Department of Conservation and Land Management

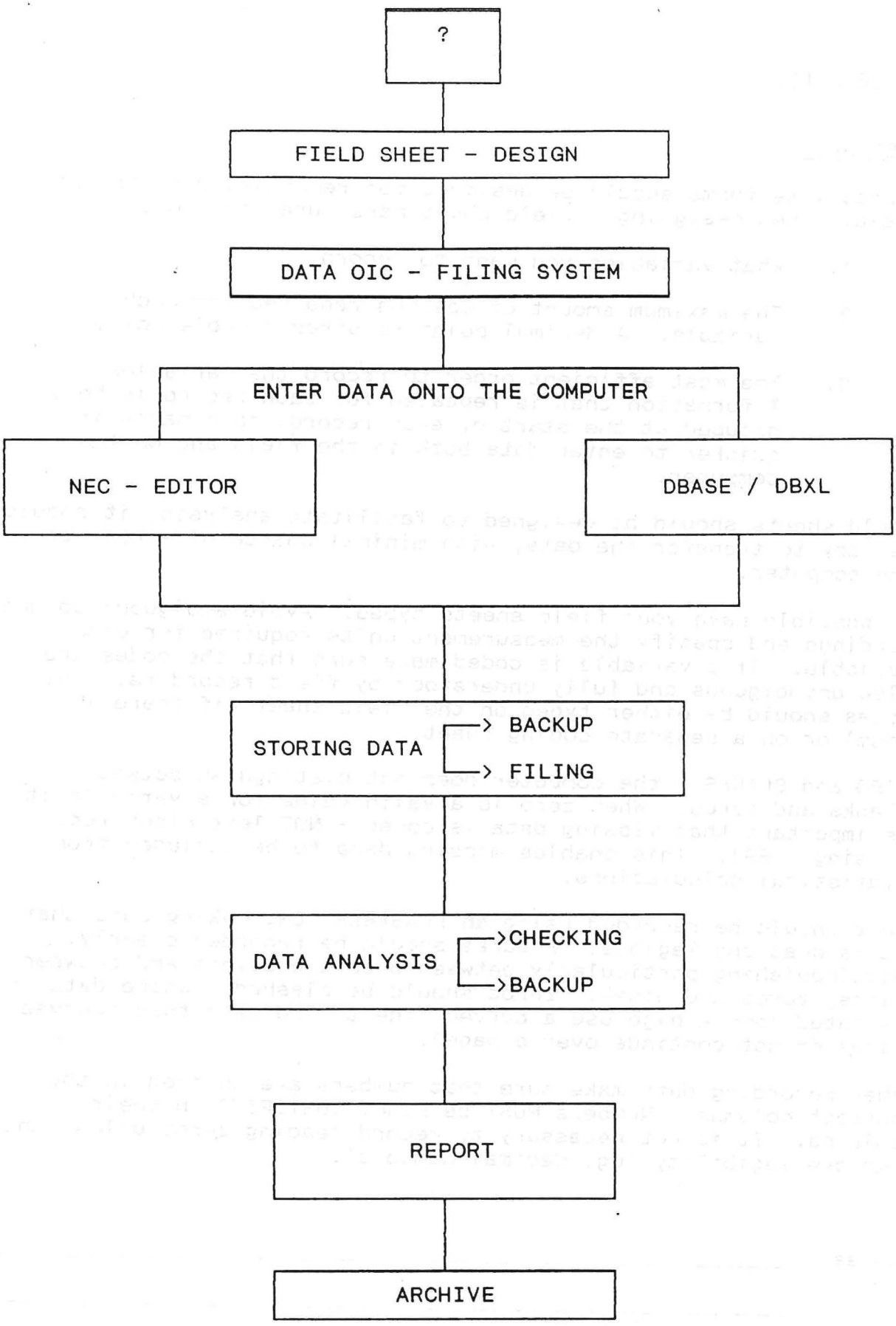
Manjimup Research Centre

GUIDELINES FOR DATA MANAGEMENT

Yvonne Woods

June 1989

INDEX


?
FIELD SHEET - DESIGN
DATA OIC - FILING SYSTEM
ENTER DATA ONTO THE COMPUTER
NEC - EDITOR
DBASE / DBXL
STORING DATA
BACKUP
FILING
DATA ANALYSIS
CHECKING
BACKUP
REPORT
ARCHIVE

RECORDING DATA:

Field Sheets:

Appropriate forms should be designed for recording data in the field. When designing a field sheet make sure you know:

1. What variables you want to record.

2. The maximum amount of columns required for each variable. A decimal point requires a whole column.

3. The most efficient order to record the variables. Information that is repeated for each record is best grouped at the start of each record, this makes it quicker to enter data both in the field and on the computer.

Field sheets should be designed to facilitate analysis, it should be easy to transfer the data, with minimal chance of error, to the computer.

If possible have your field sheets typed. Avoid ambiguous column headings and specify the measurement units required for each variable. If a variable is coded make sure that the codes are also unambiguous and fully understood by field recorders. The codes should be either typed on the field sheet (if there is room) or on a separate coding sheet.

ZERO and BLANKS - the computer does not distinguish between blanks and zeros. When zero is a valid value for a variable it is important that missing data is coded - NOT left blank (eg. missing = 99). This enables missing data to be excluded from statistical calculations.

Data should be recorded using an ERASABLE PEN, making sure that it is neat and legible. Fiqures should be recorded clearly, distinquishing particularly between ones and sevens and between sixes, zeros and nines. Zeros should be slashed. Where data is repeated down a page use a curved line to indicate this (curved lines do not continue over a page).

When recording data make sure that numbers are entered in the correct columns. Numbers MUST be RIGHT JUSTIFIED in their columns. It is not necessary to record leading zeros unless they improve legibility (eg. decimal numbers).

Notes: _____

_____

_____

FILING DATA:

Data OIC:

All data should be assigned to an officer who will be responsible for its management and maintenance.

Filing System:

The data OIC should organise a system for filing the data. Organise somewhere to keep the completed field sheets, using either ring binders or loose leaf folders, and other information relating to the data.

For example:

1. Organise a section in a vertical filing draw for all information and data relating to the study.

2. Organise a loose leaf folder for storing completed field sheets. Use a ring binder if you need to take completed sheets back into the field (eg. vegetation re-assessment).

3. Organise a manila folder or district file for storing other information relevant to the study such as:

   (i) Original field sheet (in plastic protector).
   (ii) Copy of Data Identification Sheet
   (iii) Copy of Variable Description Sheet
   (iv) Copy of Code Description Sheet
   (v) File Summary Sheet.
   (vi) Copies of all programs associated with data.
   (vii) Results of any analysis.

4. Large printouts may be kept in a suspension computer file.

Notes: _____

_____

_____

_____

_____

_____

_____

Data Identification Sheet:

Department of Conservation and Land Management
Manjimup Research Centre

DATA IDENTIFICATION SHEET

Research Working Plan: /

Project Title: _____

_____

Project OIC: _____

Data Filename: _____

No. Records: _____

Description: _____

_____

Data OIC: _____

Date: _____

Data Location: _____

Backup Location: _____

Archive Location: _____

Comments: _____

_____

The Data Identification Sheet should provide all the relevant information relating to the database.

Notes: _____

_____

_____

_____

_____

Variable Description Sheet:

Department of Conservation and Land Management
Manjimup Research

VARIABLE DESCRIPTION SHEET

Filename: _____ Date: _____

Variable	Columns	Description	Units

All variables in the database should be recorded on this sheet. Variable names should be kept to 8 characters.

Notes: _____

Code Description Sheet:

Department of Conservation and Land Management
Manjimup Research Centre

VARIABLE CODE SHEET

Filename: _____ Date: _____

Variable	Code	Description

The Variable Code Sheet should be completed for all coded variables.

Notes: _____

File Summary Sheet:

Department of Conservation and Land Management
Manjimup Research Centre

FILE SUMMARY SHEET

Filename Code: _____

General Description: _____

_____

Date	Filename	Description/Purpose	No. Records	OIC	Data Location	Backup Location

The File Summary Sheet is a reference to all the work associated with a particular database. It is a summary of all the work you have done with the data and should contain the filenames of your original data file, any other subsequent versions of your data, and any program or documentation files.

The File Summary Sheet enables you to keep track of what work has been done with a database, and where these files are located.

Notes: _____

_____

_____

ENTERING DATA ONTO THE COMPUTER:

Data can be entered onto the computer using several methods. The two most common methods are using the Norton Editor or Dbase.

Filenames:

All files created on a computer are referenced by a filename. Filenames identify files and must be unique. A filename consists of one to eight characters, followed by a period (.) and a three character extension. The file extension describes what type of file it is.

Valid file extensions for data related files are:

General	.DAT	Original ASCII data file
	.CY1-9	Versions (cycles) 1-9 of original ASCII file
	.TXT	Text file
	.LOG	Data logger file
	.OUT	Output file
	.LST	Listing file
	.TMP	Temporary file
SAS	.SAS	SAS program file
Dbase	.DBF	Database file
	.DBT	Memo fields file
Lotus	.WKS	Spreadsheet file
	.PRN	ASCII input file
	.PIC	Picture (graphics) file
Multimate	.DOC	Document file

As filenames are the basic reference for your data please choose an appropriate name with care. To avoid problems of people using the same name for different files ALWAYS use your initials for the first 2 characters. The remaining 6 characters can be used to describe your data file. If you will be collecting data periodically use the date or year as part of your filename.

For example:

PHPOOL86.DAT	Penni Hewett, Poole Thinning Expt. 1986
PHPOOL88.DAT	Penni Hewett, Poole Thinning Expt. 1988

Notes: _____

_____

_____

For larger experiments and surveys a code may be assigned to data files instead of using your initials.

For example:

GRCFSU86.DAT	Gray 6, Census & Foraging, Summer 1986
GRVEG88.DAT	Gray 6, Vegetation Assessment, 1988
TINSTA89.DAT	Tingle Study, Stand Parameters 1989
TINLAN89.DAT	Tingle Study, Landform Data 1989
TINVEG89.DAT	Tingle Study, Vegetation Data 1989

If you wish to assign a code to your data check the Data Library Index to make sure it has not already been used.

Notes: _____

USING THE NORTON EDITOR:

The Norton Editor is a full screen text editor that allows you to create ASCII files. The Norton Editor can be accessed directly from the main menu or if you are in the Operating System you can enter NEC at the DOS prompt and press <ENTER>. You will be prompted for a filename. To access a floppy disk in the A: drive you must enter A:filename (eg. A:YWDATA.DAT). If the file exists, the editor reads it, otherwise, the editor creates a new file with that name. A screen similar to the following will appear:

```
Norton Editor
A Programmers Full-Screen Editor
Version 1.3B
(c)Copyright 1986, S. Reifel & Co.
```

Press F1 for help
Press any key to begin

The Norton Editor has 3 help screens to assist you. The help screens are activated by pressing F1. Successively pressing F1 cycles through the 3 help screens and back to the editing session.

The editor provides two further sources of information. One is a summary status line that appears at the bottom of the screen. This displays the following:

(i) The line and column numbers of the cursor location
(ii) The filename of the file being edited
(iii) Whether you are in INSERT or REPLACE mode
(iv) Whether word-wrap mode is on

When a command has been activated using one of the function keys, the status line displays a command summary that shows you the letter keys that can be used to complete the command. In addition to the status line the F2 key can be pressed to display a full status screen that summarizes the current operation of the editor, including the amount of disk space available.

The Norton Editor always saves the existing file as the backup file, renaming it with the same name but a different extension. The Norton Editor shifts the extension one character and adds a "~" at the beginning of the extension. Thus if you where editing a file called YWDATA.DAT, after the editing session the original file will be named YWDATA.~DA and the current file will be called YWDATA.DAT.

COMMON COMMANDS - NORTON EDITOR:

The editor's command structure is primarily based on the use of a function key (F3-F7) followed by a letter key.

FUNCTION KEYS:

F1	Help
F2	Status display screen
F3	File and disk commands
F4	Block commands
F5	Screen and format commands
F6	Miscellaneous commands
F7	Printer commands
F8	Not used
F9	DOS command processor
F10	Not used

F3 - File commands:

E	Exit and save data - end editing session
Q	Quit and don't save data - end editing session
S	Save data - continue editing
A	Append another file to end of edit data
C	Close output file, open new output

F4 - Block commands:

S	Set a block marker
R	Remove block markers
D	Delete a block
C	Copy a block
M	Move a block

F5 - Screen format commands:

L	Set the line length (for word-wrap)
W	Word-wrap, toggle on and off

F6 - Miscellaneous commands:

G	Goto line number
Ins	INSERT mode cancel, switch to REPLACE mode

F7 - Printer commands:

P	Print entire edit buffer
B	Block, print marked block
E	Eject paper, form feed

The control CTRL and alternate ALT keys work slightly differently to the FUNCTION keys. ALT-X indicates key combinations formed by holding down the ALT key while pressing the letter key. CTRL-X indicates key combinations formed by holding down the CTRL key while pressing the letter key.

Cursor Movement:

The arrow keys can be used to move the cursor in any direction. The PgUp and PgDn keys move the cursor to the top or bottom of the screen. CRTL-PgUp will move the cursor to the start of the file and CTRL-PgDn moves to the end of the file.

Delete Commands:

The backspace deletes one character to the left, when in INSERT mode. The DEL key can be used to delete the character under the cursor. ALT-K will delete the entire current line. CTRL-U will UNDELETE text, provided that the cursor has not been moved.

Find and Replace Commands:

This is a particularly useful command. ALT-F begins a FORWARD search from the cursor location to the end of the file, and CTRL-F begins a BACKWARD search from the cursor to the start of the file. After pressing ALT-F or CTRL-F the editor will prompt you for the string you wish to search for. After entering the string press ENTER, the editor will search for the exact match to the sting you have entered, in the file. If you wand to FIND and REPLACE, rather than just find, mark the end of the search string with another ALT-F or CTRL-F and then enter in the replacement string before pressing ENTER.

Both FIND and FIND and REPLACE will stop on the first matching string. For FIND and REPLACE, the editor pauses for confirmation before replacing the text. Pressing Y will cause the editor to replace the text and continue searching for the next matching string. If you press N, the editor does not replace this instance, but continues searching. If your press *, the editor switches to the global replace operation, finding and replacing all matching strings without pausing for confirmation. If you press space, or any other key, the editor stops the find and replace operation.

ALT-C and CTRL-C can be used to continue your search. ALT-C continues forward and CTRL-C continues backwards.

Notes: _____

_____

_____

_____

USING DBASE III/DBXL:

Dbase III or DBXL can be accessed directly from the main menu by entering "D". A copywrite screen may appear and the dot (.) prompt will be displayed.

Creating a Database:

Use the CREATE command to define a database structure. For files to be stored on diskette, prefix the filename with the drive specifier (eg. A:). Thus to create a file called YWDATA on drive A: use the following command:

CREATE A:YWDATA

The system will display the file creation menu on the screen. For each field, type in a variable (field) name, select the data type (pressing the space bar will display the next field type - eg. character, numeric ...), and then enter the field length and the number of decimal places. Although Dbase/DBXL will allow field names of up to ten characters, SAS and other packages only allow field names to have 8 characters. Try to use meaningful field names and keep them to 8 characters in length. If you have many variables with similar names it may be easier to name them VAR1, VAR2, VAR3... etc.

Define all required fields. When finished hold down the CTRL key and press W (CTRL-W). The system will then ask for confirmation, and then ask if data is to be input into the file. Data may be input if 'Y' is entered.

Using a Database:

Before a database can be accessed, it must be made available to the system. If a database has just been created (as above), it is automatically in use. If a database already exists the following command is required:

USE filename (eg. USE A:YWDATA)

Remember to include the drive specifier if your database exists on a floppy disk.

Notes: _____

_____

_____

_____

_____

Editing and Appending Data:

Once a file is in use, data can be edited/added in a number of ways:

APPEND

Adds new records to the end of the database file.

To save and exit press ENTER when a new record is presented.

EDIT

Displays and edits existing records, one at a time. The PgUp and PgDn keys may be used to move backwards and forwards through the database. This command is normally used after the desired record is found by using a LOCATE, SEEK or GOTO command.

To save and exit edit press CTRL-W.

BROWSE

This is a full screen option that allows many records to be displayed on the screen at one time. The browse option allows data to be edited and appended. To edit, position the cursor on the required field and change it. To append data move the cursor beyond the end of the file, you will be asked whether you wish to append new records, answer 'Y'.

Browse has a menu bar which may be accessed by holding down the CTRL key and pressing HOME (on the Olivetti's the HOME key is the 7 on the numeric keypad). The menu bar allows you to move to the TOP or BOTTOM of the file, to LOCK the display when panning across a record and to FREEZE the cursor onto a particular field.

To pan across a record hold down the CTRL key and press the right arrow key.

To save and exit browse press CTRL-W.

Getting a Listing or a Report:

LIST

List can be used to view the contents of a database file. To pause a list press CTRL-S. Press any key to continue. To list to the printer enter:

LIST TO PRINT

To abandon a LIST, press ESC.

Notes: _____

_____

CREATE REPORT

Use to produce a report of the contents of the database in use. Enter the following command:

CREATE REPORT name (eg. CREATE REPORT YWDATA)

The create report screen will appear. There is a menu bar at the top of the screen that takes you through a series of options for creating the report.

Once you have finished creating the report produce a printout by entering:

REPORT FORM name TO PRINT (eg. REPORT FORM YWDATA TO PRINT)

To make modifications to an existing report use the MODIFY REPORT command.

MODIFY REPORT name (eg. MODIFY REPORT YWDATA)

All records in the database are included in the report unless otherwise specified by a FOR or WHILE condition.

Deleting Records:

DELETE

Marks the records in the active database for deletion. Only the current record is marked for deletion unless a FOR or WHILE statement is used. With LIST records marked for deletion are indicated by an asterisk (*) in the first position of the record. With BROWSE or EDIT, records marked for deletion are indicated by DEL on the status bar. In BROWSE or EDIT mode CTRL-U both deletes and reinstates records. Records marked for deletion are not removed from the system until the PACK command has been issued.

RECALL

Reinstates records that are marked for deletion in the active database. Only the current record is reinstated unless a FOR or WHILE statement if used.

PACK

Removes records marked for deletion from the current database. Once records have been PACKED they cannot be RECALLED.

Notes: _____

_____

_____

Searching for Records:

GO

Positions the record pointer to a specified record in the active database.

GO TOP - goes to first record in database
GO BOTTOM - goes to last record in database

LOCATE

Searches the active database file for a record that satisfies specified criteria. If a match is found, the record pointer moves to that record and displays the record number.

Eg. LOCATE FOR plot = 99

To find the next occurence of the specified condition use the CONTINUE command.

Sorting Records:

SORT

Creates a new database file in which the records of the active database are reordered alphabetically, chronologically, or numerically by the specified key fields.

SORT TO <newfile> ON <field list>

Eg. SORT TO YWDATA1 ON PLOT, TREE_NO

This will sort the database YWDATA numerically by plot and by tree number into a new database called YWDATA1.

SET commands:

There are numerous SET commands which can be used to control the Dbase III/DBXL environment.

SET

Is a full-screen, menu-driven command for displaying and changing the values of many SET commands. Some of the more common ones are:

SET CARRY ON (copies data from last record into new record)
SET STATUS ON (displays status bar at bottom of screen)
SET BELL ON (bell rings during data entry if field filled)

Notes: _____

_____

Other Commands:

CLOSE ALL

Closes all files. Use this command before removing a floppy disk during a Dbase/DBXL session.

DIR

Displays filenames, number of records, date of last update, and file size (in bytes) for each database file on the indicated disk. Eg. DIR A:

HELP

A menu driven command that provides information about the commands in and use of Dbase III/DBXL.

QUIT

Closes all open files, terminated the Dbase III/DBXL session and returns control to the operating system. If you are working from a floppy disk DO NOT remove your disk until you have QUIT out of Dbase/DBXL.

Notes: _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

STORING DATA:

Data files can be created and edited on a floppy disk. - always making sure that you have a backup copy of your file, preferably on another floppy disk. Thus every data file should exist on two separate floppy disks, one as a working copy and one as a backup. For large files it may be more convenient to work from the hard disk, this may be done by copying your working copy of the data file onto the hard disk. Data files should only exist temporarily on the hard disk and should be deleted when you have finished working on them. They must NOT be stored permanently on the hard disk.

Copying Files:

To copy a file from one floppy disk onto another it is easier to copy the file from your source floppy onto the hard disk and then to copy the file from the hard disk onto the new target floppy.

Never remove a floppy disk while the red light is on.

Use the DOS copy command to copy files. Get into the Operating System and at the DOS prompt enter:

COPY A:filename.ext (eg. COPY A:YWDATA.DAT)

This will copy the file from the floppy disk onto the hard disk (C:).

If you want to copy the file to another floppy (ie BACKUP DISK), remove your current floppy (ie WORKING DISK), and replace it with your new floppy (ie. BACKUP DISK). You now copy the same file from the hard disk back onto the new floppy:

COPY filename.ext A: (eg. COPY YWDATA.DAT A:)

This will copy the file from the hard disk (C:) onto the floppy disk in drive A:

You should now have three copies of your file, one on a WORKING DISK, one on a BACKUP disk and one on the hard disk. Put the BACKUP disk in a safe place - preferably in the BACKUP diskette holder in the ARCHIVES. You may now work with your data file on either the WORKING disk or the hard disk.

If you make any changes/corrections to your data file you MUST update your BACKUP copy (and your WORKING disk). The copy of your data file on the hard disk is only TEMPORARY and you MUST delete it when you have finished working with it.

Notes: _____

_____

_____

Deleting Files:

To delete a file from the hard disk (C:) or a floppy disk get into the Operating System and at the DOS prompt enter:

DEL filename.ext (eg. DEL YWDATA.DAT)

This will delete the file from the hard disk (C:). To delete files from a floppy you MUST include the drive specifier in the filename.

DEL d:filname.ext (eg. DEL A:YWDATA.DAT)

This will delete the file YWDATA.DAT from the floppy in drive A:.

Data Library:

Manjimup Research Centre has a data library situated in the fire proof ARCHIVES room. Copies of all data sheets should be placed in this library.

Once you have put your data onto the computer and made your backup you need to update the data library and update your filing system. The following should be updated/completed:

(i) Data Identification Sheet (photocopy)
(ii) Variable Description Sheet (photocopy)
(iii) Code Description Sheet (photocopy)
(iv) Photocopy of original data sheets
(v) File Summary Sheet - update

(i) to (iv) should be organised into a loose leaf folder and filed in the Data Library. An INDEX FILE is located in the Data Library and should be updated when you add a new data file. The File Summary Sheet and the photocopies of (i) to (iii) can be placed in your own filing system along with the original data for future reference.

Notes: _____

_____

_____

_____

_____

_____

_____

_____

DATA ANALYSIS:

Using SAS:

SAS is a software system for data analysis. The SAS system works with data which must be in a SAS data set to use SAS procedures to analyze them.

SAS can be accessed directly from the main menu. After SAS is loaded three windows will be displayed, OUTPUT, LOG and PROGRAM. The PROGRAM window is where you enter SAS programs and execute them. The LOG window displays messages from SAS as well as your SAS program statements as they are executed. The OUTPUT window is where the results of the SAS procedures are displayed.

Move between the windows using the arrow keys or PgUp, PgDn keys. Commands may be entered at the COMMAND-> prompt or you may use the function keys. Some more useful commands are:

CLEAR — Removes all information from current window.

ENDSAS — Ends SAS. You can also use BYE.

FILE — Writes the entire contents of the current window into an external file. The FILE statment can be used in any of the LOG, OUTPUT or PROGRAM windows. Include the drive specifier in your filename if you wish to write onto the A: drive. Enclose the entire filename in quotes.

FILE "d:filename" (Eg. FILE "A:K10VEG.OUT")

The FILE command can also be used to send output to the printer. Use the command: FILE "PRN" .

INCLUDE — Allows you to bring an external file into the PROGRAM EDITOR. You must include the drive specifier if your file exists on the A: drive. Enclose the entire filename in quotes.

INCLUDE "d:filename" (Eg. INCLUDE "A:K10VEG.SAS")

QUIT — Sometimes if SAS gets a hangup about some procedure it won't work until you QUIT. To QUIT create a program with 1 line with QUIT; on it and submit it.

RECALL — Brings back to the PRG window any lines you have submitted to SAS.

SUBMIT — Submits SAS program statements from the PGM window for execution.

ZOOM — Displays the current window over full screen.

SAS program statements are entered at the numbered lines in the PROGRAM EDITOR window. These statements must always end with a semicolon.

Loading an ASCII File into SAS:

To load an ASCII data file on disk into a SAS data set, you need the following SAS statements:

DATA statement
INFILE statement
INPUT statement
RUN statement

DATA

Use the DATA statement to tell SAS that you want to create a SAS data set. The DATA statement begins with the word DATA and then the name of the data set you are creating. Generally use the same filename as you are already using.

```
DATA name;      (eg. DATA K10VEG;)
```

INFILE

The INFILE statement identifies an external file to SAS. The filename should include the drive specifier if the file is on a floppy disk and be enclosed within quotes.

```
INFILE filename;     (eg. INFILE "A:K10VEG.DAT";)
```

If you have MISSING VALUES in your data set, ie. variables that are BLANK, you should include the MISSOVER option in your INFILE statement. MISSOVER prevents a SAS program from going to a next input line if it does not find values in the current line for all the INPUT statement variables. When an INPUT statement reaches the end of the current record, values that are expected but not found are set to missing.

```
Eg. DATA WEATHER;
          INFILE "A:YWTEMP.DAT" MISSOVER;
          INPUT TEMP1 1-4 TEMP2 6-9 TEMP3 11-14 TEMP4 16-19
                TEMP5 21-24;

     <A:YWTEMP.DAT>
     97.9 98.1 98.3
     98.6 99.2 99.1 98.5 97.5
     97.5 87.8 92.5 90.0 89.4
```

SAS will read the three values on the first data line as values of TEMP1, TEMP2 and TEMP3. The MISSOVER option causes SAS to set the values of TEMP4 and TEMP5 to missing for that observation because there are no values for those variables in the current input line.

If the MISSOVER option was not used, SAS would go to the second data line for the TEMP4 and TEMP5 values and then read data line 3 as the next (second) record.

INPUT

The INPUT statement describes each record in your data file to SAS. List each variable, if the variable contains character data include a $ sign after the variables name. After each variable (and $ sign if applicable) list the columns that the variable occupies. If your INPUT statement continues to a second line, do not split a variable name between the two lines.

```
INPUT variable $ column(s);

Eg. INPUT plot 1-3 dist 4 date 5-10 sp_no 11-14 sp_name $ 15-22
          herb_no 23-26 abund 27 height 28-31;
```

RUN

When you want SAS to execute your statements use a RUN statement.

```
RUN;
```

Example of SAS program to read in ASCII data:

```
     * Program K10VEG
     *
     * Karri Site Type 10 - Vegetation Assessment
     *
     * Ian Wheeler,  June 1989
     *;
     DATA K10VEG;
     INFILE "A:K10VEG.DAT" MISSOVER;
     INPUT plot 1-3 dist 4 date 5-10 sp_no 11-14 sp_name $ 15-22
           herb_no 23-26 abund 27 height 28-31;
     RUN;
```

Comments can be entered anywhere in your SAS program to document the purpose of the program, or provide any other relevant information. The form of the comment statement is:

```
*message;
```

where "message" explains or documents the program. The message can be any length, although it cannot contain semicolons. The comment statement must end in a semicolon.

Always document your programs detailing the:

- Programs name
- Programs purpose
- Author and date written

Loading a Dbase III/DBXL File into SAS:

To load a Dbase III/DBXL file on disk into a SAS data set, you need the following SAS statements:

FILENAME statement
PROC DBF statement
DATA statement
SET statement
RUN statement

FILENAME

FILENAME defines the file to SAS. It associates a SAS fileref (file reference name) with an external files complete name (including drive).

FILENAME fileref "filename";

Eg. FILENAME PHPOOL "A:PHPOOL89.DBF";

Where PHPOOL is the SAS fileref, and "A:PHPOOL89.DBF" describes a Dbase III file on the A: drive called PHPOOL89.DBF.

PROC DBF

Converts Dbase III/DBXL files into SAS data sets.

PROC DBF DB3=fileref OUT=name;

DB3=fileref specifies the fileref of a Dbase III or DBXL file. The fileref is defined in the FILENAME statement.

OUT=name specifies the name of the SAS data set created to hold the converted data.

Numeric variables in Dbase III become SAS numeric variables. Character variables become SAS character variables. Dbase III allows 10 character variable names. Variable names will be truncated to a length of 8 when converting Dbase to a SAS data set.

Eg. PROC DBF DB3=PHPOOL OUT=TEMP;

DATA

Use the DATA statement to tell SAS that you want to create a SAS data set. The data statement begins with the word DATA and then the name of the data set you are creating.

DATA name; (eg. DATA PHPOOL;)

SET

The SET statement tells SAS to read data from another SAS data set. Use SET to read records from existing SAS data sets into a new SAS data set.

SET name; (Eg. SET TEMP;)

RUN

When you want SAS to execute your statements use a RUN statement.

RUN;

Example of SAS program to read in Dbase III data:

```
* Program PHPOOL
*
* Penni Hewett - Poole Thinning Expt. 1989
*
* Penni Hewett, June 1989
*;
FILENAME PHPOOL "A:PHPOOL89.DBF";
PROC DBF DB3=PHPOOL OUT=TEMP;
*
*Dbase III file converted to SAS data set called TEMP
*;
DATA PHPOOL;
SET TEMP;
RUN;
```

This program converts a Dbase III file (A:PHPOOL89.DBF) into a SAS data set called TEMP. A new SAS data set is then created called PHPOOL and records are copied into it from TEMP.

Notes: _____

Analysing and Checking your Data:

Once you have created a SAS data set you are ready to use SAS procedures to analyze and process that data. SAS procedures are programs that read your SAS data set, perform various manipulations and calculations, and display the results.

PROC PRINT

The PRINT procedure prints the data set, using all or some of the variables. Totals and subtotals for numeric variables can also be printed.

```
PROC PRINT;
  VAR variabls;
  BY variables;
  SUM variables;
```

VAR statement	names variables to be printed. If no VAR statement is used all variables are printed.
BY statement	used to obtain separate analysis in groups defined by the BY variable. When you use a BY statement SAS expects the data set to be sorted in order of the BY variables.
SUM statement	specifies variables whose values are to be totalled. When a BY variable is specified, the SUM variable is totalled for each BY group.

PROC SORT

The SORT procedure sorts records in a SAS data set by one or more variables. Data sets must be sorted before you can process it using BY statements. When you want to sort by two or more variables, PROC SORT arranges the data set in the order of the first BY variable. Then SORT arranges the records having the lowest value of the first variable in the order of the second variable. This continues for every BY variable specified.

```
PROC SORT;
  BY variables;
```

PROC MEANS

The MEANS procedure produces simple univariate descriptive statistics for numeric variables. If you use a BY statement, PROC MEANS calculates descriptive statistics separately for groups of observations. The MEANS procedure is the easiest and most direct method of obtaining descriptive statistics for numeric data.

```
PROC MEANS options;
  VAR variables;
  BY variables;
```

Options allows you to select which statistic you want printed:

N	number of observations
NMISS	number of missing observations
MEAN	mean
STD	standard deviation
MIN	minimum value
MAX	maximum value
RANGE	range
SUM	sum
VAR	variance
USS	uncorrected sum of squares
CSS	corrected sum of squres
CV	coefficient of variation
STDERR	standard error of the mean
T	Student's T value for testing the hypothesis that the mean is 0
PRT	probability of a greater absolute value for the Student's T value

When no statistics are included as options all of the above are printed.

VAR statement	lists variables to be analysed. If no VAR statement all numeric variables are analysed.
BY statement	used to obtain separate analysis in groups defined by the BY statement. SAS expects data to be sorted in order of the BY variables.

PROC UNIVARIATE

The UNIVARIATE procedure produces simple descriptive statistics for numeric variables. It provides great detail on the distribution of the variable including:

detail on extreme values
mean, median
plots to picture the distribution
frequency tables
test for normal distribution

```
PROC UNIVARIATE options;
  VAR variables;
```

Options that can be included in the UNIVARIATE procedure:

PLOT	prints stem and leaf plot, box plot and normal probability plot.
FREQ	prints frequency table.
NORMAL	computes test statistic (W) to test if data comes from a normal distribution.
VAR statement	lists variables to be analysed. If no VAR statement all numeric variables are analysed.

The UNIVARIATE procedure produces copious amounts of information which can become confusing, an easier method of obtaining descriptive statistics is to use the MEANS procedure.

PROC FREQ

The FREQ procedure produces frequency and crosstabulation tables. Frequency tables show the distribution of variable values. Crosstabulation tables show combined frequency distributions for two or more variables.

```
PROC FREQ
  TABLES requests / options;
  BY variables;
```

TABLES	any number of TABLES statements can be included in one PROC FREQ.
requests	one or more variable names joined by asterisks (*). Eg. TABLES plot*sp_no

Options that can be included in the TABLES statement:

NOFREQ	suppress printing cell frequencies
NOPERCENT	suppress printing cell percentages
NOROW	suppress printing row percentages
NOCOL	suppress printing column percentages
NOCUM	suppress printing cumulative frequencies and cumulative percentages

When no options are specified FREQ produces crosstabulation tables that include cell frequencies, cell percentages of total frequency, cell percentages of row frequencies, and cell percentages of column frequencies.

Organising your Data:

SAS has many commands that allow you to modify your data set. For example you may wish to exclude certain observations from your analysis. Two useful commands for performing this are:

IF

The IF statement can be used to cause SAS to continue processing only those records that meet the condition specified in the IF clause.

IF expression; (Eg. IF year = 1989;)

The IF statement can also be used with a THEN clause, to execute a SAS statement only for observations that meet the condition specified.

IF expression THEN statement; (Eg. IF year = 1989 THEN DELETE;)

Thus SAS deletes all records from the current SAS data set when its value for year = 1989.

DELETE

A delete statement tells SAS to stop processing the current record.

DELETE;

For example:

```
DATA PHPOOL;
  INPUT plot 1-3 year 4-7 tree 8-10 dbhob 11-15
        bark1 16-17 bark2 18-19 bark3 20-21;
  IF year = 1989 THEN DELETE;
  avbark = (bark1 + bark2 + bark3)/3;
  RUN;
```

In this example, records where year equals 1989 are not included in the SAS data set PHPOOL. The calculation of avbark is not executed for those records.

Notes: _____

_____

_____

_____

_____

_____

Sample Program:

```
* Program K10VEG
*
* Karri Site Type 10 - Vegetation Analysis
*
* Ian Wheeler, June 1989
*;
DATA K10VEG;
INFILE "A:K10VEG.DAT" MISSOVER;
INPUT plot 1-3 dist 4 date 5-10 sp_no 11-14 sp_name $ 15-22
      herb_no 23-26 abund 27 height 28-31;
RUN;
*
* Sort data by plot number
*;
PROC SORT;
  BY plot;
RUN;
*
* Print frequency table of veg. species in each plot
*;
PROC FREQ;
  TABLES plot*sp_no / nopercent norow nocol;
RUN;
*
* Print average height of vegetation in each plot
* excluding Karri (sp_no = 183)
*;
IF sp_no = 183 THEN DELETE;
PROC MEANS MEAN STD MIN MAX STDERR;
  VAR height;
  BY plot;
RUN;
```

This job is submitted to SAS by entering SUBMIT at the command prompt in the PROGRAM window, or by pressing F10.

To print the results when the job has finished running, move the cursor to the OUTPUT window and at the command prompt and enter FILE "PRN".

To print a copy of your program, move the cursor to the PROGRAM window and enter RECALL at the command prompt, or press F9. Your program should now be displayed. Enter FILE "PRN" at the command prompt to obtain a printout.

To save a copy of your program to disk, move to the program window and RECALL your program so that it is dispayed on the screen. Enter FILE "A:filename" at the command prompt, where filename is the name of your program with a .SAS extension (Eg. FILE "A:K10VEG.SAS").

ARCHIVING DATA

If you have finished working with your data it can be archived. Please see me if you have any files that you would like archived. There are two methods of archiving:

Archiving onto the Perkin Elmer:

ASCII data files can be copied onto the Perkin Elmer and then archived onto the PE Archive Disk and magnetic tape.

Archiving onto Tape:

Any file can be archived onto the NEC Powermate in the Computer Room using the ARCHIVE option from the Main Menu. This will copy the files into a subdirectory called ARCHIVE. This subdirectory with then be backed up to streaming tape and stored in the ARCHIVES room.

TINGLE STUDY RWP 44/88 - STAND PARAMETERS

QUADRAT: Enter quadrat number

QUADRAT SIZE:
(1) 20x 20m
(2) 40 x 49m
(3) 80 x 80m

DATE: Enter DDMMYY.

SPECIES NUMBER:

(180) E. brevistylis	(9) Acacia pentadenia
(181) E. calophyla	(20) Agonis flexuosa
(183) E. diversicolor	(22) Agonis juniperina
(184) E. ficifolia	(28) Allocas. decussata
(185) E. guilfoylei	(29) Allocas. fraseriana
(186) E. jacksonii	(51) Banksia grandis
(187) E. marginata	(52) Banksia ilicifolia
(188) E. megacarpa	(53) Banksia littoralis
(189) E. patens	(55) Banksia seminuda
(191) E. staeri	(285) Trymalium floribundum

SPECIES NAME: First 3 letters of Genus and Species.

DBHOB: All stems of all species >10cm DBHOB in permanent 40 x 40m quadrat - at 1.3m on uphill side.

CIRCUMFERENCE: At 1.3m

DOMINANCE:
(1) Dominant crown - access to light from sides and from above.
(2) Co-dominant - access to light from above.
(3) Sub-dominant - restricted access from above.
(4) Suppressed - no access to direct sunlight.

LENGTH DRY SIDE: At 1.3m - lenght of circumference.

ASPECT DRY SIDE: From centre of dry side - to nearest 10°

HOLLOW BUTT?: Yes = 1

BARK THICKNESS
KARRI - measured 4 times on ridges at 20cm above breast height.
TINGLE - measured 6 times on ridges at 20cm above breast height.

Field
QUADRAT
QUADRAT SIZE
DATE
SPECIES NUMBER
SPECIES NAME
TREE NUMBER
DBHOB (CM)
CIRCUMFERENCE (CM)
DOMINANCE
LENGTH OF DRY SIDE 1 (CM)
ASPECT OF DRY SIDE 1
LENGTH OF DRY SIDE 2 (CM)
ASPECT OF DRY SIDE 2
LENGTH OF DRY SIDE 3 (CM)
ASPECT OF DRY SIDE 3
HOLLOW BUTT?
BARK THICKNESS (MM) (1)
BARK THICKNESS (MM) (2)
BARK THICKNESS (MM) (3)
BARK THICKNESS (MM) (4)
BARK THICKNESS (MM) (5)
BARK THICKNESS (MM) (6)

STAND PARAMETERS

RWP 44/88 TINGLE STUDY 185

STAND PARAMETERS

QUADRAT	QUADRAT SIZE	DATE	SPECIES NUMBER	SPECIES NAME	TREE NUMBER	DBHOB (CM)	CIRCUMFERENCE (CM)	DOMINANCE	LENGTH OF DRY SIDE 1 (CM)	ASPECT OF DRY SIDE 1	LENGTH OF DRY SIDE 2 (CM)	ASPECT OF DRY SIDE 2	LENGTH OF DRY SIDE 3 (CM)	ASPECT OF DRY SIDE 3	HOLLOW BUTT?	BARK THICKNESS (MM) (1)	BARK THICKNESS (MM) (2)	BARK THICKNESS (MM) (3)	BARK THICKNESS (MM) (4)	BARK THICKNESS (MM) (5)	BARK THICKNESS (MM) (6)
139	1	24Ø589	285	TRY FLØ	1	11.4	35.5									3	3	3	4		
			28	ALL DEC	2	11.6	37.Ø									13	18	25	12		
			9	ACA PEN	3	1Ø.4	32.Ø									3	4	4	4		
			28	ALL DEC	4	19.1	61.Ø									31	3Ø	23	28		
			285	TRY FLØ	5	12.Ø	37.5									4	4	3	3		
			185	EUC GUI	6	72.5	228.Ø	1								4Ø	34	39	52		
			185	EUC GUI	7	43.2	136.Ø	3	32	27Ø						43	44	38	37		
			28	ALL DEC	8	64.9	2Ø3.5									37	62	54	7Ø		
			285	TRY FLØ	9	15.7	49.9									4	3	3	3		
	2		28	ALL DEC	1Ø	51.3	16Ø.5									75	82	75	68		
			185	EUC GUI	11	76.	238.8	1								26	33	24	34		
			28	ALL DEC	12	14.4	45.Ø									22	21	21	23		
			185	EUC GUI	13	98.3	3Ø8.9	1								32	33	3Ø	31		
			185	EUC GUI	14	84.7	266.2	2								28	28	33	27		
			185	EUC GUI	15	84.7	266.Ø	3								32	36	34	36		
			285	TRY FLØ	16	11.1	35.3									3	3	3	3		
			185	EUC GUI	17	51.9	163.Ø	4	65.	4Ø						24	23	24	26		
			285	TRY FLØ	18	11.Ø	34.9									3	3	3	3		
			285	TRY FLØ	19	13.8	43.5									4	3	4	4		
			28	ALL DEC	2Ø	63.5	199.5									65	75	6Ø	62		
			28	ALL DEC	21	38.Ø	119.Ø									28	35	27	42		
			28	ALL DEC	22	4Ø.Ø	125.Ø									74	76	85	74		
			28	ALL DEC	23	71.4	224.Ø									48	52	64	93		
			185	EUC GUI	24	37.4	118.Ø	3								28	29	34	28		
			28	ALL DEC	25	25.Ø	78.4									29	34	35	37		

Structure for database: a:tinsta89.dbf
Number of data records: 232
Date of last update : 30/06/89

Field	Field Name	Type	Width	Dec
1	QUADRAT	Numeric	3	
2	QUADSIZE	Numeric	1	
3	DATE	Numeric	6	
4	SPECNO	Numeric	4	
5	TREENO	Numeric	3	
6	DBHOB	Numeric	5	1
7	CIRCUM	Numeric	6	1
8	DOM	Numeric	1	
9	LENDRY1	Numeric	5	1
10	ASPDRY1	Numeric	3	
11	LENDRY2	Numeric	5	1
12	ASPDRY2	Numeric	3	
13	LENDRY3	Numeric	5	1
14	ASPDRY3	Numeric	3	
15	HOLBUTT	Numeric	1	
16	BARK1	Numeric	2	
17	BARK2	Numeric	2	
18	BARK3	Numeric	2	
19	BARK4	Numeric	2	
** Total **			63	

Department of Conservation and Land Management
Manjimup Research Centre

DATA IDENTIFICATION SHEET

Research Working Plan: 44/88

Project Title: TINGLE STUDY: THE ECOLOGY AND BIOGEOGRAPHY OF FOUR SPECIES OF FOREST EUCALYPTS OCCURRING AS NARROW ENDEMICS

Project OIC: GRANT WARDELL-JOHNSON

Data Filename: TINSTA89.DBF

No. Records: 232

Description: STAND PARAMETERS, JUNE 1989

Data OIC: CHRIS VELLIOS

Date: 6/7/89

Data Location: 5¼" DISK - TINGLE STAND DATA

Backup Location: 5¼" DISK - TINGLE STAND DATA - BACKUP

Archive Location:

Comments:

Department of Conservation and Land Management
Manjimup Research Centre

VARIABLE DESCRIPTION SHEET

VARIABLE	COLUMNS	DESCRIPTION	UNITS
QUADRAT	1-3		
QUADSIZE	4		
DATE	5-10	(DDMMYY)	
SPECNO	11-14	SPECIES NUMBER	
SPECNAME	15-22	SPECIES NAME (ABBREVIATED)	
TREENO	23-24	TREE NUMBER	
DBHOB	25-29	DIAMETER BREAST HEIGHT OVER BARK	CM
CIRCUM	30-35	CIRCUMFERENCE	CM
DOM	36	DOMINANCE	
LENDRY1	37-41	LENGTH OF DRY SIDE 1	CM
ASPDRY1	42-44	ASPECT OF DRY SIDE 1	
LENDRY2	45-49	LENGTH OF DRY SIDE 2	CM
ASPDRY2	50-52	ASPECT OF DRY SIDE 2	
LENDRY3	53-57	LENGTH OF DRY SIDE 3	CM
ASPDRY3	58-60	ASPECT OF DRY SIDE 3	
HOLBUTT	61	HOLLOW BUTT (1=YES)	
BARK1	62-63	BARK THICKNESS	MM
BARK2	64-65	" "	MM
BARK3	66-67	" "	MM
BARK4	68-69	" "	MM
BARK5	70-71	" "	MM
BARK6	72-73	" "	MM

Department of Conservation and Land Management
Manjimup Research Centre

VARIABLE CODE SHEET

VARIABLE	CODE	DESCRIPTION
QUADSIZE	1	20 x 20m
	2	40 x 40m
	3	80 x 80m
SPECNO		SEE VEG. LIST (TONY ANNELS)
DOM	1	DOMINANT
	2	CO-DOMINANT
	3	SUB-DOMINANT
	4	SUPPRESSED
HOLBUTT	1	YES

Department of Conservation and Land Management
Manjimup Research Centre

FILE SUMMARY SHEET

Filename Code: TINSTA-

General Description: TINGLE STUDY, STAND PARAMETERS JUNE 1989

Date	Filename	Description/purpose	No. Recs.	OIC	Data Disk	Backup Disk
6/7/89	TINSTA89.DBF	1989 DATA - DBASE III	232	CV	5¼ TINGLE STAND	5¼ TINGLE STAND
6/7/89	TINSTA.SAS	SAS command file to analyse TINSTA89.DBF.		CV	11	11

```
* Program TINSTA.SAS
*
* Tingle Study : Stand Parameters 1989
*
* Chris Vellios : June 1989
*
*;
FILENAME TINSTA "A:TINSTA89.DBF";
PROC DBF DB3=TINSTA OUT=TEMP;
*
* Dbase III file converted to SAS data set called TEMP
*;
DATA TINSTA;
SET TEMP;
*
* Calculate average bark thickness
*;
AVBARK = (BARK1+BARK2+BARK3+BARK4)/4;
*
* Select only E. guilfoylei (185)
*;
IF SPECNO = 185;
*
* Sort database by quadrat number
*;
PROC SORT;
  BY QUADRAT;
  RUN;
*
* Descriptive stats for E. guilfoylei by quadrat
*;
PROC MEANS N MEAN STD MIN MAX VAR STDERR;
  VAR DBHOB CIRCUM AVBARK;
  BY QUADRAT;
  RUN;
```