIFICATION                              *
*                                                                      *
      DATA NC/0/
      DATA CODE/'PW','PR','PJ','PS','  ','  ','  ','  ','  ','  ',
     2'  ','SW','SB','  ','  ','  ','  ','  ','HE','BF','  ','  ',
     3'  ','CE','TA','  ','  ','  ','  ','MH','  ','MR','MS','  ',
     4'  ','  ','BY','BW','  ','OW','OR','  ','  ','BE','AB','AW',
     5'  ','  ','EM','  ','BD','  ','  ','  ','  ','ID','  ','CB',
     6'  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ',
     7'  ','  ','  ','  ','PO'/
*                                                                      *
************************************************************************
*                                                                      *
*       IF THERE ARE 18 DISCS OR LESS, GO TO 10                        *
*                                                                      *
```

```
      IF(ITD.LE.18) GO TO 10
      IPAGES=ITD/18
      IREM=ITD-(IPAGES*18)
      I1=1
      DO 20 K=1,IPAGES
*                                                                    *
*     PRINT IDENTIFYING INFORMATION                                  *
*                                                                    *
      WRITE(22,30)PN,TN,CODE(SC),NLEAD
30    FORMAT('1',/////,T31,'PLOT #',I3,5X,'TREE #',I3,5X,'SPECIES
     1 CODE ',A2,5X,'NUMBER OF LEADERS',I2,5X)
      NC=K*18
      WRITE(22,40)(DN(I),I=I1,NC)
40    FORMAT('0',T4,'DISC NUMBER',4X,18I6)
      WRITE(22,50)(BOLT(I),I=I1,NC)
50    FORMAT('0',T4,'BOLT LENGTH (M)',1X,18F6.2)
      WRITE(22,60)(DHT(I),I=I1,NC)
60    FORMAT('0',T4,'DISC HEIGHT (M)',1X,18F6.2)
      WRITE(22,70)(RC(I),I=I1,NC)
70    FORMAT('0',T4,'RING COUNT',6X,18I6)
      WRITE(22,80)(TYRS(I),I=I1,NC)
80    FORMAT('0',T4,'AGE (YEARS)',5X,18F6.0)
      WRITE(22,90)(SBT(I),I=I1,NC)
90    FORMAT('0',T4'SBT (CM)',8X,18F6.3)
      WRITE(22,100)(AVERAD(I),I=I1,NC)
100   FORMAT('0',T4,'AVE RADIUS (CM)',1X,18F6.3)
      WRITE(22,110)
110   FORMAT(/,T4,'RING WIDTH (CM)')
*                                                                    *
*     PRINT DUFF-NOLAN SERIES IN TABULAR FORMAT                      *
*                                                                    *
      NY=RC(1)
      N=(RC(1)-RC(I1))+1
      DO 120 J=N,NY
      WRITE(22,130)(DNS(J,L),L=I1,NC)
130   FORMAT(' ',T20,18F6.3)
120   CONTINUE
      I1=I1+18
20    CONTINUE
      IF(IREM.EQ.0) GO TO 150
      NC=NC+IREM
      GO TO 140
10    CONTINUE
      I1=1
      NC=ITD
140   CONTINUE
*                                                                    *
*     PRINT IDENTIFYING INFORMATION                                  *
*                                                                    *
      WRITE(22,30)PN,TN,CODE(SC),NLEAD
      WRITE(22,40)(DN(I),I=I1,NC)
      WRITE(22,50)(BOLT(I),I=I1,NC)
      WRITE(22,60)(DHT(I),I=I1,NC)
      WRITE(22,70)(RC(I),I=I1,NC)
      WRITE(22,80)(TYRS(I),I=I1,NC)
```

```
        WRITE(22,90)(SBT(I),I=I1,NC)
        WRITE(22,100)(AVERAD(I),I=I1,NC)
        WRITE(22,110)
*                                                                      *
*       PRINT DUFF-NOLAN SERIES IN TABULAR FORMAT                      *
*                                                                      *
        NY=RC(1)
        N=(RC(1)-RC(I1))+1
        DO 150 J=N,NY
        WRITE(22,130)(DNS(J,L),L=I1,NC)
150     CONTINUE
        RETURN
        END
*                                                                      *
************************************************************************
```

```
*************************************************************************
*                                                                       *
      SUBROUTINE OUTPT2(ITDL)
*                                                                       *
*************************************************************************
*                                                                       *
*                       SUBROUTINE IDENTIFICATION                       *
*                                                                       *
*             THIS SUBROUTINE ORGANIZES THE INPUT DATA INTO TABULAR     *
*       OUTPUT FOR LEADERS OF MULTIPLE TOP TREES.                       *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD        *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE      *
*       VMS OPERATING SYSTEM.                                           *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM     *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY    *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS      *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE      *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR      *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                          *
*                                                                       *
*       AUTHOR:          JOANNE KAVANAGH                                *
*                       SCHOOL OF FORESTRY                              *
*                       LAKEHEAD UNIVERSITY                             *
*                       THUNDER BAY, ONTARIO, P7B 5E1                   *
*                                                                       *
*************************************************************************
*                                                                       *
*                       VARIABLE IDENTIFICATION                         *
*                                                                       *
*       ALL VARIABLES ARE COMMON TO THE MAIN PROGRAM                    *
*                                                                       *
*************************************************************************
*                                                                       *
      REAL AGE,DHTL(25,5),SBTL(25,5),DNSL(300,25,5),TYRSL(25,5),
     1BOLTL(25,5),AVRADL(25,5)
      INTEGER ITDL(5),CODE(75),PN,TN,SC,NLEAD,DNL(25,5),RCL(25,5)
      COMMON/OUTP/PN,TN,SC,AGE,NLEAD
      COMMON/OUTP2/DHTL,BOLTL,RCL,TYRSL,SBTL,DNSL,DNL,AVRADL
*                                                                       *
*************************************************************************
*                                                                       *
*                       DATA SPECIFICATION                              *
*                                                                       *
      DATA NC/0/
      DATA CODE/'PW','PR','PJ','PS','  ','  ','  ','  ','  ','  ',
     2'  ','SW','SB','  ','  ','  ','  ','  ','HE','BF','  ','  ',
     3'  ','CE','TA','  ','  ','  ','  ','MH','  ','MR','MS','  ',
     4'  ','  ','BY','BW','  ','OW','OR','  ','  ','BE','AB','AW',
     5'  ','  ','EM','  ','BD','  ','  ','  ','  ','ID','  ','CB',
     6'  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ',
     7'  ','  ','  ','  ','PO'/
*                                                                       *
*************************************************************************
*                                                                       *
      DO 10 KK=1,NLEAD
```

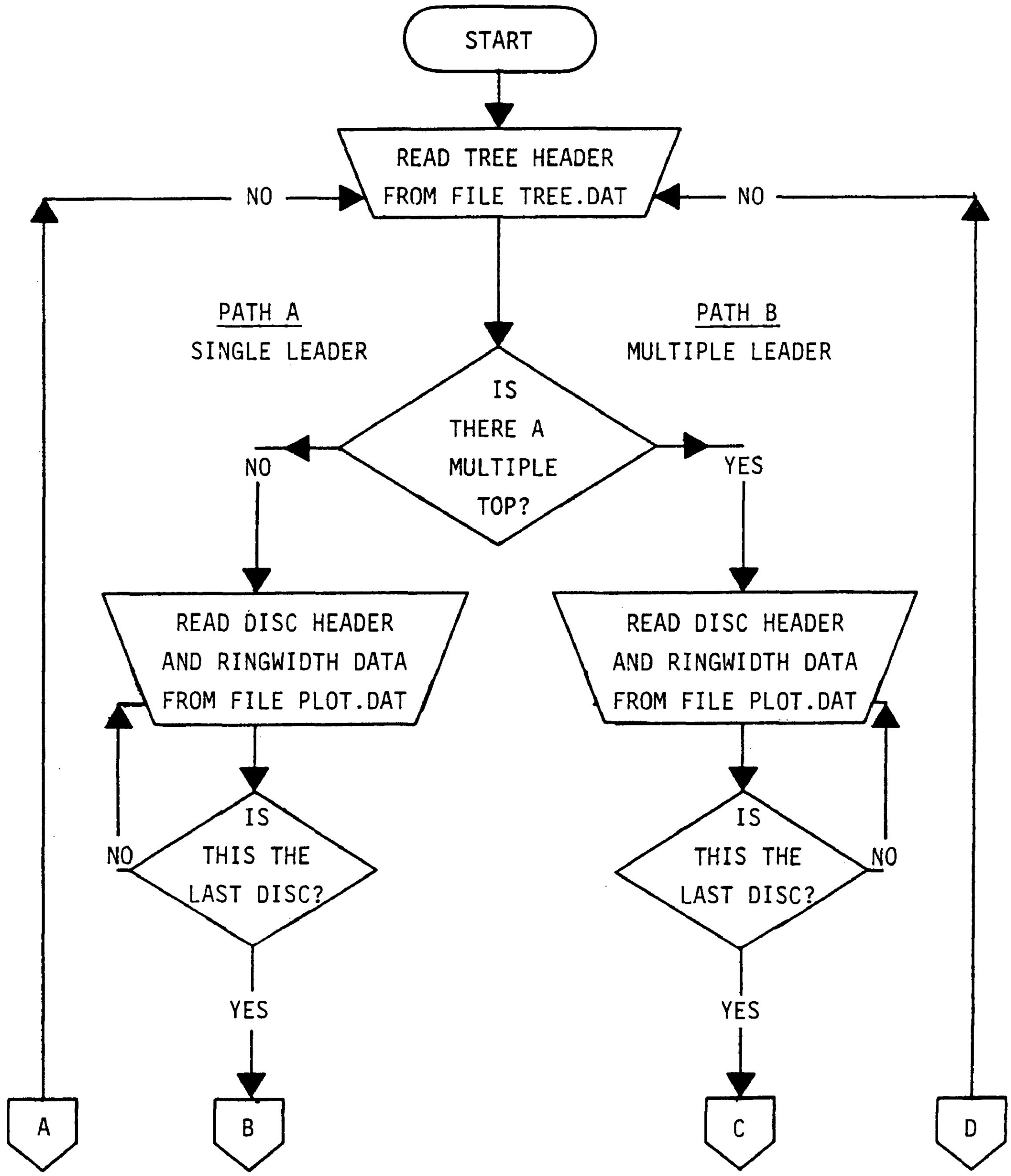
```
*                                                                      *
*       IF THERE ARE 18 DISCS OR LESS, GO TO 20                        *
*                                                                      *
        IF(ITDL(KK).LE.18) GO TO 20
        IPAGES=ITDL(KK)/18
        IREM=ITDL(KK)-(IPAGES*18)
        I1=1
        DO 30 K=1,IPAGES
*                                                                      *
*       PRINT IDENTIFYING INFORMATION                                  *
*                                                                      *
        WRITE(22,40)PN,TN,CODE(SC),NLEAD,KK
40      FORMAT('1',/////,T21,'PLOT #',I3,5X,'TREE #',I3,5X,'SPECIES
        1 CODE  ',A2,5X,'NUMBER OF LEADERS',I2,5X,'LEADER NUMBER',I2)
        NC=K*18
        WRITE(22,50)(DNL(I,KK),I=I1,NC)
50      FORMAT('0',T4,'DISC NUMBER',4X,18I6)
        WRITE(22,60)(BOLTL(I,KK),I=I1,NC)
60      FORMAT('0',T4,'BOLT LENGTH (M)',1X,18F6.2)
        WRITE(22,70)(DHTL(I,KK),I=I1,NC)
70      FORMAT('0',T4,'DISC HEIGHT (M)',1X,18F6.2)
        WRITE(22,80)(RCL(I,KK),I=I1,NC)
80      FORMAT('0',T4,'RING COUNT',6X,18I6)
        WRITE(22,90)(TYRSL(I,KK),I=I1,NC)
90      FORMAT('0',T4,'AGE (YEARS)',5X,18F6.0)
        WRITE(22,100)(SBTL(I,KK),I=I1,NC)
100     FORMAT('0',T4,'SBT (CM)',8X,18F6.3)
        WRITE(22,110)(AVRADL(I,KK),I=I1,NC)
110     FORMAT('0',T4,'AVE RADIUS (CM)',1X,18F6.3)
        WRITE(22,120)
120     FORMAT(/,T4,'RING WIDTH (CM)')
*                                                                      *
*       PRINT DUFF-NOLAN SERIES IN TABULAR FORMAT                      *
*                                                                      *
        NY=RCL(1,KK)
        N=(RCL(1,KK)-RCL(I1,KK))+1
        DO 130 J=N,NY
        WRITE(22,140)(DNSL(J,L,KK),L=I1,NC)
140     FORMAT(' ',T20,18F6.3)
130     CONTINUE
        I1=I1+18
30      CONTINUE
        IF(IREM.EQ.0) GO TO 160
        NC=NC+IREM
        GO TO 150
20      CONTINUE
        I1=1
        NC=ITDL(KK)
150     CONTINUE
*                                                                      *
*       PRINT IDENTIFYING INFORMATION                                  *
*                                                                      *
        WRITE(22,40)PN,TN,CODE(SC),NLEAD,KK
        WRITE(22,50)(DNL(I,KK),I=I1,NC)
        WRITE(22,60)(BOLTL(I,KK),I=I1,NC)
```

```
        WRITE(22,70)(DHTL(I,KK),I=I1,NC)
        WRITE(22,80)(RCL(I,KK),I=I1,NC)
        WRITE(22,90)(TYRSL(I,KK),I=I1,NC)
        WRITE(22,100)(SBTL(I,KK),I=I1,NC)
        WRITE(22,110)(AVRADL(I,KK),I=I1,NC)
        WRITE(22,120)
*                                                                      *
*       PRINT DUFF-NOLAN SERIES IN TABULAR FORMAT                      *
*                                                                      *
        NY=RCL(1,KK)
        N=(RCL(1,KK)-RCL(I1,KK))+1
        DO 160 J=N,NY
        WRITE(22,140)(DNSL(J,L,KK),L=I1,NC)
160     CONTINUE
10      CONTINUE
        RETURN
        END
*                                                                      *
************************************************************************
```

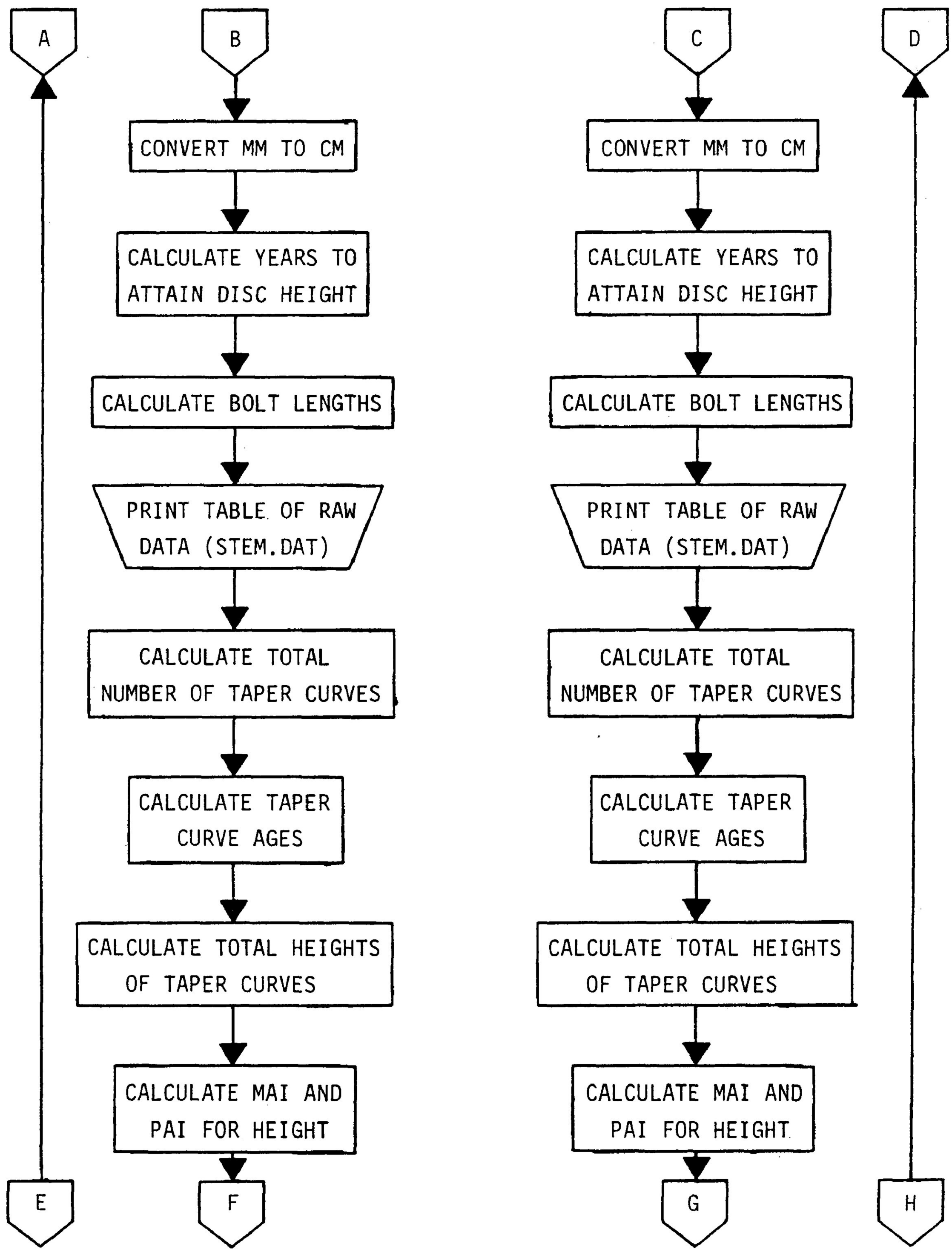
APPENDIX B

Program STEM.FOR

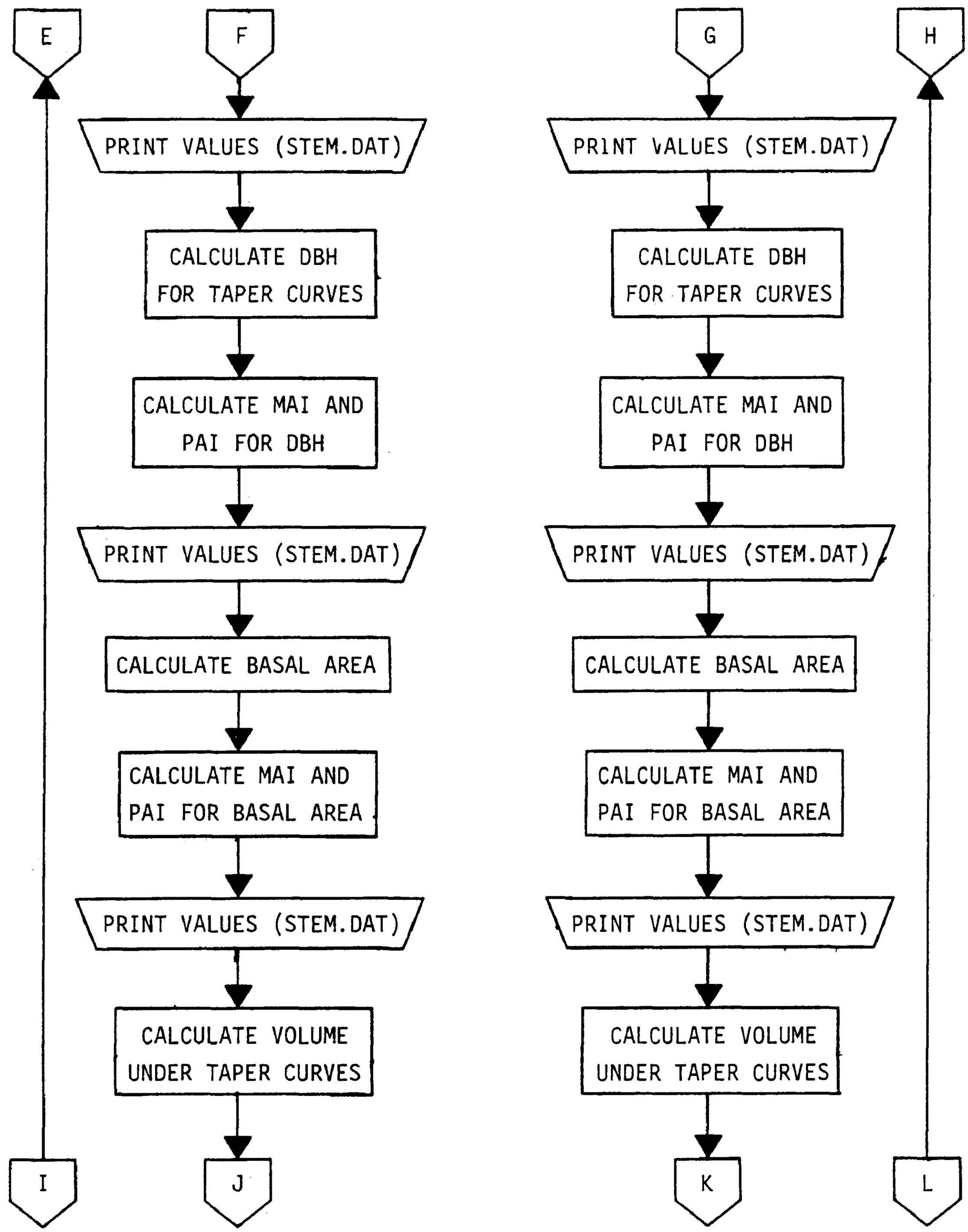
STEM



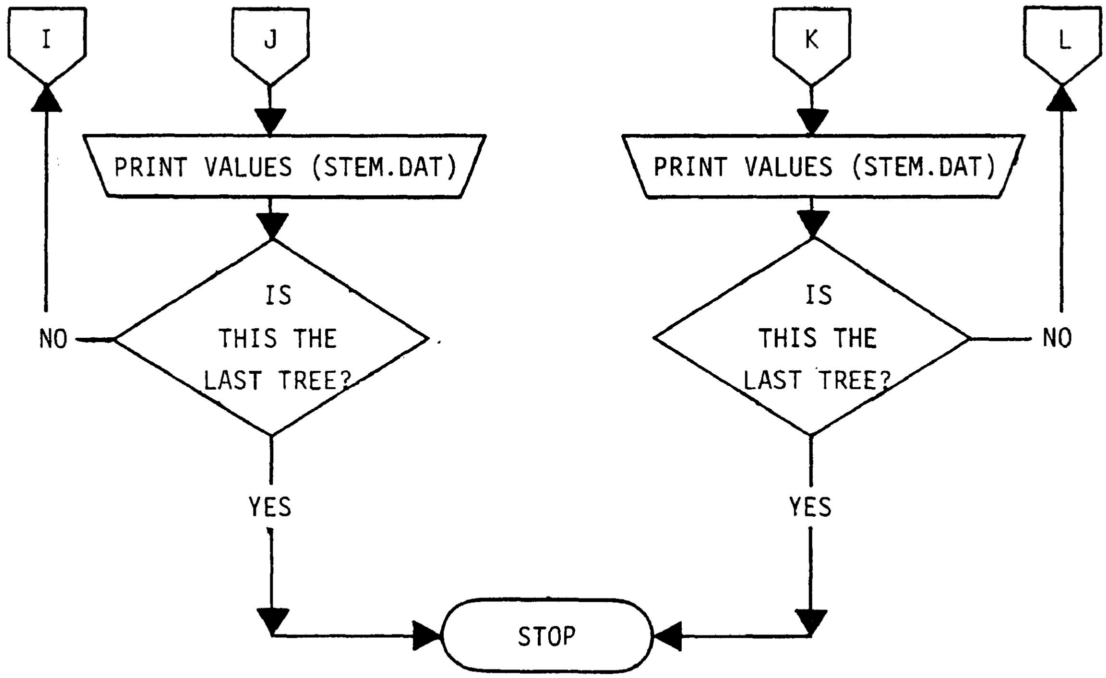
APPENDIX B-1 Flowchart for algorithm STEM.



APPENDIX B-1 Flowchart for algorithm STEM.



APPENDIX B-1 Flowchart for algorithm STEM.



APPENDIX B-1 Flowchart for algorithm STEM.

APPENDIX B-2 STEM.DAT file. Radius (inside bark) measurements are displayed for Tree #1, a white spruce.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

DISC NUMBER	0	10	13	20	30	40	50	60	70	80	90	100	110	120	130	140	0
BOLT LENGTH (M)	0.00	1.00	0.30	0.70	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.60
DISC HEIGHT (M)	0.00	1.00	1.30	2.00	3.00	4.00	5.00	6.00	7.00	8.00	9.00	10.00	11.00	12.00	13.00	14.00	14.60
RING COUNT	37	32	31	30	28	26	24	22	20	19	16	13	10	8	4	2	0
AGE (YEARS)	0.	5.	6.	7.	9.	11.	13.	15.	17.	18.	21.	24.	27.	29.	33.	35.	37.
SBT (CM)	0.385	0.586	0.455	0.531	0.637	0.426	0.431	0.382	0.338	0.313	0.338	0.302	0.277	0.293	0.181	0.196	0.000
RIB (CM)																	
1981	17.107	13.117	12.710	12.094	11.626	10.814	10.378	9.532	9.319	8.121	7.090	6.075	4.589	3.004	1.277	0.491	0.000
1980	16.613	12.629	12.297	11.731	11.285	10.525	10.048	9.094	8.791	7.695	6.564	5.541	4.075	2.588	0.886	0.131	0.000
1979	16.140	12.339	12.058	11.502	11.044	10.303	9.786	8.799	8.526	7.422	6.245	5.079	3.551	2.220	0.565	0.000	0.000
1978	15.794	12.130	11.821	11.258	10.764	10.064	9.493	8.439	8.271	7.151	5.728	4.563	2.991	1.761	0.149	0.000	0.000
1977	15.113	11.888	11.582	11.042	10.544	9.867	9.256	8.124	7.926	6.705	5.201	3.952	2.396	1.294	0.000	0.000	0.000
1976	14.674	11.748	11.493	10.901	10.408	9.707	9.082	7.891	7.701	6.380	4.790	3.633	1.979	0.867	0.000	0.000	0.000
1975	14.184	11.555	11.288	10.710	10.203	9.527	8.882	7.689	7.449	6.141	4.476	3.300	1.583	0.520	0.000	0.000	0.000
1974	13.601	11.154	10.920	10.290	9.842	9.191	8.512	7.267	6.905	5.677	3.948	2.839	1.152	0.189	0.000	0.000	0.000
1973	13.057	10.895	10.577	9.987	9.514	8.868	8.116	6.878	6.318	5.267	3.478	2.428	0.698	0.000	0.000	0.000	0.000
1972	12.598	10.602	10.272	9.656	9.090	8.490	7.623	6.343	5.721	4.724	3.049	1.902	0.230	0.000	0.000	0.000	0.000
1971	12.253	10.253	9.882	9.232	8.728	7.995	7.032	5.694	4.929	4.029	2.453	1.388	0.000	0.000	0.000	0.000	0.000
1970	11.895	9.864	9.560	8.897	8.338	7.471	6.535	5.226	4.347	3.347	1.859	0.937	0.000	0.000	0.000	0.000	0.000
1969	11.487	9.536	9.261	8.635	7.999	7.080	6.125	4.769	3.772	2.753	1.381	0.563	0.000	0.000	0.000	0.000	0.000
1968	11.198	9.207	8.852	8.293	7.613	6.614	5.800	4.398	3.363	2.278	0.989	0.000	0.000	0.000	0.000	0.000	0.000
1967	10.681	8.786	8.431	7.860	7.127	6.063	5.265	3.932	2.599	1.913	0.665	0.000	0.000	0.000	0.000	0.000	0.000
1966	10.145	8.528	8.144	7.605	6.809	5.767	4.921	3.599	2.064	1.645	0.515	0.000	0.000	0.000	0.000	0.000	0.000
1965	9.718	8.095	7.777	7.178	6.348	5.281	4.359	3.149	1.597	1.104	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1964	8.990	7.492	7.132	6.588	5.686	4.763	3.785	2.683	1.224	0.661	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1963	8.262	7.036	6.734	6.233	5.215	4.190	3.240	2.188	0.696	0.420	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1962	7.870	6.446	6.219	5.562	4.638	3.572	2.509	1.497	0.405	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1961	7.282	5.644	5.596	4.786	4.010	2.863	1.811	0.838	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1960	6.700	5.097	5.101	4.214	3.430	2.223	1.235	0.411	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1959	5.833	4.408	4.518	3.526	2.763	1.565	0.747	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1958	5.163	3.926	4.011	2.934	2.195	1.075	0.355	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1957	4.575	3.363	3.480	2.335	1.650	0.654	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1956	3.989	2.868	2.963	1.779	1.196	0.308	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1955	3.504	2.444	2.464	1.361	0.759	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1954	3.068	1.995	1.986	0.905	0.247	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1953	2.463	1.506	1.515	0.561	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1952	2.012	1.119	1.058	0.303	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1951	1.666	0.706	0.595	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1950	1.182	0.254	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1949	0.792	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1948	0.612	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1947	0.505	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1946	0.387	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1945	0.168	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

APPENDIX B-2 STEM.DAT file. Continued.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

MAI AND PAI FOR HEIGHT (METRES)

INTERVAL	AGE	HEIGHT	MAI	PAI
1.0000	37.0000	14.6000	0.3946	0.3000
1.0000	36.0000	14.3000	0.3972	0.3000
1.0000	35.0000	14.0000	0.4000	0.5000
1.0000	34.0000	13.5000	0.3971	0.5000
1.0000	33.0000	13.0000	0.3939	0.2500
1.0000	32.0000	12.7500	0.3984	0.2500
1.0000	31.0000	12.5000	0.4032	0.2500
1.0000	30.0000	12.2500	0.4083	0.2500
1.0000	29.0000	12.0000	0.4138	0.5000
1.0000	28.0000	11.5000	0.4107	0.5000
1.0000	27.0000	11.0000	0.4074	0.3333
1.0000	26.0000	10.6667	0.4103	0.3333
1.0000	25.0000	10.3333	0.4133	0.3333
1.0000	24.0000	10.0000	0.4167	0.3333
1.0000	23.0000	9.6667	0.4203	0.3333
1.0000	22.0000	9.3333	0.4242	0.3333
1.0000	21.0000	9.0000	0.4286	0.3333
1.0000	20.0000	8.6667	0.4333	0.3333
1.0000	19.0000	8.3333	0.4386	0.3333
1.0000	18.0000	8.0000	0.4444	1.0000
1.0000	17.0000	7.0000	0.4118	0.5000
1.0000	16.0000	6.5000	0.4063	0.5000
1.0000	15.0000	6.0000	0.4000	0.5000
1.0000	14.0000	5.5000	0.3929	0.5000

APPENDIX B-2 STEM.DAT file. Continued.

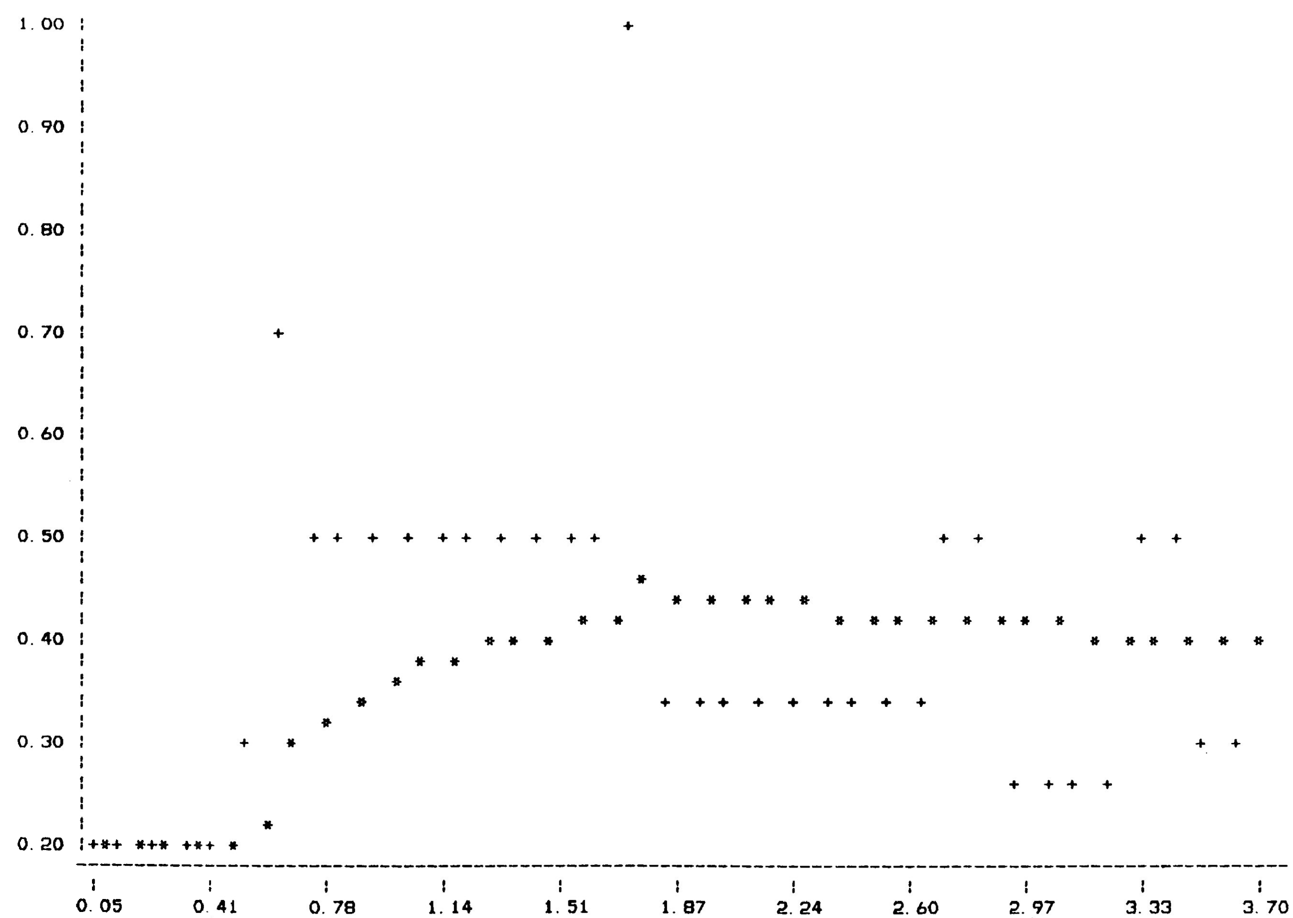
1.0000	13.0000	5.0000	0.3846	0.5000
1.0000	12.0000	4.5000	0.3750	0.5000
1.0000	11.0000	4.0000	0.3636	0.5000
1.0000	10.0000	3.5000	0.3500	0.5000
1.0000	9.0000	3.0000	0.3333	0.5000
1.0000	8.0000	2.5000	0.3125	0.5000
1.0000	7.0000	2.0000	0.2857	0.7000
1.0000	6.0000	1.3000	0.2167	0.3000
1.0000	5.0000	1.0000	0.2000	0.2000
1.0000	4.0000	0.8000	0.2000	0.2000
1.0000	3.0000	0.6000	0.2000	0.2000
1.0000	2.0000	0.4000	0.2000	0.2000
1.0000	1.0000	0.2000	0.2000	0.2000

APPENDIX B-2 STEM.DAT file. Continued.

MAI AND PAI FOR HEIGHT (METRES)

NO. OF POINTS = 74

	PLOT SIZE (PRINT POSITIONS)	MIN VAL	MAX VAL	SCALE FACTOR
HORIZ. (X)	100.0	0.5000E+00	0.3700E+02	10.0000
VERT. (Y)	40.0	0.2000E+00	0.1000E+01	1.0000



APPENDIX B-2 STEM.DAT file. Continued.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

MAI AND PAI FOR DBH (CENTIMETRES)

INTERVAL	AGE	DBH	MAI	PAI
1.0000	37.0000	25.4200	0.6870	
				0.8260
1.0000	36.0000	24.5940	0.6832	
				0.4780
1.0000	35.0000	24.1160	0.6890	
				0.4740
1.0000	34.0000	23.6420	0.6954	
				0.4780
1.0000	33.0000	23.1640	0.7019	
				0.1780
1.0000	32.0000	22.9860	0.7183	
				0.4100
1.0000	31.0000	22.5760	0.7283	
				0.7360
1.0000	30.0000	21.8400	0.7280	
				0.6860
1.0000	29.0000	21.1540	0.7294	
				0.6100
1.0000	28.0000	20.5440	0.7337	
				0.7800
1.0000	27.0000	19.7640	0.7320	
				0.6440
1.0000	26.0000	19.1200	0.7354	
				0.5980
1.0000	25.0000	18.5220	0.7409	
				0.8180
1.0000	24.0000	17.7040	0.7377	
				0.8420
1.0000	23.0000	16.8620	0.7331	
				0.5740
1.0000	22.0000	16.2880	0.7404	
				0.7340
1.0000	21.0000	15.5540	0.7407	
				1.2900
1.0000	20.0000	14.2640	0.7132	
				0.7960
1.0000	19.0000	13.4680	0.7088	
				1.0300
1.0000	18.0000	12.4380	0.6910	
				1.2460
1.0000	17.0000	11.1920	0.6584	
				0.9900
1.0000	16.0000	10.2020	0.6376	
				1.1660
1.0000	15.0000	9.0360	0.6024	
				1.0140
1.0000	14.0000	8.0220	0.5730	
				1.0620

APPENDIX B-2 STEM.DAT file. Continued.

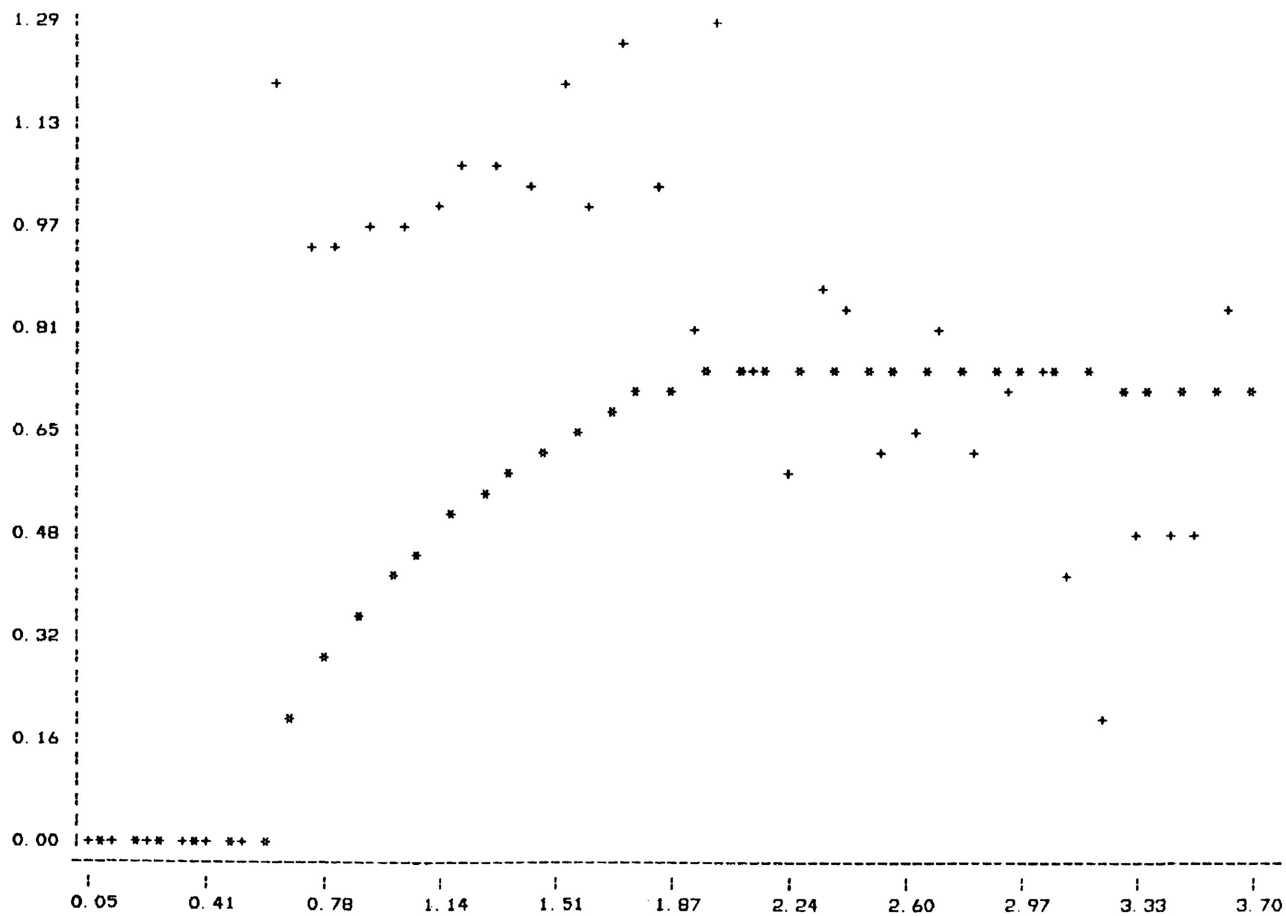
1.0000	13.0000	6.9600	0.5354	1.0340
1.0000	12.0000	5.9260	0.4938	0.9980
1.0000	11.0000	4.9280	0.4480	0.9560
1.0000	10.0000	3.9720	0.3972	0.9420
1.0000	9.0000	3.0300	0.3367	0.9140
1.0000	8.0000	2.1160	0.2645	0.9260
1.0000	7.0000	1.1900	0.1700	1.1900
1.0000	6.0000	0.0000	0.0000	0.0000
1.0000	5.0000	0.0000	0.0000	0.0000
1.0000	4.0000	0.0000	0.0000	0.0000
1.0000	3.0000	0.0000	0.0000	0.0000
1.0000	2.0000	0.0000	0.0000	0.0000
1.0000	1.0000	0.0000	0.0000	0.0000

APPENDIX B-2 STEM.DAT file. Continued.

MAI AND PAI FOR DBH (CENTIMETRES)

NO. OF POINTS = 74

	PLOT SIZE (PRINT POSITIONS)	MIN VAL	MAX VAL	SCALE FACTOR
HORIZ. (X)	100.0	0.5000E+00	0.3700E+02	10.0000
VERT. (Y)	40.0	0.0000E+00	0.1290E+01	1.0000



APPENDIX B-2 STEM.DAT file. Continued.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

MAI AND PAI FOR BASAL AREA (SQUARE DECIMETRES)

INTERVAL	AGE	BASAL AREA	MAI	PAI
1.0000	37.0000	5.0751	0.1372	
				0.3245
1.0000	36.0000	4.7506	0.1320	
				0.1829
1.0000	35.0000	4.5677	0.1305	
				0.1778
1.0000	34.0000	4.3899	0.1291	
				0.1757
1.0000	33.0000	4.2142	0.1277	
				0.0645
1.0000	32.0000	4.1497	0.1297	
				0.1467
1.0000	31.0000	4.0030	0.1291	
				0.2567
1.0000	30.0000	3.7462	0.1249	
				0.2316
1.0000	29.0000	3.5146	0.1212	
				0.1998
1.0000	28.0000	3.3148	0.1184	
				0.2469
1.0000	27.0000	3.0679	0.1136	
				0.1967
1.0000	26.0000	2.8712	0.1104	
				0.1768
1.0000	25.0000	2.6944	0.1078	
				0.2327
1.0000	24.0000	2.4617	0.1026	
				0.2286
1.0000	23.0000	2.2331	0.0971	
				0.1494
1.0000	22.0000	2.0837	0.0947	
				0.1836
1.0000	21.0000	1.9001	0.0905	
				0.3021
1.0000	20.0000	1.5980	0.0799	
				0.1734
1.0000	19.0000	1.4246	0.0750	
				0.2096
1.0000	18.0000	1.2150	0.0675	
				0.2312
1.0000	17.0000	0.9838	0.0579	
				0.1663
1.0000	16.0000	0.8175	0.0511	
				0.1762
1.0000	15.0000	0.6413	0.0428	
				0.1358
1.0000	14.0000	0.5054	0.0361	
				0.1250

APPENDIX B-2 STEM.DAT file. Continued.

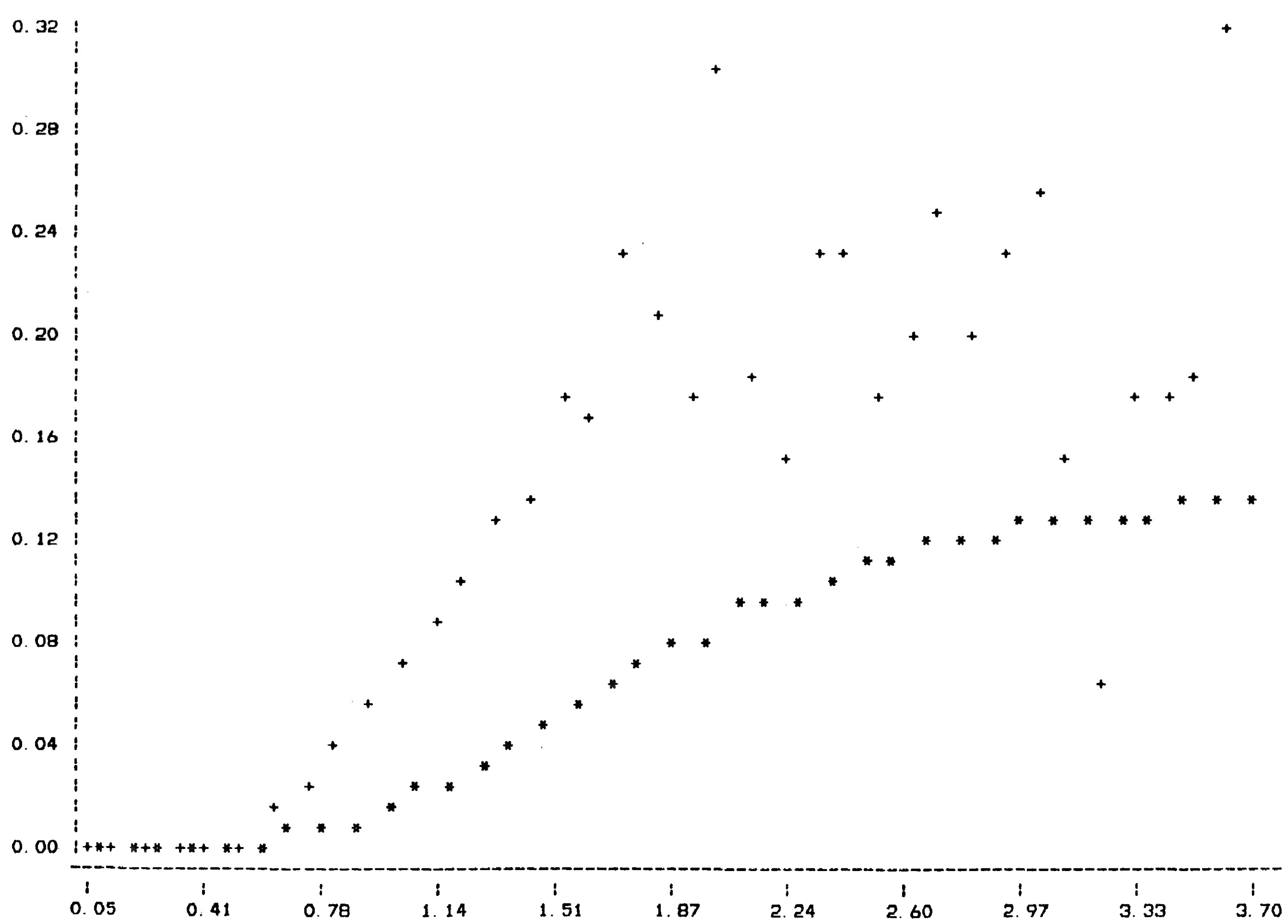
1.0000	13.0000	0.3805	0.0293	0.1046
1.0000	12.0000	0.2758	0.0230	0.0851
1.0000	11.0000	0.1907	0.0173	0.0668
1.0000	10.0000	0.1239	0.0124	0.0518
1.0000	9.0000	0.0721	0.0080	0.0369
1.0000	8.0000	0.0352	0.0044	0.0240
1.0000	7.0000	0.0111	0.0016	0.0111
1.0000	6.0000	0.0000	0.0000	0.0000
1.0000	5.0000	0.0000	0.0000	0.0000
1.0000	4.0000	0.0000	0.0000	0.0000
1.0000	3.0000	0.0000	0.0000	0.0000
1.0000	2.0000	0.0000	0.0000	0.0000
1.0000	1.0000	0.0000	0.0000	0.0000

APPENDIX B-2 STEM.DAT file. Continued.

MAI AND PAI FOR BASAL AREA (SQUARE DECIMETRES)

NO. OF POINTS = 74

	PLOT SIZE (PRINT POSITIONS)	MIN VAL	MAX VAL	SCALE FACTOR
HORIZ. (X)	100.0	0.5000E+00	0.3700E+02	10.0000
VERT. (Y)	40.0	0.0000E+00	0.3245E+00	1.0000



APPENDIX B-2 STEM.DAT file. Continued.

PLOT # 1 TREE # 1 SPECIES CODE SW NUMBER OF LEADERS 1

MAI AND PAI FOR VOLUME (CUBIC DECIMETRES)

INTERVAL	AGE	VOLUME	MAI	PAI
1.0000	37.0000	372.5313	10.0684	
				29.9648
1.0000	36.0000	342.5665	9.5157	
				20.2055
1.0000	35.0000	322.3610	9.2103	
				20.2377
1.0000	34.0000	302.1233	8.8860	
				21.0004
1.0000	33.0000	281.1229	8.5189	
				13.3157
1.0000	32.0000	267.8071	8.3690	
				13.8728
1.0000	31.0000	253.9343	8.1914	
				23.1862
1.0000	30.0000	230.7481	7.6916	
				19.7629
1.0000	29.0000	210.9852	7.2754	
				20.7683
1.0000	28.0000	190.2169	6.7935	
				22.1063
1.0000	27.0000	168.1106	6.2263	
				18.3784
1.0000	26.0000	149.7322	5.7589	
				14.7564
1.0000	25.0000	134.9758	5.3990	
				13.2085
1.0000	24.0000	121.7673	5.0736	
				15.9616
1.0000	23.0000	105.8057	4.6002	
				9.9905
1.0000	22.0000	95.8153	4.3552	
				12.6502
1.0000	21.0000	83.1651	3.9602	
				14.7968
1.0000	20.0000	68.3683	3.4184	
				10.9888
1.0000	19.0000	57.3795	3.0200	
				11.3298
1.0000	18.0000	46.0497	2.5583	
				11.3018
1.0000	17.0000	34.7479	2.0440	
				7.7947
1.0000	16.0000	26.9532	1.6846	
				7.7715
1.0000	15.0000	19.1817	1.2788	
				5.1147
1.0000	14.0000	14.0670	1.0048	
				4.1217

APPENDIX B-2 STEM.DAT file. Continued.

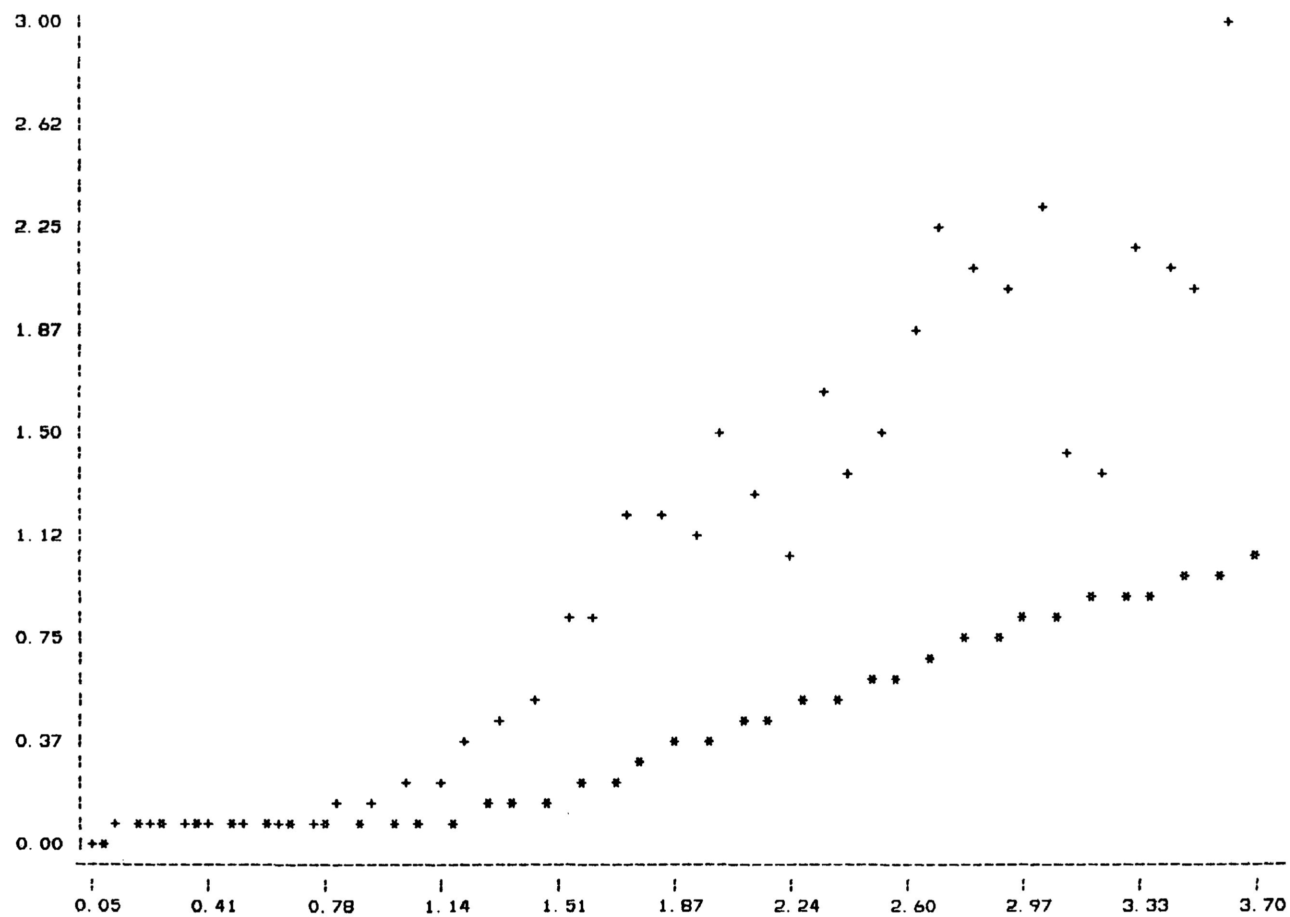
1.0000	13.0000	9.9452	0.7650	3.0701
1.0000	12.0000	6.8751	0.5729	2.0974
1.0000	11.0000	4.7777	0.4343	1.6338
1.0000	10.0000	3.1439	0.3144	1.2833
1.0000	9.0000	1.8606	0.2067	0.7759
1.0000	8.0000	1.0847	0.1356	0.4913
1.0000	7.0000	0.5934	0.0848	0.3607
1.0000	6.0000	0.2326	0.0388	0.1341
1.0000	5.0000	0.0985	0.0197	0.0515
1.0000	4.0000	0.0471	0.0118	0.0230
1.0000	3.0000	0.0240	0.0080	0.0146
1.0000	2.0000	0.0094	0.0047	0.0085
1.0000	1.0000	0.0009	0.0009	0.0009

APPENDIX B-2 STEM.DAT file. Continued.

MAI AND PAI FOR VOLUME (CUBIC DECIMETRES)

NO. OF POINTS = 74

	PLOT SIZE (PRINT POSITIONS)	MIN VAL	MAX VAL	SCALE FACTOR
HORIZ. (X)	100.0	0.5000E+00	0.3700E+02	10.0000
VERT. (Y)	40.0	0.8867E-03	0.2996E+02	10.0000



APPENDIX B-3

STEM Program Listing

```
************************************************************************
*                                                                      *
*                       PROGRAM STEM.FOR                               *
*                                                                      *
*             STEM.FOR ANALYZES STEM ANALYSIS DATA PRODUCED BY         *
*        THE HOLMAN DIGIMICROMETER.  ANY NUMBER OF TREES MAY BE        *
*        PROCESSED AT ONE TIME.  RAW DATA IS STORED IN TWO FILES.      *
*        TREE.DAT CONTAINS PERTINENT INFORMATION REQUIRED TO PROCESS   *
*        EACH TREE.  PLOT.DAT CONTAINS ALL RINGWIDTH DATA FOR ALL      *
*        THE DISCS TAKEN FROM EACH TREE.  THE ORDER OF TREES IN        *
*        TREE.DAT MUST CORRESPOND WITH THE DISC DATA IN PLOT.DAT.      *
*        DISC DATA MUST BE ORGANIZED WITH DISCS IN CONSECUTIVE AND     *
*        ASCENDING ORDER (IE BASAL DISC TO TIP DISC).                  *
*             STEM.FOR SHOULD BE USED IN CONJUNCTION WITH THE          *
*        ALGORITHM DUFFNO.FOR, TO ENSURE THAT THE RAW DATA IS ERROR    *
*        FREE.                                                         *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD       *
*        FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE    *
*        VMS OPERATING SYSTEM.                                         *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM    *
*        THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY  *
*        ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS    *
*        RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE    *
*        FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR    *
*        H. GARY MURCHISON AT THE SAME ADDRESS.                        *
*                                                                      *
*        AUTHOR:        JOANNE KAVANAGH                                *
*                       SCHOOL OF FORESTRY                             *
*                       LAKEHEAD UNIVERSITY                            *
*                       THUNDER BAY, ONTARIO, P7B 5E1                  *
*                          OCTOBER,  1981                              *
*                                                                      *
*                       COPYRIGHT APPLIED FOR                          *
*                                                                      *
************************************************************************
*                                                                      *
*                       VARIABLE IDENTIFICATION                        *
*                                                                      *
*        PN = PLOT NUMBER                                              *
*        TN = TREE NUMBER                                              *
*        SC = SPECIES CODE; ACCORDING TO OMNR FOREST INVENTORY CODES   *
*        AGE = TOTAL TREE AGE IN YEARS                                 *
*        YR = YEAR CUT                                                 *
*        TOTHT = TOTAL TREE HEIGHT IN METRES                           *
*        BOLEHT = TOTAL BOLE HEIGHT IN METRES TO SPLIT                 *
*        NLEAD = NUMBER OF LEADERS (5 MAXIMUM ALLOWED)                 *
*        NDISCS = TOTAL NUMBER OF TREE DISCS                           *
*        ITD = NDISCS + 1                                              *
*        NBDISC = TOTAL NUMBER OF BOLE DISCS                           *
*        NDISCM = TOTAL NUMBER OF DISCS IN MAIN STEM (MULTIPLE         *
*             TOP TREES)                                               *
*        NDISCL(K) = TOTAL NUMBER OF DISCS PER LEADER                  *
*        ITD(K) = NDISCL(K) + 1                                        *
*        LLNGTH(K) = LEADER LENGTH IN METRES                           *
*        DN(J) = DISC NUMBER                                           *
```

```
*       RC(J) = RING COUNT ON EACH DISC                                     *
*       DNL(JJ,K) = DISC NUMBER FOR MULTIPLE TOPS                            *
*       RCL(JJ,K) = RING COUNT ON EACH DISC FOR MULTIPLE TOPS                *
*       NTC = NUMBER OF TAPER CURVES                                         *
*       NTC(K) = NUMBER OF TAPER CURVES FOR EACH LEADER                      *
*       DHT(J) = DISC HEIGHT IN METRES                                       *
*       AVERAD(J) = AVERAGE RADIUS OF DISC                                   *
*       SBT(J) = SINGLE BARK THICKNESS OF DISC                               *
*       RW(I,J) = RING WIDTHS BY YEAR PER DISC                               *
*       RAD(I,J) = RADIUS MEASUREMENTS BY YEAR PER DISC                      *
*       TYRS(J) = TOTAL YEARS REQUIRED TO ATTAIN DISC HEIGHT                 *
*       BOLT(J) = BOLT LENGTH BETWEEN CONSECUTIVE DISCS                      *
*       DHTL(JJ,K) = DISC HEIGHT OF DISCS ON MULTIPLE TOPS                   *
*       AVRADL(JJ,K) = AVERAGE RADIUS OF DISC ON MULTIPLE TOPS               *
*       SBTL(JJ,K) = SINGLE BARK THICKNESS OF DISCS ON MULTIPLE TOPS         *
*       RWL(I,JJ,K) = RING WIDTHS BY YEAR PER DISC ON MULTIPLE TOPS          *
*       RADL(I,JJ,K) = RADIUS MEASUREMENTS BY YEAR PER DISC ON               *
*               MULTIPLE TOPS                                                *
*       TYRSL(JJ,K) = TOTAL YEARS REQUIRED TO ATTAIN DISC HEIGHT ON          *
*               MULTIPLE TOPS                                                *
*       BOLTL(JJ,K) = BOLT LENGTH BETWEEN CONSECUTIVE DISCS ON               *
*               MULTIPLE TOPS                                                *
*       MAI(I) = MEAN ANNUAL INCREMENT                                       *
*       PAI(I) = PERIODIC ANNUAL INCREMENT                                   *
*       CVHT(I) = CURVE HEIGHTS FOR TAPER CURVES                             *
*       DBH(I) = DIAMETER AT BREAST HEIGHT (1.3 METRES)                      *
*       BA(I) = BASAL AREA BASED ON DBH FOR TAPER CURVES                     *
*       VOL(I) = VOLUME BASED ON SMALIAN'S FORMULA FOR TAPER CURVES          *
*       CVAGE(I) = AGES USED TO PLOT PAI                                     *
*       MIDAGE(I) = AGES USED TO PLOT MAI                                    *
*       INT = TIME INTERVAL BETWEEN TAPER CURVES (YEARS)                     *
*       CVAGEL(I,K) = AGES USED TO PLOT PAI FOR MULTIPLE TOPS                *
*       MDAGEL(I,K) = AGES USED TO PLOT MAI FOR MULTIPLE TOPS                *
*       VOLL(I,K) = VOLUME BASED ON SMALIAN'S FORMULA FOR TAPER              *
*               CURVES ON MULTIPLE TOPS                                      *
*       CVHTL(I,K) = CURVE HEIGHTS FOR TAPER CURVES IN MULTIPLE TOPS         *
*       BLL(K) = BOLT LENGTH BETWEEN MAIN BOLE AT THE FORK AND THE           *
*               FIRST DISC ON THE LEADER                                     *
*       BLM = BOLT LENGTH BETWEEN MAIN BOLE AT THE FORK AND THE LAST         *
*               DISC ON THE BOLE OF A MULTIPLE TOP TREE                      *
*       TIPLEN = INTERMEDIATE VALUE FOR VOLUME CALCULATION                   *
*       TIPVOL = INTERMEDIATE VALUE FOR VOLUME CALCULATION                   *
*       NPP(I) = NUMBER OF PLOTTED POINTS FOR THE MAIN BOLE TAPER            *
*               CURVES, INCLUDING THE MAIN LEADER                            *
*       NPP(I,K) = THE NUMBER OF PLOTTED POINTS FOR THE MULTIPLE             *
*               LEADER TAPER CURVES                                          *
*       CODE = ALPHABETIC SPECIES CODE                                       *
*       I = 1 TO A MAXIMUM OF 300 YEARS                                      *
*       J = 1 TO A MAXIMUM OF 60 DISCS                                       *
*       JJ = 1 TO A MAXIMUM OF 25 DISCS ON EACH LEADER                       *
*       K = 1 TO A MAXIMUM OF 5 MULTIPLE TOPS                                *
*                                                                            *
******************************************************************************
*                                                                            *
*                       STORAGE ALLOCATION                                   *
```

```
*                                                                             *
        REAL AGE, TOTHT, BOLEHT, DHT(60), AVERAD(60), SBT(60), RW(300, 60),
        1RAD(300, 60), TYRS(60), BOLT(60), DHTL(25, 5), AVRADL(25, 5), SBTL(25,
        25), RWL(300, 25, 5), RADL(300, 25, 5), TYRSL(25, 5), BOLTL(25, 5),
        3LLNGTH(5), MAI(300), PAI(300), CVHT(300), DBH(300), BA(300), VOL(300)
        4, CVAGE(300), MIDAGE(300), INT, CVAGEL(300, 5), MDAGEL(300, 5), VOLL
        5(300, 5), CVHTL(300, 5), BLL(5), TIPLEN, TIPVOL, BLM
        INTEGER PN, TN, SC, YR, NLEAD, NDISCS, NBDISC, NDISCL(5), DN(60), RC
        1(60), ITD, DNL(25, 5), RCL(25, 5), ITDL(5), NTC, NPP(300), CODE(75)
        2, NTCL(5), NPPL(300, 5), NDISCM
        COMMON/OUTP/PN, TN, SC, AGE, YR, NLEAD
        COMMON/OUTP1/DHT, BOLT, RC, TYRS, SBT, RAD, DN
        COMMON/OUTP2/DHTL, BOLTL, RCL, TYRSL, SBTL, RADL, DNL
        COMMON/PAIMAI/INT, NTC, MIDAGE, CVAGE
*                                                                             *
*******************************************************************************
*                                                                             *
*                         DATA SPECIFICATION                                  *
*                                                                             *
        DATA INT/1.0/
        DATA CODE/'PW', 'PR', 'PJ', 'PS', '  ', '  ', '  ', '  ', '  ', '  ', '  ',
        2'  ', 'SW', 'SB', '  ', '  ', '  ', '  ', '  ', '  ', 'HE', 'BF', '  ', '  ',
        3'  ', 'CE', 'TA', '  ', '  ', '  ', '  ', 'MH', '  ', 'MR', 'MS', '  ',
        4'  ', '  ', 'BY', 'BW', '  ', 'OW', 'OR', '  ', '  ', 'BE', 'AB', 'AW',
        5'  ', '  ', 'EM', '  ', 'BD', '  ', '  ', '  ', '  ', 'ID', '  ', 'CB',
        6'  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ',
        7'  ', '  ', '  ', '  ', 'PO'/
*                                                                             *
*******************************************************************************
*                                                                             *
*                         OPEN I/O UNITS                                      *
*                                                                             *
        OPEN(UNIT=20, FILE='TREE.DAT', STATUS='OLD')
        OPEN(UNIT=21, FILE='PLOT.DAT', STATUS='OLD')
        OPEN(UNIT=22, FILE='STEM.DAT', STATUS='NEW')
*                                                                             *
*******************************************************************************
*                                                                             *
*       START LOOP FOR STEM ANALYSIS; ONE TREE AT A TIME                      *
*                                                                             *
*******************************************************************************
*                                                                             *
*                         INITIALIZE VARIABLES                                *
*                                                                             *
10      CONTINUE
        PN=0
        TN=0
        SC=0
        AGE=0.0
        YR=0
        TOTHT=0.0
        BOLEHT=0.0
        NLEAD=0
        NDISCS=0
        NBDISC=0
```

```
      ITD=0
      DO 20 I=1,5
      NDISCL(I)=0
      LLNGTH(I)=0.0
      ITDL(I)=0
20    CONTINUE
      DO 30 J=1,60
      DHT(J)=0.0
      AVERAD(J)=0.0
      SBT(J)=0.0
      TYRS(J)=0.0
      BOLT(J)=0.0
      RC(J)=0
      DN(J)=0
      DO 40 I=1,300
      RW(I,J)=0.0
      RAD(I,J)=0.0
40    CONTINUE
30    CONTINUE
      DO 50 J=1,25
      DO 60 K=1,5
      DHTL(J,K)=0.0
      AVRADL(J,K)=0.0
      SBTL(J,K)=0.0
      TYRSL(J,K)=0.0
      BOLTL(J,K)=0.0
      DNL(J,K)=0
      RCL(J,K)=0
      DO 70 I=1,300
      RWL(I,J,K)=0.0
      RADL(I,J,K)=0.0
70    CONTINUE
60    CONTINUE
50    CONTINUE
      DO 80 I=1,300
      DBH(I)=0.0
      BA(I)=0.0
      VOL(I)=0.0
      MAI(I)=0.0
      PAI(I)=0.0
      CVHT(I)=0.0
      CVAGE(I)=0.0
      MIDAGE(I)=0.0
80    CONTINUE
*                                                                    *
**********************************************************************
*                                                                    *
*     READ TREE HEADER FROM FILE TREE.DAT                            *
*     WHEN END OF FILE OCCURS, GO TO 100                             *
*                                                                    *
      READ(20,90,END=100)PN,TN,SC,AGE,YR,TOTHT,BOLEHT,NLEAD,NDISCS,
     1NBDISC,(NDISCL(I),I=1,5),(LLNGTH(I),I=1,5)
90    FORMAT(I2,I3,I2,F3.0,I5,2F5.2,I2,2I3,5I2,5F4.2)
*                                                                    *
*     TEST FOR A MULTIPLE LEADER TREE.  IF THERE IS A MULTIPLE TOP   *
```

```
*       GO TO 110                                                         *
*                                                                         *
        IF(NLEAD.GT.1) GO TO 110
*                                                                         *
*       READ DISC HEADERS WITH ACCOMPANYING RING DATA FOR SINGLE          *
*       LEADER TREES FROM FILE PLOT.DAT                                   *
*                                                                         *
        DO 120 J=1,NDISCS
        READ(21,130)DN(J),DHT(J),RC(J),AVERAD(J),SBTT
130     FORMAT(12X,I4,4X,F4.2,I3,12X,F9.2,F7.2)
*                                                                         *
*       CONVERT SINGLE BARK THICKNESS MEASUREMENT UNITS FROM              *
*       MILLIMETRES TO CENTIMETRES                                        *
*                                                                         *
        SBT(J)=SBTT/10.0
        READ(21,140)(RW(I,J),I=1,RC(J))
140     FORMAT(4X,10F7.2)
120     CONTINUE
*                                                                         *
*       CREATE TIP DISC FOR THE TREE                                      *
*                                                                         *
        ITD=NDISCS+1
        DHT(ITD)=TOTHT
        BOLT(ITD)=TOTHT-DHT(NDISCS)
        RC(ITD)=0
        SBT(ITD)=0.0
        RAD(1,ITD)=0.0
*                                                                         *
*       CONVERT RING WIDTH MEASUREMENTS TO RADIUS MEASUREMENTS, AND        *
*       CHANGE MEASUREMENT UNITS FROM MILLIMETRES TO CENTIMETRES          *
*                                                                         *
        DO 150 J=1,NDISCS
        RAD(1,J)=AVERAD(J)/10.0
        NN=RC(J)
        DO 160 I=2,NN
        II=I-1
        RAD(I,J)=RAD(II,J)-(RW(II,J)/10.0)
160     CONTINUE
150     CONTINUE
*                                                                         *
*       CALCULATE THE NUMBER OF YEARS REQUIRED TO ATTAIN DISC HEIGHT      *
*                                                                         *
        DO 170 J=1,ITD
        TYRS(J)=AGE-(RC(J)*1.0)
170     CONTINUE
*                                                                         *
*       CALCULATE BOLT LENGTHS BETWEEN CONSECUTIVE DISCS                  *
*                                                                         *
        BOLT(1)=DHT(1)-0.0
        DO 180 J=2,NDISCS
        JJ=J-1
        BOLT(J)=DHT(J)-DHT(JJ)
180     CONTINUE
*                                                                         *
*       PRINT OUT ORIGINAL MEASUREMENTS IN TABULAR FORMAT                 *
```

```
*                                                                          *
        CALL OUTPT1(ITD)
*                                                                          *
*       CALCULATE THE TOTAL NUMBER OF TAPER CURVES FOR THE TREE BASED      *
*       ON THE SPECIFIED TIME INTERVAL BETWEEN TAPER CURVES                *
*                                                                          *
        NTC=AGE/INT
        X=(AGE/INT)-NTC
        IF(X.GE.0.1) NTC=NTC+1
*                                                                          *
*       CALCULATE AGES OF TAPER CURVES BASED ON THE SPECIFIED              *
*       TIME INTERVAL                                                      *
*                                                                          *
        A=AGE
        CVAGE(1)=AGE
        DO 190 I=2,NTC
        A=A-INT
        CVAGE(I)=A
190     CONTINUE
        NN=NTC-1
        DO 200 I=1,NN
        MIDAGE(I)=CVAGE(I)-(INT/2.0)
200     CONTINUE
        MIDAGE(NTC)=CVAGE(NTC)/2.0
*                                                                          *
*       CALCULATE TOTAL HEIGHTS OF TAPER CURVES                            *
*                                                                          *
        CVHT(1)=DHT(ITD)
        DO 210 I=2,NTC
        DO 220 K=1,ITD
        J=(1-K)+ITD
        IF(TYRS(J).LE.CVAGE(I)) GO TO 230
        IF(J.NE.1) GO TO 220
        IF(DHT(J).EQ.0.0) GO TO 230
        GO TO 209
220     CONTINUE
230     CONTINUE
        JJ=J+1
        RR=RC(JJ)-RC(J)
        IF(RR.GE.0.0) GO TO 235
        SLOPE=(DHT(JJ)-DHT(J))/RR
        X=(TYRS(J)-CVAGE(I))*SLOPE
        CVHT(I)=DHT(J)+X
        GO TO 210
209     CONTINUE
        SLOPE=(DHT(J)-0.0)/(RC(J)-AGE)
        X=(0.0-CVAGE(I))*SLOPE
        CVHT(I)=0.0+X
        GO TO 210
235     CONTINUE
        CVHT(I)=DHT(JJ)
210     CONTINUE
*                                                                          *
*       CALCULATE NUMBER OF PLOTTED POINTS ON EACH TAPER CURVE             *
*                                                                          *
```

```
      DO 240 I=1,NTC
      DO 250 J=1,ITD
      K=(ITD+1)-J
      X=CVHT(I)-DHT(K)
      IF(X.GE.0.0) GO TO 260
250   CONTINUE
260   CONTINUE
      IF(X.LE.0.009) GO TO 270
      NPP(I)=K+1
      GO TO 240
270   CONTINUE
      NPP(I)=K
240   CONTINUE
*                                                                  *
*     PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI          *
*                                                                  *
      WRITE(22,280)PN,TN,CODE(SC),NLEAD
280   FORMAT('1',/////,T31,'PLOT #',I3,5X,'TREE #',I3,5X,'SPECIES
     2CODE ',A2,5X,'NUMBER OF LEADERS',I2)
      WRITE(22,290)
290   FORMAT(' ',//,12X,'MAI AND PAI FOR HEIGHT (METRES)')
      WRITE(22,300)
300   FORMAT(' ',//,12X,'INTERVAL',4X,'AGE',5X,'HEIGHT',6X,'MAI',7X,
     2'PAI',/)
*                                                                  *
*     CALCULATE MAI AND PAI FOR HEIGHT                             *
*                                                                  *
      CALL MAIPAI(CVHT,1)
*                                                                  *
*     DETERMINE DBH VALUES FOR EACH TAPER CURVE                    *
*                                                                  *
      DO 310 I=1,NTC
      NPP1=NPP(I)+1
      DO 320 J=1,NPP1
      K=J-1
      IF(DHT(J).GE.1.3) GO TO 330
320   CONTINUE
330   CONTINUE
      IF(DHT(J).EQ.1.3) GO TO 309
      IF(K.EQ.0) GO TO 340
      IF(CVHT(I).LT.1.3) GO TO 311
      SLOPE=(RAD(I,J)-RAD(I,K))/(DHT(J)-DHT(K))
      X=(1.3-DHT(K))*SLOPE
      DBH(I)=(RAD(I,K)+X)*2.0
      IF(DBH(I).LE.0.0) DBH(I)=0.0
      IF(DBH(1).LE.0.0) GO TO 341
      GO TO 310
309   CONTINUE
      DBH(I)=RAD(I,J)*2.0
      IF(DBH(1).LE.0.0) GO TO 341
      GO TO 310
311   CONTINUE
      DBH(I)=0.0
      IF(DBH(1).LE.0.0) GO TO 341
310   CONTINUE
```

```
        GO TO 350
*                                                                      *
*       PRINT WARNING MESSAGE IF THERE ARE NO MEASUREMENTS RECORDED    *
*       BELOW BREAST HEIGHT                                            *
*                                                                      *
340     CONTINUE
        WRITE(22,360)
360     FORMAT('1',//,10X,'UNABLE TO CALCULATE MAI AND PAI FOR DBH AS
       2THERE ARE NO RECORDED MEASUREMENTS BELOW BREAST HEIGHT')
        GO TO 370
341     CONTINUE
        WRITE(22,342)
342     FORMAT('1',//,10X,'UNABLE TO CALCULATE DBH BECAUSE THE TREE HAS
       1NOT ATTAINED BREAST HEIGHT')
        GO TO 370
350     CONTINUE
*                                                                      *
*       PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI            *
*                                                                      *
        WRITE(22,280)PN,TN,CODE(SC),NLEAD
        WRITE(22,380)
380     FORMAT(' ',//,12X,'MAI AND PAI FOR DBH (CENTIMETRES)')
        WRITE(22,390)
390     FORMAT(' ',//,12X,'INTERVAL',4X,'AGE',7X,'DBH',7X,'MAI',7X,'PAI'
       2,/)
*                                                                      *
*       CALCULATE MAI AND PAI FOR DBH                                  *
*                                                                      *
        CALL MAIPAI(DBH,2)
370     CONTINUE
*                                                                      *
*       CALCULATE BASAL AREA BASED ON DBH                              *
*                                                                      *
        IF(DBH(1).LE.0.0) GO TO 431
        DO 400 I=1,NTC
        BA(I)=(DBH(I)**2)*0.007854
400     CONTINUE
*                                                                      *
*       PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI            *
*                                                                      *
        WRITE(22,280)PN,TN,CODE(SC),NLEAD
        WRITE(22,410)
410     FORMAT(' ',//,12X,'MAI AND PAI FOR BASAL AREA (SQUARE
       2DECIMETRES)')
        WRITE(22,420)
420     FORMAT(' ',//,12X,'INTERVAL',4X,'AGE',3X,'BASAL AREA',4X,'MAI',
       27X,'PAI',/)
*                                                                      *
*       CALCULATE MAI AND PAI FOR BASAL AREA                           *
*                                                                      *
        CALL MAIPAI(BA,3)
*                                                                      *
*       CALCULATE TOTAL VOLUME UNDER EACH TAPER CURVE                  *
*                                                                      *
431     CONTINUE
```

```
      DO 430 I=1,NTC
      VOL(I)=0.0
      V=(RAD(I,1)**2)*2*BOLT(1)*0.15708
      VOL(I)=VOL(I)+V
      NP=NPP(I)
      DO 440 J=2,NP
      JJ=J-1
      IF(CVHT(I).LE.DHT(J)) GO TO 445
      V=((RAD(I,J)**2)+(RAD(I,JJ)**2))*BOLT(J)*0.15708
      VOL(I)=VOL(I)+V
440   CONTINUE
      GO TO 446
445   CONTINUE
      TIPLEN=CVHT(I)-DHT(JJ)
      TIPVOL=(RAD(I,JJ)**2)*TIPLEN*0.15708
      VOL(I)=VOL(I)+TIPVOL
446   CONTINUE
430   CONTINUE
*                                                                     *
*     PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI             *
*                                                                     *
      WRITE(22,280)PN,TN,CODE(SC),NLEAD
      WRITE(22,450)
450   FORMAT(' ',//,12X,'MAI AND PAI FOR VOLUME (CUBIC DECIMETRES)')
      WRITE(22,460)
460   FORMAT(' ',//,12X,'INTERVAL',4X,'AGE',5X,'VOLUME',6X,'MAI',7X,
     2'PAI',/)
*                                                                     *
*     CALCULATE MAI AND PAI FOR VOLUME                                *
*                                                                     *
      CALL MAIPAI(VOL,4)
*                                                                     *
*     END OF STEM ANALYSIS LOOP FOR SINGLE LEADER TREES; GO TO 470    *
*                                                                     *
      GO TO 470
110   CONTINUE
*                                                                     *
*     READ DISC HEADERS WITH ACCOMPANYING RING DATA FOR MULTIPLE      *
*     LEADER TREES FROM FILE PLOT.DAT                                 *
*                                                                     *
      DO 480 J=1,NBDISC
      READ(21,130)DN(J),DHT(J),RC(J),AVERAD(J),SBTT
*                                                                     *
*     CONVERT SINGLE BARK THICKNESS MEASUREMENT UNITS FROM            *
*     MILLIMETRES TO CENTIMETRES                                      *
*                                                                     *
      SBT(J)=SBTT/10.0
      READ(21,140)(RW(I,J),I=1,RC(J))
480   CONTINUE
      DO 490 K=1,NLEAD
      ND=NDISCL(K)
      IF(ND.EQ.0) GO TO 500
      DO 510 J=1,ND
      READ(21,130)DNL(J,K),DHTL(J,K),RCL(J,K),AVRADL(J,K),SBTT
      SBTL(J,K)=SBTT/10.0
```

```
      READ(21,140)(RWL(I,J,K),I=1,RCL(J,K))
510   CONTINUE
*                                                                      *
*     CREATE TIP DISC FOR EACH LEADER                                  *
*                                                                      *
500   CONTINUE
      ITDL(K)=ND+1
      J=ITDL(K)
      DHTL(J,K)=BOLEHT+LLNGTH(K)
      RCL(J,K)=0
      SBTL(J,K)=0.0
      RADL(1,J,K)=0.0
      IF(ND.EQ.0) GO TO 520
      BOLTL(J,K)=DHTL(J,K)-DHTL(ND,K)
      GO TO 490
520   CONTINUE
      BOLTL(J,K)=LLNGTH(K)
490   CONTINUE
*                                                                      *
*     CONVERT RING WIDTH MEASUREMENTS TO RADIUS MEASUREMENTS, AND      *
*     CHANGE MEASUREMENT UNITS FROM MILLIMETRES TO CENTIMETRES         *
*                                                                      *
      DO 530 J=1,NBDISC
      RAD(1,J)=AVERAD(J)/10.0
      NN=RC(J)
      DO 540 I=2,NN
      II=I-1
      RAD(I,J)=RAD(II,J)-(RW(II,J)/10.0)
540   CONTINUE
530   CONTINUE
      DO 550 K=1,NLEAD
      ND=NDISCL(K)
      DO 560 J=1,ND
      RADL(1,J,K)=AVRADL(J,K)/10.0
      NN=RCL(J,K)
      DO 570 I=2,NN
      II=I-1
      RADL(I,J,K)=RADL(II,J,K)-(RWL(II,J,K)/10.0)
570   CONTINUE
560   CONTINUE
550   CONTINUE
*                                                                      *
*     CALCULATE THE NUMBER OF YEARS REQUIRED TO ATTAIN DISC HEIGHT     *
*                                                                      *
      DO 580 J=1,NBDISC
      TYRS(J)=AGE-(RC(J)*1.0)
580   CONTINUE
      DO 590 K=1,NLEAD
      DO 600 J=1,ITDL(K)
      TYRSL(J,K)=AGE-(RCL(J,K)*1.0)
600   CONTINUE
590   CONTINUE
*                                                                      *
*     CALCULATE BOLT LENGTHS BETWEEN CONSECUTIVE DISCS                 *
*                                                                      *
```

```
      BOLT(1)=DHT(1)-0.0
      DO 610 J=2,NBDISC
      JJ=J-1
      BOLT(J)=DHT(J)-DHT(JJ)
610   CONTINUE
      DO 620 K=1,NLEAD
      BOLTL(1,K)=DHTL(1,K)-DHT(NBDISC)
      ND=NDISCL(K)
      DO 630 J=2,ND
      JJ=J-1
      BOLTL(J,K)=DHTL(J,K)-DHTL(JJ,K)
630   CONTINUE
620   CONTINUE
*                                                                       *
*     PRINT OUT ORIGINAL MEASUREMENTS IN TABULAR FORMAT                 *
*                                                                       *
      CALL OUTPT1(NBDISC)
      CALL OUTPT2(ITDL)
*                                                                       *
*     INCORPORATE THE BOLE DATA WITH THE DATA FROM THE FIRST LEADER     *
*     TO CREATE THE DATA SET FOR THE "MAIN STEM"                        *
*                                                                       *
      NDISCM=NBDISC+ITDL(1)
      NN=ITDL(1)
      DO 640 J=1,NN
      JJ=NBDISC+J
      DN(JJ)=DNL(J,1)
      DHT(JJ)=DHTL(J,1)
      BOLT(JJ)=BOLTL(J,1)
      RC(JJ)=RCL(J,1)
      TYRS(JJ)=TYRSL(J,1)
      NM=RC(JJ)
      DO 650 I=1,NM
      RAD(I,JJ)=RADL(I,J,1)
650   CONTINUE
640   CONTINUE
*                                                                       *
*     CALCULATE THE TOTAL NUMBER OF TAPER CURVES FOR THE TREE           *
*     BASED ON THE SPECIFIED TIME INTERVAL BETWEEN TAPER CURVES         *
*                                                                       *
      NTC=AGE/INT
      X=(AGE/INT)-NTC
      IF(X.GE.0.1) NTC=NTC+1
*                                                                       *
*     CALCULATE AGES OF TAPER CURVES BASED ON THE SPECIFIED             *
*     TIME INTERVAL                                                     *
*                                                                       *
      A=AGE
      CVAGE(1)=A
      DO 660 I=2,NTC
      A=A-INT
      CVAGE(I)=A
660   CONTINUE
      NN=NTC-1
      DO 670 I=1,NN
```

```
      MIDAGE(I)=CVAGE(I)-(INT/2.0)
670   CONTINUE
      MIDAGE(NTC)=CVAGE(NTC)/2.0
*                                                                        *
*     CALCULATE TOTAL NUMBER OF TAPER CURVES FOR THE LEADERS             *
*     BASED ON THE SPECIFIED TIME INTERVAL BETWEEN TAPER CURVES          *
*                                                                        *
      DO 680 K=1,NLEAD
      DO 690 I=1,NTC
      IF(CVAGE(I).LE.TYRSL(1,K)) GO TO 700
690   CONTINUE
700   CONTINUE
      NTCL(K)=NTC-(CVAGE(I)/INT)
680   CONTINUE
*                                                                        *
*     CALCULATE AGES OF TAPER CURVES FOR THE LEADERS                     *
*                                                                        *
      DO 710 K=1,NLEAD
      A=AGE
      CVAGEL(1,K)=A
      N=NTCL(K)
      DO 720 I=2,N
      A=A-INT
      CVAGEL(I,K)=A
720   CONTINUE
      N=NTCL(K)-1
      DO 730 I=1,N
      MDAGEL(I,K)=CVAGEL(I,K)-(INT/2.0)
730   CONTINUE
      N=NTCL(K)
      MDAGEL(N,K)=CVAGEL(N,K)/2.0
710   CONTINUE
*                                                                        *
*     CALCULATE TOTAL HEIGHT OF TAPER CURVES FOR THE "MAIN STEM"         *
*                                                                        *
      CVHT(1)=DHT(NDISCM)
      DO 740 I=2,NTC
      DO 750 K=1,NDISCM
      J=(1-K)+NDISCM
      IF(TYRS(J).LE.CVAGE(I)) GO TO 760
      IF(J.NE.1) GO TO 750
      IF(DHT(J).EQ.0.0) GO TO 760
      GO TO 739
750   CONTINUE
760   CONTINUE
      JJ=J+1
      SLOPE=(DHT(JJ)-DHT(J))/(RC(JJ)-RC(J))
      X=(TYRS(J)-CVAGE(I))*SLOPE
      CVHT(I)=DHT(J)+X
      GO TO 740
739   CONTINUE
      SLOPE=(DHT(J)-0.0)/(RC(J)-AGE)
      X=(0.0-CVAGE(I))*SLOPE
      CVHT(I)=0.0+X
740   CONTINUE
```

```
*                                                                          *
*       CALCULATE TOTAL HEIGHT OF TAPER CURVES FOR THE LEADERS             *
*                                                                          *
        DO 770 KK=2,NLEAD
        N=NTCL(KK)
        CVHTL(1,KK)=DHTL(ITDL(KK),KK)
        DO 780 I=2,N
        M=ITDL(KK)
        DO 790 K=1,M
        J=(1-K)+M
        IF(TYRSL(J,KK).LE.CVAGEL(I,KK)) GO TO 800
790     CONTINUE
800     CONTINUE
        JJ=J+1
        SLOPE=(DHTL(JJ,KK)-DHTL(J,KK))/(RCL(JJ,KK)-RCL(J,KK))
        X=(TYRSL(J,KK)-CVAGEL(I,KK))*SLOPE
        CVHTL(I,KK)=DHTL(J,KK)+X
780     CONTINUE
770     CONTINUE
*                                                                          *
*       CALCULATE NUMBER OF PLOTTED POINTS ON EACH TAPER CURVE FOR         *
*       THE "MAIN STEM"                                                    *
*                                                                          *
        DO 810 I=1,NTC
        DO 820 J=1,NDISCM
        K=(NDISCM+1)-J
        X=CVHT(I)-DHT(K)
        IF(X.GE.0.0) GO TO 830
820     CONTINUE
830     CONTINUE
        IF(X.LE.0.009) GO TO 840
        NPP(I)=K+1
        GO TO 810
840     CONTINUE
        NPP(I)=K
810     CONTINUE
*                                                                          *
*       PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI                *
*                                                                          *
        WRITE(22,280)PN,TN,CODE(SC),NLEAD
        WRITE(22,290)
        WRITE(22,300)
*                                                                          *
*       CALCULATE MAI AND PAI FOR HEIGHT FOR THE "MAIN STEM"               *
*                                                                          *
        CALL MAIPAI(CVHT,1)
*                                                                          *
*       CALCULATE NUMBER OF PLOTTED POINTS ON EACH TAPER CURVE FOR THE    *
*       LEADERS                                                            *
*                                                                          *
        DO 850 K=2,NLEAD
        N=NTCL(K)
        DO 860 I=1,N
        M=ITDL(K)
        DO 870 J=1,M
```

```
      KK=(M+1)-J
      X=CVHTL(I,K)-DHTL(KK,K)
      IF(X.GE.0.0) GO TO 880
870   CONTINUE
880   CONTINUE
      IF(X.LE.0.009) GO TO 890
      NPPL(I,K)=KK+1
      GO TO 860
890   CONTINUE
      NPPL(I,K)=KK
860   CONTINUE
850   CONTINUE
*                                                                    *
*     DETERMINE DBH VALUES FOR EACH TAPER CURVE OF THE "MAIN STEM"   *
*                                                                    *
      DO 900 I=1,NTC
      NPP1=NPP(I)+1
      DO 910 J=1,NPP1
      JJ=J-1
      IF(DHT(J).GE.1.3) GO TO 920
910   CONTINUE
920   CONTINUE
      IF(DHT(J).EQ.1.3) GO TO 899
      IF(JJ.EQ.0) GO TO 930
      IF(CVHT(I).LT.1.3) GO TO 901
      SLOPE=(RAD(I,J)-RAD(I,JJ))/(DHT(J)-DHT(JJ))
      X=((1.3)-DHT(JJ))*SLOPE
      DBH(I)=(RAD(I,JJ)+X)*2.0
      IF(DBH(I).LE.0.0) DBH(I)=0.0
      IF(DBH(1).LE.0.0) GO TO 931
      GO TO 900
899   CONTINUE
      DBH(I)=RAD(I,J)*2.0
      IF(DBH(1).LE.0.0) GO TO 931
      GO TO 900
901   CONTINUE
      DBH(I)=0.0
      IF(DBH(1).LE.0.0) GO TO 931
900   CONTINUE
      GO TO 940
*                                                                    *
*     PRINT WARNING MESSAGE IF THERE ARE NO MEASUREMENTS RECORDED    *
*     BELOW BREAST HEIGHT                                            *
930   CONTINUE
      WRITE(22,360)
      GO TO 950
931   CONTINUE
      WRITE(22,342)
      GO TO 950
940   CONTINUE
*                                                                    *
*     PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI            *
*                                                                    *
      WRITE(22,280)PN,TN,CODE(SC),NLEAD
      WRITE(22,380)
```

```
      WRITE(22,390)
*                                                                    *
*     CALCULATE MAI AND PAI FOR DBH                                  *
*                                                                    *
      CALL MAIPAI(DBH,2)
950   CONTINUE
*                                                                    *
*     CALCULATE BASAL AREA BASED ON DBH                              *
*                                                                    *
      IF(DBH(1).LE.0.0) GO TO 971
      DO 960 I=1,NTC
      BA(I)=(DBH(I)**2)*00.007854
960   CONTINUE
*                                                                    *
*     PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI            *
*                                                                    *
      WRITE(22,280)PN,TN,CODE(SC),NLEAD
      WRITE(22,410)
      WRITE(22,420)
*                                                                    *
*     CALCULATE MAI AND PAI FOR BASAL AREA                           *
*                                                                    *
      CALL MAIPAI(BA,3)
*                                                                    *
*     CALCULATE TOTAL VOLUME UNDER EACH TAPER CURVE FOR THE          *
*     "MAIN STEM"                                                    *
*                                                                    *
971   CONTINUE
      DO 970 I=1,NTC
      VOL(I)=0.0
      V=(RAD(I,1)**2)*2*BOLT(1)*0.15708
      VOL(I)=VOL(I)+V
      NP=NPP(I)
      DO 980 J=2,NP
      JJ=J-1
      IF(CVHT(I).LE.DHT(J)) GO TO 985
      V=((RAD(I,J)**2)+(RAD(I,JJ)**2))*BOLT(J)*0.15708
      VOL(I)=VOL(I)+V
980   CONTINUE
      GO TO 986
985   CONTINUE
      TIPLEN=CVHT(I)-DHT(JJ)
      TIPVOL=(RAD(I,JJ)**2)*TIPLEN*0.15708
      VOL(I)=VOL(I)+TIPVOL
986   CONTINUE
      K=NBDISC
      KK=K+1
      V=((RAD(I,KK)**2)+(RAD(I,K)**2))*BOLT(KK)*0.15708
      VOL(I)=VOL(I)-V
970   CONTINUE
*                                                                    *
*     CALCULATE TOTAL VOLUME UNDER EACH TAPER CURVE FOR THE LEADERS, *
*     EXCLUDING THE MAIN LEADER                                      *
*                                                                    *
      DO 990 K=2,NLEAD
```

```
      N=NTCL(K)
      DO 1000 I=1,N
      VOLL(I,K)=0.0
      NP=NPPL(I,K)
      DO 1010 J=2,NP
      JJ=J-1
      IF(CVHTL(I,K).LE.DHTL(J,K)) GO TO 1015
      V=((RADL(I,J,K)**2)+(RADL(I,JJ,K)**2))*BOLTL(J,K)*0.15708
      VOLL(I,K)=VOLL(I,K)+V
1010  CONTINUE
      GO TO 1016
1015  CONTINUE
      TIPLEN=CVHTL(I,K)-DHTL(JJ,K)
      TIPVOL=(RADL(I,JJ,K)**2)*TIPLEN*0.15708
      VOLL(I,K)=VOLL(I,K)+TIPVOL
1016  CONTINUE
1000  CONTINUE
990   CONTINUE
*                                                                    *
*     CALCULATE VOLUME FOR THE BOLT BETWEEN THE LAST BOLE DISC AND   *
*     THE FIRST LEADER DISC, AND REMOVE THE AMOUNT FROM THE TOTAL    *
*     VOLUME                                                         *
*                                                                    *
      X=DHTL(1,1)-DHT(NBDISC)
      IF(X.LE.0.0) GO TO 1020
      BLM=BOLEHT-DHT(NBDISC)
      N=RC(NBDISC)/INT
      X=(RC(NBDISC)/INT)-N
      IF(X.GE.0.5) N=N+1
      J=NBDISC
      DO 1030 I=1,N
      V=((RAD(I,J)**2)*2)*BLM*0.15708
      VOL(I)=VOL(I)+V
      VOLL(I,1)=0.0
1030  CONTINUE
      DO 1040 K=1,NLEAD
      BLL(K)=DHTL(1,K)-BOLEHT
      N=NTCL(K)
      DO 1050 I=1,N
      V=((RADL(I,1,K)**2)*2)*BLL(K)*0.15708
      VOLL(I,K)=VOLL(I,K)+V
1050  CONTINUE
1040  CONTINUE
1020  CONTINUE
*                                                                    *
*     CALCULATE THE TOTAL VOLUME UNDER EACH TAPER CURVE, INCLUDING   *
*     ALL LEADER VOLUMES                                             *
*                                                                    *
      DO 1060 K=1,NLEAD
      DO 1070 I=1,NTCL(K)
      VOL(I)=VOL(I)+VOLL(I,K)
1070  CONTINUE
1060  CONTINUE
*                                                                    *
*     PRINT IDENTIFYING INFORMATION BEFORE CALLING MAIPAI            *
```

```
*                                                                        *
        WRITE(22,280)PN,TN,CODE(SC),NLEAD
        WRITE(22,450)
        WRITE(22,460)
*                                                                        *
*       CALCULATE MAI AND PAI FOR TOTAL VOLUME                           *
*                                                                        *
        CALL MAIPAI(VOL,4)
470     CONTINUE
*                                                                        *
*       END OF STEM ANALYSIS LOOP; GO TO 10 AND PROCESS THE NEXT TREE    *
*                                                                        *
        GO TO 10
*                                                                        *
*       WHEN ALL TREES ARE ANALYZED, END THE RUN                         *
*                                                                        *
100     CONTINUE
        CLOSE(UNIT=20)
        CLOSE(UNIT=21)
        CLOSE(UNIT=22)
        STOP
        END
*                                                                        *
**************************************************************************
```

```
*************************************************************************
*                                                                       *
      SUBROUTINE OUTPT1(ITD)
*                                                                       *
*************************************************************************
*                                                                       *
*                     SUBROUTINE IDENTIFICATION                         *
*                                                                       *
*             THIS SUBROUTINE ORGANIZES THE INPUT DATA INTO             *
*       TABULAR OUTPUT FOR SINGLE LEADER TREES, OR THE MAIN BOLE        *
*       OF MULTIPLE TOP TREES.                                          *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD        *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE      *
*       VMS OPERATING SYSTEM.                                           *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM     *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY    *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS      *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE      *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR      *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                          *
*                                                                       *
*       AUTHOR:          JOANNE KAVANAGH                                *
*                        SCHOOL OF FORESTRY                             *
*                        LAKEHEAD UNIVERSITY                            *
*                        THUNDER BAY, ONTARIO, P7B 5E1                  *
*                                                                       *
*************************************************************************
*                                                                       *
*                        VARIABLE IDENTIFICATION                        *
*                                                                       *
*       ALL VARIABLES ARE COMMON TO THE MAIN PROGRAM                    *
*                                                                       *
*************************************************************************
*                                                                       *
      REAL AGE,DHT(60),SBT(60),RAD(300,60),TYRS(60),BOLT(60)
      INTEGER ITD,CODE(75),PN,TN,SC,YR,NLEAD,DN(60),RC(60),YEAR
      COMMON/OUTP/PN,TN,SC,AGE,YR,NLEAD
      COMMON/OUTP1/DHT,BOLT,RC,TYRS,SBT,RAD,DN
      DATA NC/0/
      DATA CODE/'PW','PR','PJ','PS','  ','  ','  ','  ','  ','  ',
     2'  ','SW','SB','  ','  ','  ','  ','  ','HE','BF','  ','  ',
     3'  ','CE','TA','  ','  ','  ','  ','MH','  ','MR','MS','  ',
     4'  ','  ','BY','BW','  ','OW','OR','  ','  ','BE','AB','AW',
     5'  ','  ','EM','  ','BD','  ','  ','  ','  ','ID','  ','CB',
     6'  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ',
     7'  ','  ','  ','  ','PO'/
*                                                                       *
*       IF THERE ARE 18 DISCS OR LESS, GO TO 10                         *
*                                                                       *
      IF(ITD.LE.18) GO TO 10
      IPAGES=ITD/18
      IREM=ITD-(IPAGES*18)
      I1=1
      DO 20 K=1,IPAGES
*                                                                       *
```

```
*       PRINT IDENTIFYING INFORMATION                                    *
*                                                                        *
        WRITE(22,30)PN,TN,CODE(SC),NLEAD
30      FORMAT('1',/////,T31,'PLOT #',I3,5X,'TREE #',I3,5X,'SPECIES
       1 CODE ',A2,5X,'NUMBER OF LEADERS',I2)
        NC=K*18
        WRITE(22,40)(DN(I),I=I1,NC)
40      FORMAT('0',T4,'DISC NUMBER',4X,18I6)
        WRITE(22,50)(BOLT(I),I=I1,NC)
50      FORMAT('0',T4,'BOLT LENGTH (M)',1X,18F6.2)
        WRITE(22,60)(DHT(I),I=I1,NC)
60      FORMAT('0',T4,'DISC HEIGHT (M)',1X,18F6.2)
        WRITE(22,70)(RC(I),I=I1,NC)
70      FORMAT('0',T4,'RING COUNT',6X,18I6)
        WRITE(22,80)(TYRS(I),I=I1,NC)
80      FORMAT('0',T4,'AGE (YEARS)',5X,18F6.0)
        WRITE(22,90)(SBT(I),I=I1,NC)
90      FORMAT('0',T4'SBT (CM)',8X,18F6.3)
        WRITE(22,100)
100     FORMAT(/,T4,'RIB (CM)',/)
        YEAR=YR
        NY=RC(I1)
*                                                                        *
*       PRINT RADIUS MEASUREMENTS FOR EACH DISC FOR THE TAPER CURVES     *
*                                                                        *
        DO 110 I=1,NY
        WRITE(22,120)YEAR,(RAD(I,J),J=I1,NC)
120     FORMAT(' ',T12,I4,4X,18F6.3)
        YEAR=YEAR-1
110     CONTINUE
        I1=I1+18
20      CONTINUE
        IF(IREM.EQ.0) GO TO 140
        NC=NC+IREM
        GO TO 130
10      CONTINUE
        I1=1
        NC=ITD
130     CONTINUE
*                                                                        *
*       PRINT IDENTIFYING INFORMATION                                    *
*                                                                        *
        WRITE(22,30)PN,TN,CODE(SC),NLEAD
        WRITE(22,40)(DN(I),I=I1,NC)
        WRITE(22,50)(BOLT(I),I=I1,NC)
        WRITE(22,60)(DHT(I),I=I1,NC)
        WRITE(22,70)(RC(I),I=I1,NC)
        WRITE(22,80)(TYRS(I),I=I1,NC)
        WRITE(22,90)(SBT(I),I=I1,NC)
        WRITE(22,100)
        YEAR=YR
        NY=RC(I1)
*                                                                        *
*       PRINT RADIUS MEASUREMENTS FOR EACH DISC FOR THE TAPER CURVES     *
*                                                                        *
```

```
        DO 140 I=1,NY
        WRITE(22,120)YEAR,(RAD(I,J),J=I1,NC)
        YEAR=YEAR-1
140     CONTINUE
        RETURN
        END
*                                                                            *
******************************************************************************
```

```
***********************************************************************
*                                                                     *
      SUBROUTINE OUTPT2(ITDL)
*                                                                     *
***********************************************************************
*                                                                     *
*                       SUBROUTINE IDENTIFICATION                     *
*                                                                     *
*              THIS SUBROUTINE ORGANIZES THE INPUT DATA INTO          *
*     TABULAR OUTPUT FOR LEADERS OF MULTIPLE TOP TREES.               *
*              THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD     *
*     FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE      *
*     VMS OPERATING SYSTEM.                                           *
*              EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM  *
*     THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY    *
*     ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS      *
*     RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE      *
*     FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR      *
*     H. GARY MURCHISON AT THE SAME ADDRESS.                          *
*                                                                     *
*     AUTHOR:          JOANNE KAVANAGH                                *
*                     SCHOOL OF FORESTRY                              *
*                     LAKEHEAD UNIVERSITY                             *
*                     THUNDER BAY, ONTARIO, P7B 5E1                   *
*                                                                     *
***********************************************************************
*                                                                     *
*                       VARIABLE IDENTIFICATION                       *
*                                                                     *
*     ALL VARIABLES ARE COMMON TO THE MAIN PROGRAM                    *
*                                                                     *
***********************************************************************
*                                                                     *
      REAL AGE,DHTL(25,5),SBTL(25,5),RADL(300,25,5),TYRSL(25,5),
     1BOLTL(25,5)
      INTEGER ITDL(5),CODE(75),PN,TN,SC,YR,NLEAD,DNL(25,5),RCL(25,5)
     1,YEAR
      COMMON/OUTP/PN,TN,SC,AGE,YR,NLEAD
      COMMON/OUTP2/DHTL,BOLTL,RCL,TYRSL,SBTL,RADL,DNL
      DATA NC/0/
      DATA CODE/'PW','PR','PJ','PS','  ','  ','  ','  ','  ','  ',
     2'  ','SW','SB','  ','  ','  ','  ','  ','HE','BF','  ','  ',
     3'  ','CE','TA','  ','  ','  ','  ','MH','  ','MR','MS','  ',
     4'  ','  ','BY','BW','  ','OW','OR','  ','  ','BE','AB','AW',
     5'  ','  ','EM','  ','BD','  ','  ','  ','  ','ID','  ','CB',
     6'  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ','  ',
     7'  ','  ','  ','  ','PO'/
      DO 10 KK=1,NLEAD
*                                                                     *
*     IF THERE ARE 18 DISCS OR LESS GO TO 20                          *
*                                                                     *
      IF(ITDL(KK).LE.18) GO TO 20
      IPAGES=ITDL(KK)/18
      IREM=ITDL(KK)-(IPAGES*18)
      I1=1
```

```
      DO 30 K=1,IPAGES
*                                                                  *
*     PRINT IDENTIFYING INFORMATION                                *
*                                                                  *
      WRITE(22,40)PN,TN,CODE(SC),NLEAD,KK
40    FORMAT('1',/////,T21,'PLOT #',I3,5X,'TREE #',I3,5X,'SPECIES
     1 CODE ',A2,5X,'NUMBER OF LEADERS',I2,5X,'LEADER NUMBER',I2)
      NC=K*18
      WRITE(22,50)(DNL(I,KK),I=I1,NC)
50    FORMAT('0',T4,'DISC NUMBER',4X,18I6)
      WRITE(22,60)(BOLTL(I,KK),I=I1,NC)
60    FORMAT('0',T4,'BOLT LENGTH (M)',1X,18F6.2)
      WRITE(22,70)(DHTL(I,KK),I=I1,NC)
70    FORMAT('0',T4,'DISC HEIGHT (M)',1X,18F6.2)
      WRITE(22,80)(RCL(I,KK),I=I1,NC)
80    FORMAT('0',T4,'RING COUNT',6X,18I6)
      WRITE(22,90)(TYRSL(I,KK),I=I1,NC)
90    FORMAT('0',T4,'AGE (YEARS)',5X,18F6.0)
      WRITE(22,100)(SBTL(I,KK),I=I1,NC)
100   FORMAT('0',T4,'SBT (CM)',8X,18F6.3)
      WRITE(22,110)
110   FORMAT(/,T4,'RIB (CM)',/)
      YEAR=YR
      NY=RCL(I1,KK)
*                                                                  *
*     PRINT RADIUS MEASUREMENTS FOR EACH DISC FOR THE TAPER CURVES *
*                                                                  *
      DO 120 I=1,NY
      WRITE(22,130)YEAR,(RADL(I,J,KK),J=I1,NC)
130   FORMAT(' ',T12,I4,4X,18F6.3)
      YEAR=YEAR-1
120   CONTINUE
      I1=I1+18
30    CONTINUE
      IF(IREM.EQ.0) GO TO 150
      NC=NC+IREM
      GO TO 140
20    CONTINUE
      I1=1
      NC=ITDL(KK)
140   CONTINUE
*                                                                  *
*     PRINT IDENTIFYING INFORMATION                                *
*                                                                  *
      WRITE(22,40)PN,TN,CODE(SC),NLEAD,KK
      WRITE(22,50)(DNL(I,KK),I=I1,NC)
      WRITE(22,60)(BOLTL(I,KK),I=I1,NC)
      WRITE(22,70)(DHTL(I,KK),I=I1,NC)
      WRITE(22,80)(RCL(I,KK),I=I1,NC)
      WRITE(22,90)(TYRSL(I,KK),I=I1,NC)
      WRITE(22,100)(SBTL(I,KK),I=I1,NC)
      WRITE(22,110)
      YEAR=YR
      NY=RCL(I1,KK)
*                                                                  *
```

```
*       PRINT RADIUS MEASUREMENTS FOR EACH DISC FOR THE TAPER CURVES    *
*                                                                       *
        DO 150 I=1,NY
        WRITE(22,130)YEAR,(RADL(I,J,KK),J=I1,NC)
        YEAR=YEAR-1
150     CONTINUE
10      CONTINUE
        RETURN
        END
*                                                                       *
*************************************************************************
```

```
************************************************************************
*                                                                      *
      SUBROUTINE MAIPAI(DIM,NUM)
*                                                                      *
************************************************************************
*                                                                      *
*                        SUBROUTINE IDENTIFICATION                     *
*                                                                      *
*             THIS SUBROUTINE CALCULATES MEAN ANNUAL INCREMENTS;        *
*       MAI, AND PERIODIC ANNUAL INCREMENTS; PAI, FOR ANY GIVEN        *
*       SET OF MEASUREMENTS, SUCH AS HEIGHT, DBH, BASAL AREA OR        *
*       VOLUME.                                                        *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD       *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE     *
*       VMS OPERATING SYSTEM.                                          *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM    *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY   *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS     *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE     *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR     *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                         *
*                                                                      *
*       AUTHOR:        JOANNE KAVANAGH                                 *
*                      SCHOOL OF FORESTRY                              *
*                      LAKEHEAD UNIVERSITY                             *
*                      THUNDER BAY, ONTARIO, P7B 5E1                   *
*                                                                      *
************************************************************************
*                                                                      *
*                        VARIABLE IDENTIFICATION                       *
*                                                                      *
*       MOST OF THE VARIABLES ARE COMMON TO THE MAIN PROGRAM           *
*                                                                      *
*       DIM = ANY GIVEN ARRAY OF DIMENSIONS, SUCH AS HEIGHT            *
*       X = AGES TO BE PLOTTED WITH THE MAI AND PAI VALUES             *
*       Y = MAI AND PAI VALUES TO BE PLOTTED                           *
************************************************************************
*                                                                      *
      REAL MAI(300),PAI(300),DIM(300),INT,CVAGE(300),MIDAGE(300),
     2X(300,2),Y(300,2)
      INTEGER NTC,NUM
      COMMON/PAIMAI/INT,NTC,MIDAGE,CVAGE
      NN=NTC-1
*                                                                      *
*     CALCULATE PAI                                                    *
*                                                                      *
      DO 10 I=1,NN
      II=I+1
      PAI(I)=(DIM(I)-DIM(II))/INT
10    CONTINUE
      PAI(NTC)=DIM(NTC)/CVAGE(NTC)
*                                                                      *
*     CALCULATE MAI AND ORGANIZE  MAI'S, PAI'S, AND AGES FOR PLOTTING  *
*                                                                      *
      S=0.0
```

```
        DO 25 I=2,NTC
        J=I-1
        SS=ABS(PAI(I)-PAI(J))
        IF(SS.LE.0.0009) SS=0.0
        S=S+SS
25      CONTINUE
        DO 20 I=1,NTC
        MAI(I)=DIM(I)/CVAGE(I)
        X(I,1)=CVAGE(I)
        X(I,2)=MIDAGE(I)
        Y(I,1)=MAI(I)
        Y(I,2)=PAI(I)
20      CONTINUE
*                                                                    *
*       PRINT MAI AND PAI OUTPUT IN TABULAR FORMAT                   *
*                                                                    *
        DO 30 I=1,NTC
        WRITE(22,40)INT,CVAGE(I),DIM(I),MAI(I),PAI(I)
40      FORMAT(' ',7X,4(F10.4),/,48X,F10.4)
30      CONTINUE
        IF(S.EQ.0.0) GO TO 110
        WRITE(22,50)
50      FORMAT('1')
        GO TO (1,2,3,4),NUM
1       CONTINUE
        WRITE(22,60)
60      FORMAT(' ',1X,'MAI AND PAI FOR HEIGHT (METRES)',/)
        GO TO 70
2       CONTINUE
        WRITE(22,80)
80      FORMAT(' ',1X,'MAI AND PAI FOR DBH (CENTIMETRES)',/)
        GO TO 70
3       CONTINUE
        WRITE(22,90)
90      FORMAT(' ',1X,'MAI AND PAI FOR BASAL AREA (SQUARE
        2DECIMETRES)',/)
        GO TO 70
4       CONTINUE
        WRITE(22,100)
100     FORMAT(' ',1X,'MAI AND PAI FOR VOLUME (CUBIC DECIMETRES)',/)
70      CONTINUE
*                                                                    *
*       CALL THE PLOTTING ROUTINE TO PLOT MAI AND PAI VERSUS TIME     *
*                                                                    *
        CALL MLPLOT(22,X,Y,300,NTC,NTC,0,0)
        GO TO 130
110     CONTINUE
        WRITE(22,120)
120     FORMAT('1',//,10X,'UNABLE TO PLOT MAI AND PAI DUE TO INSUFFICIEN
        1T RANGE OF VALUES')
130     CONTINUE
        RETURN
        END
*                                                                    *
**********************************************************************
```

```
*************************************************************************
*                                                                       *
        SUBROUTINE MLPLOT (IOUT, X, Y, NROW, N1, N2, N3, N4)
*                                                                       *
*************************************************************************
*                                                                       *
*                       SUBROUTINE IDENTIFICATION                       *
*                                                                       *
*               THIS SUBROUTINE IS A MODIFIED VERSION OF BPLOT4         *
*       DESIGNED TO PLOT FOUR FUNCTIONS SIMULTANEOUSLY.                 *
*               THE FUNCTIONS TO BE PLOTTED ARE STRUNG TOGETHER AS      *
*       A LONG VECTOR OF ORDERED PAIRS WITH A CORRESPONDING CHARACTER   *
*       VECTOR.  THESE PAIRS ARE PLOTTED AS BEFORE BUT THE SPECIFIC     *
*       CHARACTER ASSOCIATED WITH A PAIR IS USED FOR THE PLOT           *
*       CHARACTER.  THUS, IF ALL THE PAIRS BELONGING TO ONE HAVE        *
*       THE SAME CHARACTER, A CHARACTER DIFFERENT FROM THE OTHERS,      *
*       THE PLOT OF THESE PAIRS WILL APPEAR DISTINCT FROM THE OTHERS.   *
*               THIS SUBROUTINE SETS UP THE INPUT DATA (IN MATRICES     *
*       X AND Y) FOR USE BY THE PPLOTM SUBROUTINE.                      *
*               SUBROUTINE MLPLOT IS CALLED TO SET THE FOUR SETS        *
*       OF ORDERED PAIRS INTO ONE VECTOR ALONG WITH A CORRESPONDING     *
*       VECTOR OF PLOT CHARACTERS.                                      *
*               THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD      *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE      *
*       VMS OPERATING SYSTEM.                                           *
*               EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM   *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY    *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS      *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE      *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR      *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                          *
*                                                                       *
*       ADAPTED FOR USE IN STEM.FOR BY:                                 *
*                                                                       *
*                       JOANNE KAVANAGH                                 *
*                       SCHOOL OF FORESTRY                              *
*                       LAKEHEAD UNIVERSITY                             *
*                       THUNDER BAY, ONTARIO, P7B 5E1                   *
*                                                                       *
*       CREDIT TO:                                                      *
*                                                                       *
*               DR. J. WARKENTIN  LAKEHEAD UNIVERSITY MATH DEPT.        *
*                       FOR THE ORIGINAL VERSION OF MLPLOT              *
*               DR. M. BENSON  LAKEHEAD UNIVERSITY MATH DEPT.           *
*                       FOR MODIFYING MLPLOT FOR USE ON THE VAX 11/780  *
*               MR. L. MAYES  LAKEHEAD UNIVERSITY COMPUTER CENTRE       *
*                       FOR CORRECTING MINOR ERRORS IN MLPLOT           *
*                                                                       *
*************************************************************************
*                                                                       *
*                       VARIABLE IDENTIFICATION                         *
*                                                                       *
*       X = NROW BY 4 MATRIX WHOSE COLUMNS ARE ABSCISSAS FOR PLOTS      *
*       Y = NROW BY 4 MATRIX WHOSE COLUMNS ARE ORDINATES FOR PLOTS      *
*               (ONE COLUMN PER FUNCTION)                               *
```

```
*       NROW = ACTUAL NUMBER OF ROWS DECLARED FOR X AND Y IN CALLING    *
*               PROGRAM                                                  *
*       N1,N2,N3,N4 = NUMBER OF POINTS IN EACH CURVE                     *
*               A ZERO INDICATES NO PLOT FOR THE CORRESPONDING           *
*               COLUMNS IN X AND Y.                                      *
*       IOUT = OUTPUT DEVICE # 22 (PRINTER)                              *
*                                                                        *
**************************************************************************
*                                                                        *
        REAL X(NROW,4),Y(NROW,4)
        REAL XO(480),YO(480)
        INTEGER NF,N,NT,CHRV(480),CHAR,IOUT,SSSS
*                                                                        *
*       THE MAXIMUM NUMBER OF ALLOWABLE PLOTTED POINTS IS 480            *
*                                                                        *
        MAXPT=480
        J=1
*                                                                        *
*       TRANSFER FIRST FUNCTION TO PLOT VECTOR IF NECESSARY              *
*                                                                        *
        IF(N1.EQ.0) GO TO 10
        CHAR='*'
        DO 20 I=1,N1
        XO(J)=X(I,1)
        YO(J)=Y(I,1)
        CHRV(J)=CHAR
20      J=J+1
10      CONTINUE
*                                                                        *
*       TRANSFER SECOND IF NECESSARY                                     *
*                                                                        *
        IF(N2.EQ.0) GO TO 30
        CHAR='+'
        DO 40 I=1,N2
        XO(J)=X(I,2)
        YO(J)=Y(I,2)
        CHRV(J)=CHAR
40      J=J+1
30      CONTINUE
*                                                                        *
*       THIRD FUNCTION                                                   *
*                                                                        *
        IF(N3.EQ.0) GO TO 50
        CHAR='X'
        DO 60 I=1,N3
        XO(J)=X(I,3)
        YO(J)=Y(I,3)
        CHRV(J)=CHAR
60      J=J+1
50      CONTINUE
*                                                                        *
*       FOURTH FUNCTION                                                  *
*                                                                        *
        IF(N4.EQ.0) GO TO 70
        CHAR='O'
```

```
        DO 80 I=1,N4
        XO(J)=X(I,4)
        YO(J)=Y(I,4)
        CHRV(J)=CHAR
80      J=J+1
70      CONTINUE
        NT=N1+N2+N3+N4
        IF(NT.LE.MAXPT) GO TO 90
*                                                                        *
*       IF MORE THAN 480 PLOTTED POINTS ARE ASKED FOR, PRINT WARNING     *
*       MESSAGE ON THE SCREEN BEFORE ABORTING                            *
*                                                                        *
        TYPE 100,MAXPT
100     FORMAT(' ','MORE THAN ',I5,'POINTS------ABORTING')
        RETURN
90      CONTINUE
        SSSS=1
*                                                                        *
*       PRINTER                                                          *
*                                                                        *
        W=100
        H=40
*                                                                        *
*       FOR DOCUMENT GENERATION                                          *
*                                                                        *
        CALL PPLOTM (XO,YO,NT,W,H,CHRV,SSSS,IOUT)
        RETURN
        END
*                                                                        *
**************************************************************************
```

```
************************************************************************
*                                                                      *
      SUBROUTINE PPLOTM (X,Y,N,W,H,CHRV,SSSS,IOUT)
*                                                                      *
************************************************************************
*                                                                      *
*                      SUBROUTINE IDENTIFICATION                       *
*                                                                      *
*             SUBROUTINE PPLOTM PLOTS A MAXIMUM OF FOUR FUNCTIONS      *
*       AS REQUESTED BY SUBROUTINE MLPLOT.                             *
*             THIS ALGORITHM IS WRITTEN IN 1977 ANSIFOR STANDARD       *
*       FORTRAN FOR USE IN A DIGITAL VAX 11/780 COMPUTER USING THE     *
*       VMS OPERATING SYSTEM.                                          *
*             EVERY ATTEMPT HAS BEEN MADE TO REMOVE ALL ERRORS FROM    *
*       THIS ALGORITHM.  NEITHER THE AUTHOR, NOR LAKEHEAD UNIVERSITY   *
*       ACCEPT ANY RESPONSIBILITY FOR MISINTERPRETATIONS OR ERRORS     *
*       RESULTING FROM THE USE OF THE ALGORITHM.  SHOULD ERRORS BE     *
*       FOUND TO EXIST IN THE PROGRAM, PLEASE NOTIFY THE AUTHOR OR     *
*       H. GARY MURCHISON AT THE SAME ADDRESS.                         *
*                                                                      *
*       AUTHOR:         JOANNE KAVANAGH                                *
*                       SCHOOL OF FORESTRY                             *
*                       LAKEHEAD UNIVERSITY                            *
*                       THUNDER BAY, ONTARIO, P7B 5E1                  *
*                                                                      *
************************************************************************
*                                                                      *
*                      VARIABLE IDENTIFICATION                         *
*                                                                      *
*       X,Y = THE ORDERED PAIR VECTORS                                 *
*       CHRV = CHARACTER VECTOR ASSOCIATED WITH X,Y                    *
*       N = LENGTH OF ABOVE VECTORS                                    *
*       W = WIDTH OF PLOT                                              *
*       H = HEIGHT OF PLOT                                             *
*       SSSS = ORDERED PAIR OUTPUT SWITCH                              *
*       IOUT = OUTPUT DEVICE NUMBER                                    *
*                                                                      *
************************************************************************
*                                                                      *
      INTEGER N,K,I,IS,IL,J,K1,SSSS
      REAL VAL(250),X(N),Y(N),MX,MY,M,M1,XM,YM
      INTEGER BUF(111),CHAR,DASH,TO,EXCL
      INTEGER XPOS,YPOS,YNEXT,IX(250),WINT,CHRV(N)
      DOUBLE PRECISION FILNAM
      EQUIVALENCE (BUF,VAL)
      SCALE(I,A,B)=(I-A)*B
      SCLFAC(A,B)=10.**(-INT(ALOG10(0.5*(ABS(A)+ABS(B)))))
      W=AMAX1(10.0,AMIN1(W,122.0))
      H=AMAX1(10.0,AMIN1(H,45.0))
      DASH='-'
      TO='TO'
      EXCL='!'
      IF(N.LE.2)GO TO 10
      MX=1
      MY=1
```

```
 20      CONTINUE
         K=MOD(N,2)
         IF(K)30,30,40
 *                                                                    *
 *       IF N IS EVEN THEN DO                                         *
 *                                                                    *
 30      IS=N-1
         IF(X(IS+1).LT.X(IS))IS=IS+1
         XMIN=X(IS)
         XMAX=X(N-1+MOD(IS,2))
         GO TO 50
 *                                                                    *
 *       IF N IS ODD THEN DO                                          *
 *                                                                    *
 40      XMIN=X(N)
         XMAX=XMIN
 50      CONTINUE
         DO 60 I=1,N-3+K,2
         IS=I
         IF(X(IS+1).LT.X(IS))IS=IS+1
         IL=I+MOD(IS,2)
         XMIN=AMIN1(X(IS),XMIN)
 60      XMAX=AMAX1(X(IL),XMAX)
         XS=MX*W/(XMIN-XMAX)
         XC=(W+(XMIN+XMAX)*XS)*0.5+2.0001
         XS=-XS
 *                                                                    *
 *       SORT X AND Y VALUES INTO DESCENDING ORDER USING TRSRT2       *
 *                                                                    *
 *       1ST PHASE                                                    *
 *       CREATE TREE                                                  *
 *                                                                    *
         DO 70 K=2,N
 *                                                                    *
 *       REPOSITION Y(K) CORRECTLY                                    *
 *                                                                    *
         I=K
         M=X(K)
         M1=Y(K)
         CHAR=CHRV(K)
 80      IF(I.LE.1)GO TO 90
         J=I/2
         IF(M1.GE.Y(J))GO TO 90
         X(I)=X(J)
         Y(I)=Y(J)
         CHRV(I)=CHRV(J)
         I=J
         GO TO 80
 90      X(I)=M
         Y(I)=M1
         CHRV(I)=CHAR
 70      CONTINUE
 *                                                                    *
 *       2ND PHASE                                                    *
 *                                                                    *
```

```
        DO 100 K1=2,N
        K=N-K1+2
*                                                              *
*       PUT K'TH LARGEST NUMBER IN K'TH POSITION               *
*                                                              *
        CHAR=CHRV(K)
        M=X(K)
        M1=Y(K)
        X(K)=X(1)
        Y(K)=Y(1)
        CHRV(K)=CHRV(1)
*                                                              *
*       INSERT M AND M1 IN CORRECT POSITIONS                   *
*                                                              *
        I=1
110     J=2*I
        IF(J+1-K)120,130,140
120     IF(Y(J).GT.Y(J+1))J=J+1
130     IF(Y(J).GE.M1)GO TO 140
        X(I)=X(J)
        Y(I)=Y(J)
        CHRV(I)=CHRV(J)
        I=J
        GO TO 110
140     X(I)=M
        Y(I)=M1
        CHRV(I)=CHAR
100     CONTINUE
        YS=MY*H/(Y(1)-Y(N))
        YC=(H+(Y(1)+Y(N))*YS)*0.5  +  1.0001
        YS=-YS
        IF(SSSS.NE.0)GO TO 150
        WRITE(IOUT,160)(X(L1),Y(L1),CHRV(L1),L1=1,N)
160     FORMAT('-',14X,'X',21X,'Y',15X//(2G22.7,7X,A1))
150     CONTINUE
        XS1=SCLFAC(XMIN,XMAX)
        XS2=XS1/XS
        YS1=SCLFAC(Y(1),Y(N))
        YS2=YS1/YS
        IF(SSSS.EQ.0)PRINT 170
*                                                              *
*       OUTPUT BLOCK                                           *
*                                                              *
170     FORMAT('1')
        XM=1./XS1
        YM=1./YS1
        WINT=W+2.0001
        WRITE(IOUT,180)N,W,XMIN,XMAX,XM,H,Y(N),Y(1),YM,(DASH,L1=1,66)
180     FORMAT(' ','NO. OF POINTS =',I4,
        1//' ',T15,'PLOT SIZE',T30,'MIN VAL',T43,'MAX VAL',T53,
        2'SCALE FACTOR',/' ',T11,'(PRINT POSITIONS)',
        3//' ','HORIZ.(X)',T11,F10.1,T28,E11.4,T41,E11.4,F11.4,
        4//' ','VERT.(Y)',T11,F10.1,T28,E11.4,T41,E11.4,F11.4,
        9//' ',66A1/)
*                                                              *
```

```
*       THESE ADDITIONAL LINES OF FORMATTING MAY BE INCLUDED ON THE      *
*       OUTPUT PLOTS IF DESIRED                                           *
*                                                                          *
*       5///' ','NOTE:  THE VALUES LABELLING THE AXES ON THE GRAPH',     *
*       6' PRINTED BELOW'/' ',7X,'MUST BE MULTIPLIED BY A SCALE FACTOR',*
*       7' (AS PRINTED ABOVE)',                                           *
*       8/' ',7X,'IN ORDER TO OBTAIN THE TRUE AXIS VALUES. ',           *
        BUF(1)='!'
        DO 190 L=1,111
 190    BUF(L)=' '
        K=0
        I=1
        J=2
        YPOS=YC+YS*Y(1)
 210    IF(J.GT.N)GO TO 200
        YNEXT=YC+YS*Y(J)
        IF(YNEXT.NE.YPOS)GO TO 200
 220    J=J+1
        GO TO 210
 200    L=K+1
 230    IF(L.GE.YPOS)GO TO 240
        IF(MOD(L,5).NE.1)GO TO 250
        A=SCALE(L,YC,YS2)
        WRITE(IOUT,260)A,EXCL
 260    FORMAT(' ',F7.2,1X,122A1)
        GO TO 270
 250    WRITE(IOUT,320)EXCL
 270    L=L+1
        GO TO 230
 240    LL=J-1
        IIII=0
*                                                                          *
*       PLOT THE FUNCTIONS                                                 *
*                                                                          *
        DO 280 L=I,LL
        XPOS=XC+XS*X(L)
        IF((BUF(XPOS).NE.CHRV(L)).AND.(BUF(XPOS).NE.' '))BUF(XPOS)='@'
        IF (BUF(XPOS).NE.' ') GO TO 280
        IIII=IIII+1
        IX(IIII)=XPOS
        BUF(XPOS)=CHRV(L)
 280    CONTINUE
        IF(BUF(1).NE.' ')GO TO 290
        BUF(1)='!'
        IIII=IIII+1
        IX(IIII)=1
 290    CONTINUE
        IF(MOD(YPOS,5).NE.1.AND.J.NE.N+1)GO TO 300
        A=SCALE(YPOS,YC,YS2)
        WRITE(IOUT,260)A,(BUF(I),I=1,WINT)
        GO TO 310
 300    WRITE(IOUT,320)(BUF(I),I=1,WINT)
 320    FORMAT(' ',8X,122A1)
 310    CONTINUE
        DO 330 L=1,IIII
```

```
330     BUF(IX(L))=' '
        IF(J.GT.N)GO TO 340
        I=J
        K=YPOS
        YPOS=YNEXT
        GO TO 220
340     CONTINUE
        L=MINO(WINT+2,123)
        WRITE(IOUT,350)(DASH,I=1,L)
350     FORMAT(' ',8X,130A1)
        DO 360 I=1,L,10
360     BUF(I)='!'
        WRITE(IOUT,370)(BUF(I),I=1,L)
370     FORMAT(' ',9X,121A1)
        DO 380 I=2,L,10
380     VAL(I)=SCALE(I,XC,XS2)
        WRITE(IOUT,390)(VAL(I),I=2,L,10)
390     FORMAT(' ',5X,12(F7.2,3X))
400     CONTINUE
        RETURN
*                                                                      *
*       IF THERE ARE TOO FEW PAIRS FOR PLOTTING, PRINT AN ERROR        *
*       MESSAGE BEFORE CONTINUING                                      *
*                                                                      *
10      WRITE(IOUT,410)
410     FORMAT(' TOO FEW PAIRS OF VALUES')
        GO TO 400
        END
*                                                                      *
************************************************************************
```