PRENTICE COMPUTER CENTRE

University of Queensland

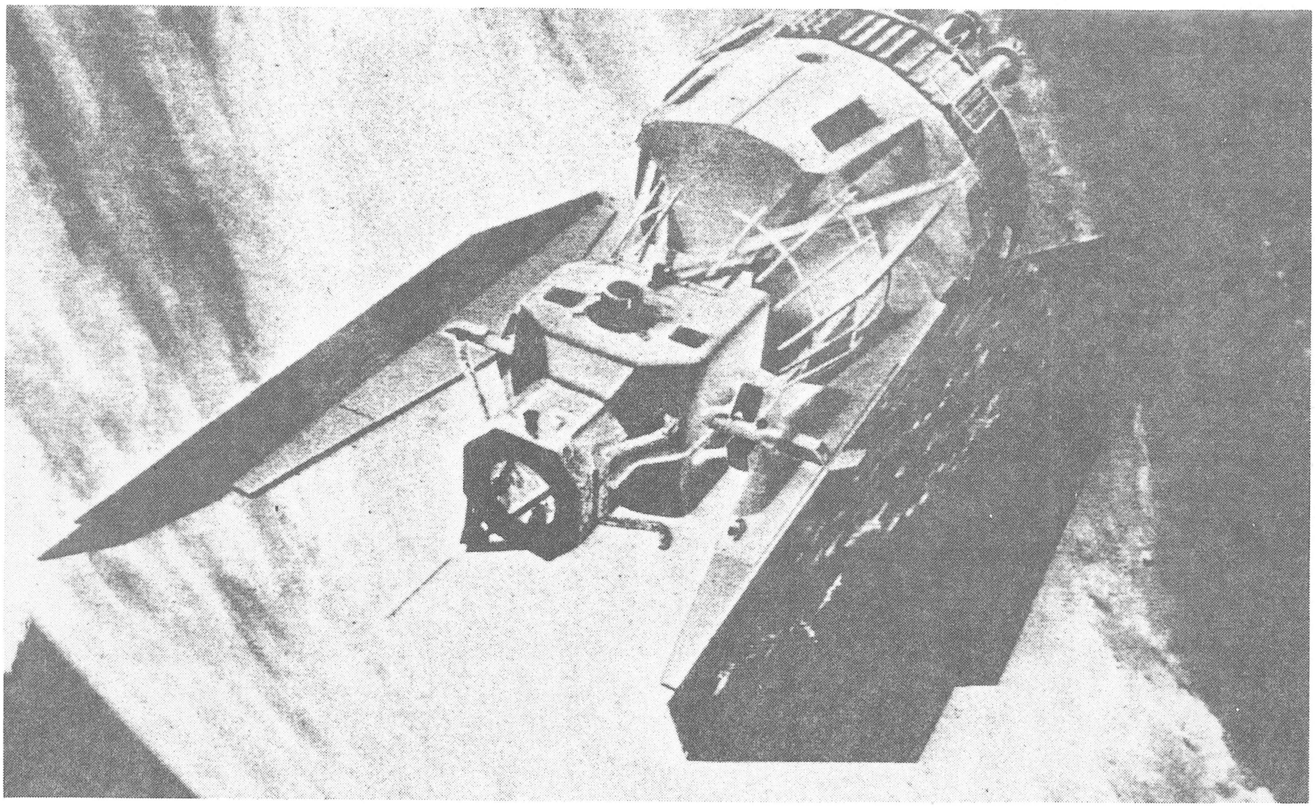
TECHNICAL MANUAL No. 5

VG USER GUIDE

MNT-5/ed 2
February 1981

This manual has been authorized by the Director of the Prentice Computer Centre

VG



A Text-oriented Data-base System

G.A. Vandenberg

TECHNICAL MANUAL NUMBER 5

VG USERS MANUAL

(VG Version II)

Revised Edition: 1981

G.A. Vandenberg
Barry Maher

Prentice Computer Centre
University of Queensland
Australia

CONTENTS

Conventions Used in this Manual

The reader should note that the following standards are used throughout this manual:

Reserved Symbols and their Meanings

(i) © denotes Control Key,
e.g. ©**Z** indicates that the key marked CTRL (Control) is held depressed while the key Z is struck.

(ii) ® denotes Carriage Return Key.

(In previous editions of this manual, these have been denoted by ↑ and <cr> respectively.)

Worked Examples

In all worked examples in this manual, the following conventions apply:

(i) Medium face — indicates the terminal's response, e.g.

```
*** VG ***

Enter Master File name :
```

is the response displayed on the screen after the user has issued the instruction to run the VG program.

(ii) **Bold face** — indicates the instruction/text typed by the user, e.g.

R VG®

denotes what the user must type at the terminal in order to run the VG program.

(iii) *Italic face* — indicates a comment or explanatory note, e.g.

Select the UPDATE function

Thus a complete line within a worked example could appear:

```
Output to 'File' or 'Tty' (F or T) : T®          Output to terminal is requested
                    ↑                 ↑                        ↑
                 Terminal            User                   Comment
```

CHAPTER 1

INTRODUCTION TO VG

This chapter introduces VG, briefly describes the method of use and includes a summary of all functions.

VG is a data-storage-and-retrieval system for 'textual'-oriented data. It is made up of a suite of programs that are *menu*-controlled. Its uses include name and address files, student record systems, medical history records, bibliographies, catalogues, equipment registers and so on.

No programming knowledge is required to use VG. The user designs his own 'record profile' by way of 'field declarations'. Up to 77 such field declarations are allowed. Up to 20,000 characters may be stored within any one record. Disk-storage requirements are minimal (i.e. there is very little wasted space). There is a nominal upper limit of 99,999 to the number of records in any one file.

Features available include a powerful interactive editor (similar to LINED), file sorting, a search and select option, 'column'-type listings, 'across'-type listings, adhesive labels, tabulations on string values, form letters, fast terminal enquiries, plus the ability to remember report formats for repeated usage.

VG has inbuilt safeguards to prevent file corruption in the event of power failures, etc.

New users of VG will find Chapter 4 ('Creating New VG Files') adequately describes how to begin using VG. In the case of existing files in other formats, such as card decks and so on, a conversion procedure is available from the author at the Prentice Computer Centre. There is no general conversion program because of the wide variation in medium for file storage.

There is a chapter devoted to examples of VG files for various applications. It is advisable to examine these before starting off new.

VG was created at the Prentice Computer Centre, University of Queensland, St Lucia, Brisbane, Australia in 1978, and upgraded to Version II in 1979.

1.1 Technical Information

For those who are interested, the files are indexed random access with unformatted image mode variable length records. No attempt should be made to alter a file with any text-editor (such as TECO, SOS, or LINED) other than the editor contained within VG, as *severe damage will result*.

Program languages used are Fortran-10 and Macro-10.

VG will run only on a PDP-10 (KA, KI, KL) processor.

VG does no type-checking, i.e. integers and ascii are *not* distinguished.

1.2 Method of Use

VG runs interactively on a PDP-10 computer via a *menu*-system, by typing the command

```
R VG®
```

The system responds with the question

```
Enter Master File Name
```

to which the response is the name of the VG file required.

To select a specific function, the name of the function is entered. To obtain a list of all functions, type ®.

A Master File in VG is the name used to descibe a user's defined file. If no Master File has been defined, then the user can select from the 'function list' only the NEW option. The NEW function is used for creating a new VG file.

After performing the required function, type ©**Z** (control-Z) to exit from the Menu. Most output from VG is generated in the form of an output file on the user's area for printing on the high-speed lineprinter. In most cases, an automatic print procedure is invoked to allow the output to be printed on the lineprinter with correct stationery and limits. Short reports may be typed on a terminal.

1.3 Main-menu Functions

BIBLIO Used to call up a sub-menu of commands suitable especially for bibliographic material. See Section **1.4** for a brief description of the functions available within this bibliographic option.

COLREP Column-type reports are generated by this function. The user selects the width of each column, which field goes into which column, and a sub-heading option with index.

EDIT The edit function is used for changing or deleting existing records.

EXIT To release the VG menu and return to Monitor level.

FILES Displays all VG file names on the terminal.

GENER To create a new (but empty) Master File from an existing Master File.

IBUILD Rebuilds an Index if lost or corrupted.

INFORM Information about the current VG file being used. Displays field names and number of records on file.

JOINER Allows user to combine a number of existing VG Master Files into a single Master File.

KJOB Logs user off the PDP-10 system.

LABELS Adhesive labels are produced using selected fields from each record. The labels are 45 characters wide by 5 lines deep.

LETTER For the production of 'form letters', using the RUNOFF program formatting. Can also be used to produce many varied layouts of reports.

MATFRM Allows contents of selected fields to be output as a formatted data file. Suitable where further processing by other programs is required.

NEW Used to define a new VG file record profile format.

REFRES Used to reorganise a VG master file after extensive editing, particularly when records have been deleted or increased in size.

REPORT Report is used for interactive enquiries, with limits on the fields typed out. Output may also be directed to the line-printer via an output file.

SETS The SET function is used to create a special *index* which accesses only a selected portion of your file. This allows the performance of a range of functions on segments of a file without the need to use the 'Search and Select' feature every time. An example may be all those people who belong to a selection of postcodes such as in a name and address file. Then it would be more efficient to produce a set of adhesive labels, plus a column report, and perhaps some form letters from the one subset.

SORT Files may be sorted into an order dependent on the values contained in a maximum of four fields. This function also includes a fast 'keysort' option. VG files are indexed by a set of *keys.* These keys are extracted from the first ten characters of the first field in each record. If a file used Field (1) as SURNAME, then a key sort would order the keys into alphabetic order, effectively ordering the Master File. In fact, all sorting only re-orders the index.

SETSRT Sorts a selected *set.*

SWAP	To change from the current VG file being used to another VG file.
TABUL	Performs frequencies on ‘strings’ of characters from a field, one-way or two-way tabulation on the ‘strings’ in two fields.
UPDATE	The function for adding new records to an existing VG file.
©Z	Control-Z is the same as EXIT.

1.4 Biblio-menu Functions

BATKEY	Allows user to perform the KEYIND function (see below) as a batch job.
CREF	To generate a cross-reference listing.
EXIT	Identical to EXIT function in main menu, i.e. returns user to Monitor level.
EXTSRT	Similar to SORT function, except that the sorting is performed on a total of 100 characters rather than just 10 characters.
FORMAT	To produce a formatted output of bibliographic material.
FREQ	Produces a frequency listing from the Keyed Index.
INFORM	Identical to INFORM function in main menu.
KEYIND	To create a *keyword index* for a VG file.
KJOB	Identical to KJOB function in main menu.
QSEA	To search Keyed Index File for keywords (4500 limit).
SEARCH	Similar to QSEA function, except that limit is 17500.
SETSRT	Identical to SETSRT in main menu.
SWAP	Identical to main-menu function SWAP.
VG	To return to main VG menu.

CHAPTER 2

USING THE PDP-10

2.1 PDP-10 Terminology

This chapter is for new users of the PDP-10. It describes briefly how to use the PDP-10 and provides some background information about files, records, fields, random access, login, passwords, cost limits, PPN's, disk areas, quotas, and monitor commands.

The information is ordered on logical relationship, starting with files.

FILE — Is a collection of logically related information gathered in groups of individual records,
e.g. A file of all the names and addresses of the members of a club.

RECORD — A unit of information usually made up of *fields*,
e.g. in the above file there may be 100 members of the Club, so there would be 100 records in the file. One for each member.

FIELD — The term used to describe the components of a record,
e.g. Surname, Address, Phone Number may be described as being the fields of a name and address record.

DISK — A device for storage of computer files, based on the same principle as a tape recorder except magnetic disks are used. The main advantage of disks is their size and Random Access capability.

QUOTA — Each user is allocated 1500 *blocks* of storage space on the PDP-10 (KL-system). A block is the equivalent of 640 characters of information. You have about 960000 characters of available space!

RANDOM ACCESS — Is the name used to describe the selection of records from a file by a *key* rather than serial processing. This allows only those records of interest to be examined without the need to process every record up to the one of concern.

KEY — A value assigned to a Record or extracted from the record for retrieval and identification purposes.

INDEX — A collection of Keys used to store the location of a record in a master VG file. A Key may be a Surname or a Student Number and so on.

PPN — Is the computer identification for a user disk area. It is made up of two parts: Project Number and Programmer Number,
e.g [663,522] is a PPN.
The square brackets are standard notation. All disk files are stored under the user PPN.

PROGRAM — A collection of instructions for a computer, e.g. LOGIN is a program on the PDP-10 where the instructions are to identify users to the PDP-10, to ensure they are financial, and to obtain the correct password. Users may then proceed to use other programs like VG.

LOGIN — The procedure used to gain access to the PDP-10 computer.

PASSWORD — A secret code each user has to establish their identity with the PDP-10.

MONITOR — The Master program that runs all the time on the PDP-10. This program allows many users to use the PDP-10 simultaneously. It ensures that users do not interfere with each others work and performs various tasks for user programs.

MONITOR COMMAND — An instruction to the Monitor by a user, usually via a terminal. A list of frequently used Monitor Commands is contained in this chapter.

MONITOR LEVEL — When a user has no program running and is logged in he is at Monitor Level. Now all commands are directed to the Monitor. When an instruction is given to the Monitor to begin a program, and the program has started, the user is now at program level. Now all instructions will go direct to the program rather than the Monitor. When the program finishes, the user returns to Monitor level.

PROGRAM LEVEL
When a program is in execution, the user may have to respond to questions and provide information to a program. This is a level below Monitor level. When the program completes the user is returned to Monitor level.

MENU
The name used to describe a Master Control Program for a selected set of common programs. An example is the VG system. The VG menu program controls the use of each function. It usually performs checks to see that all relevant files necessary are present and inhibits actions which are foreign to the Menu's repertoire. The Menu makes life easier for the user by using meaningful names for functions.

LEVELS
There are various levels at which the user operates with the computer. Already discussed are Monitor level(the highest) then program level. With a *menu*-controlled system as in VG, there is a level refered to as MENU level. This level fits in between Monitor level and Program level. Frequently programs have various levels of their own, but these are usually self evident. It is important to be able to distinguish between Monitor level and Program or Menu level. This is usually achieved by the nature of the *prompt*. The Monitor Level is prompted with a '.' (period) and programs with '*' or ':' or some other distinguishable character.

FUNCTION
All available functions in the VG system are listed in Chapter 1.

2.2 Login

Once you have access to a terminal (TTY) you need to 'LOGIN'. The login procedure gives you access to all the PDP-10's facilities.

Example

```
.LOG 123,456®
Password:                            Enter your password
Cost limit:                          Enter a cost limit, e.g. 10.00
```

There will be some type-out here, but you have logged in.

2.3 Login Failure

Some reasons for failure to login:

Invalid password, check it.

Cost Limit exceeds balance, reduce cost limit or put more money in your account.

Job capacity exceeded, wait and try again later when PDP-10 is not so busy.

Password fails but is it correct?, then are you on the correct PDP-10, you should be on the 'KA' or the 'KL'. If you are not sure contact the Centre. Your terminal may be 'switched' to the wrong 'HOST' PDP-10, try **.SET HOST UQKA10** or **.SET HOST UQKL10**, whichever is required.

The stupid terminal does nothing at all? Is the power on? Is the terminal set to 'LOCAL'?—there is probably a switch for this on it. If it is then set the switch to 'REMOTE'. Perhaps the line 'SPEED' is different to that at which the terminal is set. Try setting the speed to a slower speed. To do this see if there is a 'Speed' control switch on the terminal. If you can find it, try setting it to 300 BAUD.

2.4 Monitor Commands

The term 'Monitor Command' is used to describe instructions to the PDP-10.

R PROG	Monitor command to cause execution of a program called PROG.
R VG	Begin execution of VG.
COST	Reports how much money has been spent.
PRINT MYFILE.DAT	Prints the file 'MYFILE' on the line-printer.
PRI/COP:2 MYFILE.DAT	Prints the file with 2 copies.
PRI/DEL MYFILE.DAT	Prints the file, removes it immediately.
TYPE MYFILE.DAT	Types the file on the terminal.
DIRECT	Displays the names of all the files on user disk area, including the size and date.
DEL MYFILE.DAT	Deletes the file from the user's area!!!
KJOB	Logs the user off the PDP-10.

Note:

A '.' in the left-most position is typed by the MONITOR. This signifies that the Computer is waiting for instructions. If the computer replies with a ?......? then the command was not understood. Try again? NO! Check that the command is a valid one.

For detailed information about all Monitor Commands refer to Manual MNT-2.

2.5 Cost Limits

If you receive the message ?COST LIMIT EXCEEDED, then respond with **SET COST +10.00** to gain a further $10.00, then type **CONTINUE** to carry on with no loss of processing.

2.6 Printing

In most cases printing is performed on the high speed line printer at the Hawken Batch Station. To find out if a queued request has been serviced, use the **Q/L[PPN]** command.

Special stationery usually incurs additional expense to the user. For pricing details refer to MNT-1.

2.7 Large Files

If a VG file is too large for the normal user disk quota there are two possible options:

1. Use the ARCHIVE system for storage of files, which is low cost but expect up to two hour waits for file retrievals.
2. Hire or purchase a private disk pack. for full details on disk pack hire, refer to MNT-1.

CHAPTER 3

TERMINALS

The following is a summary of popular terminal commands:

1. *Carriage Return* ® : the RETURN key indicates to the monitor or other program that you have reached the end of the command or line. The program responds to the command or accepts the line.
2. *Escape* **<ESC>** : the ESC (or ALT) key is echoed by the monitor as $.
3. *Control* © : this symbol indicates that the following character is typed while the CTRL (control) key is held down.
4. *Control-C* ©**C** : is used before login to get the attention of the monitor. It may be used as an emergency stop by pressing ©**C** twice to terminate a running program.
5. *Control-O* ©**O** : stops further printing of text on the terminal but the program continues to run until the end of the text. A second ©**O** will resume printing if the end of the text has not yet been reached.
6. *Control-S* ©**S** : halts printing at the terminal. For example, if you are displaying the contents of a file, a ©**S** will stop the display so you can read it.
7. *Control-Q* ©**Q** : restarts typing from the point at which it was stopped by ©**S**.
8. *Control-R* ©**R** : retypes a corrected line to show the effects of removing deleted characters.
9. *Control-U* ©**U** : directs the program to ignore the line you are currently typing.
10. *Delete* **<DEL>** : DEL or RUB key causes the character just typed to be deleted.
11. *Period* **.** : the period is displayed by the monitor to indicate that it is ready to accept commands from you. Type a command immediately following the period.
12. *Asterisk* ***** : asterisk is used by most user programs, e.g. the VG menu, to indicate that it is ready to accept one of the commands it recognizes.

    Note: You should *not* type a monitor command for a user program *nor* a user program command for the monitor.

3.1 Typing Errors

3.1.1 *Deleting incorrect characters*

If you make a typing error, you can correct it immediately at the keyboard if you have not yet already pressed the RETURN key (®). Press the key marked DEL or RUB, once for each character back to the first incorrect character, then type the correct characters and continue as if nothing had happened. If you want to make sure the correction was properly performed, use the command ©**R** before you press RETURN (®). The corrected line will then be displayed. Then simply press ® and carry on.

3.1.2 *Deleting an entire line*

If a mistake was made early in a long line, or many mistakes made, it may be easier to delete the entire line and start again. If you have not already pressed ®, you can erase the line with ©**U**. The monitor will print ↑U and you can retype the line.

These error-correcting techniques are a form of editing that may be used at any time; the only restriction is that you must perform them *before* you use the RETURN key (®).

3.2 Logging-off

When you have reached a convenient point in your work and wish to leave the terminal, you should log off the system. Logging-off serves two important functions:

1. Terminates your communication with the monitor so that someone else may use the terminal.
2. Prevents further terminal access to your files so that other users of the terminal may not damage them. Logging off also terminates monitor accounting of your computer use.

Logging off is performed by typing

K/F®

The terminal responds with

```
Job 33 User JUDY [72,140]
Logged-off TTY67 at 10:32:00 on 10-Jan-80
Cost $0.05 [Excluding spooled I/O & MOUNT charges]
Runtime: 0:00:00, KCS:2, Connect time: 0:00:31
Disk reads:39, Writes:0, Blocks saved:145
```

. *Monitor prompt*

This summary of your activity at the terminal marks the end of your terminal session. Once you have logged off, login must be performed again before you can use the system.

CHAPTER 4

CREATING NEW VG FILES

4.1 Function CREATE

4.2 Purpose

To define a new VG file with user-defined Field Names.
Note: The function NEW is identical in operation to the function CREATE in earlier versions of the VG program.

4.3 Instructions

After the command R VG the menu asks for the name of the VG master file. Enter the name of the file to be created even though it does not exist! VG does not allow a file to be created with the same name as an existing one.

Then select the function NEW. (It may be necessary to enter the new name with the SWAP function if another VG file has just been used). Starting from Field One and using up to 20 characters, give each field a suitable name. (Refer to the last chapter for examples of field names.) There is a limit of 77 fields.

4.3.1 *Key*

The key is an important field in that it is used to identify each record. If a surname is used for the key then it is likely several records may end up with the same key. To assist in separating the records, append the initials to the key.

Some other types of file have a natural key such as a part number or catalogue number. These should be used. In some cases arbitrary numbers may have to be assigned. A special switch in the Edit program can be used to force *unique* keys in Update mode. (See Chapter 5.)

4.3.2 *Now What?*

You should now proceed to put information into the new file with the UPDATE function. Select the Update Function, enter Update mode, type ® to the Constant Field question this time and proceed to enter the data.

Example

```
.R VG®

        ***VG ***

Name of Master file : Test1®                  Name of new file
%Master file TEST1 not present
*NEW®                                         To prepare VG for a new file

Creating new VG file :

Descriptor for field 1 : Surname®             The key
Descriptor for field 2 : Full Name®
Descriptor for field 3 : Address®
Descriptor for field 4 : Postcode®
Descriptor for field 5 : ©Z®                  Exit
```

Now the new file ‘TEST1’ has been initialised. Refer to Chapter 5 (Updating) for instructions on how to add information.

CHAPTER 5

UPDATING

5.1 Purpose

The Update function is for adding records to a VG file.

5.2 Instructions

Select the Update function from the Menu. You may then be asked if Unique Key Checking is required, then if constant fields are to be used, and finally you add the required records field by field.

The following explains Unique Keys, Constant fields, prompts and multi-line input.

5.2.1 *Unique Keys*

To prevent the duplication of Keys within a file an option is available to detect non-unique keys. If the option is required reply **YES** to the Question: **Unique Key Checking Required** : Then a file will be created on your area for this purpose. It will have the name 'XXXXX.BTR' where XXXXX is the name of your master file. Do not delete this file unless you wish to discontinue the unique key checks. The existence of the file causes a default of checking, so the question will not be asked repeatedly.

Note: Only the first 10 characters of a key are used for unique key checking.

5.2.2 *Constant Fields*

Certain fields may be defined to remain constant for the duration of an update run. For example a group of persons may all have the same postcode, so this field could be set to be a constant.

If this option is required answer **YES** to the question and proceed to enter the data into the fields as the respective names are displayed. Fields set to constant will not be displayed. Blank constant fields are entered with a single blank. To reset the constant fields type ©**Z** to the first field of the next record, which returns to the menu. Then start the 'update' again.

5.2.3 *Prompts*

Prompts are provided to make updating easier. The first line of a field is preceded by the Number of the field and the Name of the field enclosed in square brackets,
e.g. [2: **Fullname**].

The next prompt is a ':' (colon). The ':' denotes input is required in the form of data. By typing ©**Z**, a record may be aborted at any point. A ©**Z** to the first field will return to the constant field question.

5.2.4 *Multi-Line Fields*

The above example showed the use of a multi-line address field. When entering multi-line fields use an escape (echoed on the screen as the symbol $) to end the first line, then terminate the last line with *two* carriage returns.

Example

```
.R VG®

           *** VG ***

Name of Master File : Test®
Master file is TEST.
*UPDATE®                                      Select the Update function

Is Unique Key Checking required (Y or N) : N®
Any constant fields (NO,YES) ? NO®
Start first record for update.
[1:Surname]
: Smith®
[2:Fullname]
: Mr R. Smith®
[3:Address]
: 22 Main Str$                                Escape key
: Brisbane®
:®                                            Another®
[4:Phone]
: 123-5522®
The record is automatically stored at this point
Finished with SMITH, start another.
[1:Surname]
: ©Z                                          ©Z to exit from UPDATE

           *** VG ***

Master file is TEST.
*©Z                                           ©Z to exit from VG
```

CHAPTER 6

EDITING

6.1 Purpose

The Edit function has the purposes of deleting and altering records in a VG file.

6.2 Instructions

Select the Edit function from the menu. You may then be asked if *unique* key checking is required. (This prevents keys being changed to that of one that already exists!)

Now the Edit will ask for a Key. The following is an outline and guide on how to understand prompts, use multi-line fields and location of the required Keys.

6.2.1 *Commands*

There is a full list of all EDIT commands at the end of this chapter.

6.2.2 *Prompts*

Prompts are provided to make editing easier. The first line of a field is preceded by the Number of the field and the Name of the field enclosed in square brackets, eg **[2: Fullname]**. Underneath this is the data contained in the field. The final prompt is either a ':' (colon) or an '*' (asterisk).

The ':' denotes input is required in the form of data, as a result of the command **INPUT**. This data is added as a new line in the field (i.e., it goes underneath).

The '*' denotes that an Edit command is required.

The commands **Q** and ©**Z** quit the current operation, with no effect on the original record. The command **W** causes the current record to be written to disk.

These commands only operate when an '*' is displayed as the prompt. By typing ©**Z**, a record may be aborted at any field.

6.2.3 *Multi-Line Fields*

When entering multi-line fields use an *escape* (echoed on the screen as the character $) *to end the first line*, then terminate the last line with two carriage returns.

6.2.4 *Locating the Required Record*

When asked for the KEY of a record to be altered enter the key of the record. If there are several records with the same key then *skip* through the file by using the wild type Key.

e.g. Key : **SMITH*** will locate all the SMITHS.

If the first SMITH is not the correct one then type **ESC** (echoed on the terminal as the character $) to skip to the next SMITH. If not sure of the spelling,

e.g. SMITH or SMYTHE,

then try Key : **SM?TH***®, where the **?** and ***** are wild.

6.2.5 *Wild Keys*

When using *wild* keys, the Index should be sorted on Key for easier searching. Also a wild key remains set until a new key is entered. When a search reaches the end it returns to the start again.

When the required record has been located, type ® to the first field prompt. This causes the record to be readied for editing. A ©**Z** will skip to the next record.

Example

```
.R VG®
             * * *  VG  * * *
Master file is TEST                                VG file
*EDIT®                                             Select EDIT function
Is unique key checking required (Y or N) : N®      No unique key checking
Key? : banks®                                      Key of the record to be changed
→ BANKS? : ®                                       Yes, alter this record
[1:Name]
BANKS
*®                                                 Look at next field
[2:Age]
21
*®                                                 Look at next field
[3:Address]
Woolloongabba                                      Change this field
*c /Woolloongabba/Salisbury/®
Salisbury                                          Now Salisbury
*W®                                                Write record back
Key? : ©Z                                          No further changes
```

Now the record has been written back to disk, and the function EDIT *is asking for another record. Type* ©**Z** *to return to the Menu.*

6.2.6 *Edit Command Set*

The following description of the Edit command set uses the follow abbreviations:

F/L Means Field and Line level.
(There may be several lines to a field).

REC Record Level, i.e. after a record has been retrieved and the command **EDIT** is required for changes.

Command	*Abbreviation*	*Argument*	*Mode/Description*
ALLCHANGE	**A**	[n]/str1/str2	F/L Change all occurrences of string 1 into string 2. The delimiter '/' may be any non-blank character.
APPEND	**AP**	[n]/string	F/L Append the string to the field/line.
BOTTOM	**B**	none	F/L Move to the last field/line.
CHANGE	**C**	[n]/str1/str2	F/L Change the first occurrence of string 1 into string 2. If string 1 is omitted, string 2 is prepended. If string 2 is omitted, string 1 is deleted.
DELETE	**DEL**	[n]	F/L Delete the field/line. You are then at the next field/line.
F????			Force field mode in command ????. ???? may be any command whose mode is described as F/L. For example, if in a multiline field, DEL will delete the current line but FDEL will delete all the lines in the field.
GET	**G**	file spec	F/L Insert text from the specified file. You are then at the last line inserted.
HELP	**H**	none	Type this text.
INPUT	**I**	[n]/string	F/L See INSERT.
INSERT	**I**	[n]/string	F/L Insert the string after (or before if the repeater is negative) the current line. If the string is omitted, then Input mode is entered in which all lines typed are inserted until a blank line is received (i.e. ®).
KILL	**K**	none	REC Kill (delete) the current record from the file. This gets out of edit.
LOCATE	**L**	n	F/L Locate field/line n.
NEXT	**N**	[n]	F/L Move down to next field/line.
PRINT	**P**	[n]	F/L Print current field/line. You are left at the last line printed.
PUT	**PU**	[n]filespec	F/L Write out the current field/line to the file specified. You are left at the last line put out.
PUTDELETE	**PUTD**	[n]filespec	F/L Same as PUT but then deletes whatever was put out.
QUIT	**Q**	none	REC Quit editing this record without changing it in any way.
TOP	**T**	none	F/L Move to first field/line.
UP	**U**	[n]	F/L Move up to previous field.

UPDATE	**UPD**	none	F Enter update mode. This provides automatic insertion into empty fields. You may insert more than 1 line in a field by ending the first line with **ESC**. You leave this mode after doing the last field or by typing just **ESC** to an empty field.
WRITE	**W**	none	REC Write the modified record to the file.
*	*	none	REC Print out all fields in the record. You are left at the last line of the last field.
©**Z**			Abort.
®			Same as NEXT.
ESC			Same as NEXT.

[n] is an optional repeat count; it causes the command to be executed n times. If n is negative, we move up on each repeat instead of down. For example, **P 2** prints this line and the next while **P -2** prints this line then the previous line.

CHAPTER 7

FILE ENQUIRIES

7.1 Function REPORT

7.2 Purpose

To retrieve records to a terminal or line printer file. An example may be in searching a file for a selected key word in a bibliography. Another example may be a motor vehicle file where the number plates are used as the Keys, who owns the car?.

7.2.1 *Instructions*

Select the REPORT function from the Menu. If you have many fields and will require a list of their names, it is possible to display the table of field names by answering **YES** to the first question. If working from a Video terminal then write down or remember the fields required.

Then a choice for the output to go to a printable disk file or direct to terminal must be made. Disk file is only really needed for special "dump" type listings. For terminal output Respond to this question with T® or ®.

If field names are required to precede each Field typeout, then answer **YES** to the next question. By using this option the typeout is easily identified.

Where the field that is typed out is wider than the available space the additional part is placed underneath the first line, not under the Field Name. If the field is a multi-line field then the additional lines are also placed underneath.

Next respond to the question Fields to be displayed : with either ® or the numbers of the fields to be displayed. Separate the numbers with a comma. This allows examination of only those fields which are of interest. Useful with records that contain many fields (i.e. over 20).

7.3 Response to Key :

The last question is Key :. Respond to this question with one of the following:

7.3.1 *Wild keys*

This allows the selection of groups of keys with something in common.

e.g. **SMITH*** will select only those keys that start with SMITH, and **370*** will select all keys the commence with 370, and so on.

To select all keys enter '*'.

If an individual record is required then enter the key of that record.

7.3.2 *Search and Select*

To invoke the Search and Select option respond to the question Key : with +. Refer to chapter 'Search and Select' for detailed instructions. This option allows the selection of records with specific values in nominated fields, e.g. select only those students enrolled in GE100 or GE200 and so on.

7.3.3 *Using a Set*

If a SET file is to be used (refer Chapter SETS), then type **@FILNAM** to the question Key : (where FILNAM is the name of a set file). Then the set file will be used as required.

7.3.4 *End of enquiry*

When no more records are to be examined then type ©**Z** to return to the Menu.

Example

```
.R VG®
          *** VG ***

Master file is TEST
*REPORT®                                     From menu
Interactive enquiries
Display Table of Names : Y®
[ 1]      Name
[ 2]      Age
[ 3]      Address
[ 4]      Phone
[ 5]      Marital (s,m,d,w)
[ 6]      Children
[ 7]      Employer
[ 8]      Position
[ 9]      Salary
[10]     Motor Vehicle

Output to "File" or "Tty" (F or T) : T®
Field names required ? : N®                  No

Fields to be displayed : [<cr> for all]

Key : Rex®                                   Select 'Rex'

REX
31
Kensington
6676767
m
none
Dept.Agriculture
Technician
$432.00 pw
1964 Holden

Key : ©Z                                     No more
```

And again

```
Fields to be displayed [<cr> for all] : 5®   Only field 5

Key : Rex®                                   Select 'Rex'

Marital (s,m,d,w) : m                        5 only

Key : ©Z                                     End
```

7.3.5 *Disk File Output*

For files created on disk, a page control option is available. Where records are less than one page in size, page control prevents splitting of records over pages. The automatic print option is also invoked when a disk file is created. The following is an example of a disk file dump.

Example of a disk dump

```
Number   :  EEE-531
Type     :  Holden
Color    :  RED
Owner    :  Jones r
Dept     :  History
Phone    :  5555

Number   :  EEK-000
Type     :  Ford
Color    :  Red
Owner    :  Carter
Dept     :  Geography
Phone    :  4410

Number   :  ETR-777
Type     :  Ford
Color    :  Red
Owner    :  MacFizz L
Dept     :  Geography
Phone    :  4410

Number   :  NNN-876
Type     :  Holden
Color    :  Blue
Owner    :  Power J
Dept     :  History
Phone    :  3535

Number   :  TTT-008
Type     :  Holden
Color    :  green
Owner    :  Smythe
Dept     :  History
Phone    :  5511

Number   :  VAG-000
Type     :  Holden
Color    :  cream
Owner    :  Vandenberg
Dept     :  History
Phone    :  3943
```

CHAPTER 8

SORTING

8.1 Functions SORT and SETSRT

8.2 Purpose

To order a VG file (with SORT) or selected set (with SETSRT) on the values contained in the record fields. For example, it may be desirable to sort a VG file (or set) into the order of Student Number or Postcode. It is possible to sort orders within orders, e.g., Postcode by Surname.

8.3 Method

If the required order is on 'Key' only, enter **1** to the question **Enter fields :**. If the required order is on more than one field, enter the field numbers separated with a ','. Four fields are allowed in any sequence.

If descending order is required then use the negative of the field number, e.g. to sort on field 4 descending, enter **-4**.

The left most value is the *dominant order*, with the *sub-orders* from left to right. This allows a file to be sorted, for example, AGE *by* HEIGHT, *by* WEIGHT.

Note: The sort function only looks at the first 10 characters in each field for the ordering. Also, by sorting on Field Number 1 only (the key) a fast *keysort* option is invoked.

Sorting by SORT does *not* have any effect on the order of *sets*; the order of a *set* is the order the file was in when the SET was created. SETS *cannot* be sorted by SORT, only by SETSRT.

To use the function SETSRT, first call a defined *set* into play (see Chapter 12), then select the SETSRT function and proceed as if using the function SET.

Example

```
.R VG®

            *** VG ***

Master file is CARS

*SORT®

SORT FUNCTION

Default is Ascending order, for descending order enter the field number as negative;
e.g., "-1,2 for descending order on field 1 and ascending on field 2".

Fields for file order.
Left most field most significant.
Separate field numbers with a ",".
Enter fields : 1,-2,3®                    Enter required fields
File in order 1/a 2/d 3/a
```

CHAPTER 9

SWITCHING VG FILES

9.1 Purpose

To change from using one VG file to another. This function is necessary because when the Menu is invoked for the first time after Login, it is required to enter the name of the particular VG file. This file name remains *sticky* until the SWAP function is used.

9.2 Instructions

Select the SWAP function and then enter the name of the required VG file. This file is now ready for processing.

Example

```
.R VG®

           * * *  VG  * * *

Master file is TEST
*SWAP®                                   Swap function
Changing master file from TEST

           * * *  VG  * * *

Enter master file name — CARS®           New master

Master file is CARS
*                                        Ready for next function
```

CHAPTER 10

ADHESIVE LABELS

10.1 Function LABELS

10.2 Purpose

Produce a set of adhesive labels, that are 45 characters wide by 5 lines deep. Labels may be useful for identifying items such as books, specimen jars, stock shelves, mail addressing and so on.

10.3 Instructions

Enter the required field numbers in the appropriate place when asked. Be aware that (a) *multiple-line fields*, e.g. an address field, will occupy more than one line on the label and (b) *a set of 5 header alignment labels* are produced at the beginning of each run.

Each label file produced will have the name XXXXX?.DAT, where XXXXX is the master file name and the '?' is in the range from 'A' to 'Z', giving each file produced a unique name.

Note: Sorting must be done *prior* to the production of a set of labels The labels will come out in the order of the VG file. Refer Sorting for details.

10.3 Constant Line

By responding with a $ (Escape) to the field number for a particular line on a label, it is possible to have a constant line appear on every label, e.g. FINAL NOTICE, and so on.

In the above example, the words 'FINAL NOTICE' appear on every label.

At the end of the Label run, the automatic print routine is invoked to print the labels.

Note on Mailing lists

Where labels are being produced for mailing purposes, be aware that the postal authorities prefer bulk mail to be arranged in postcode order. Therefore a field for postcode in the file definition should be included (separate from the address, if necessary), for sorting purposes. Then sort the file on primary order postcode by surname before using the Label function.

10.5 Costs

A guide to label-production costs in a file with about 20 fields with no selection used, is about $2.00 per 1000 labels at priority 10 normal shift rate. The cost of the stationery is contained in manual MNT-1.

Example

```
Master file is TEST.
*LABEL®

Labels function

Fields may be selected in any order. If you wish to insert a special message line on a label,
then type "$" for the field number, and you will be asked to enter a "literal" string of up
to 45 characters for insertion in the label. If you wish to not use all fields on a label type
<cr> and that field will be left "blank".

Output file is : TESTB.DAT®            Output file
Field for line 1 on label : 1®         First label line
Line 1 >> "name"                       Confirmation only
Field for line 2 on label : 7®         Second label line
Line 2 >> "employer"
Field for line 3 on label : $          ESCAPE key for constant field
Enter special line 3 here
FINAL NOTICE®                          Will appear on every label
Field for line 4 on label : ®          Empty line

Key ? : *®                             Indicates all records
Produced 29 labels                     Made 29 labels
Key : ©Z                               Return to Menu
```

Note: *The auto-print feature is invoked at the end of the run.*

Example of Output

```
Mr R. Jones
Brisbane City Council
FINAL NOTICE

Mr W. Smith
University of Queensland
FINAL NOTICE

Mr A. Bert
22 Main St.,
Brisbane
FINAL NOTICE
```

CHAPTER 11

SEARCHING AND SELECTING

11.1 Purpose

The purpose of the function is to allow the selection of records by the presence of selected values in nominated fields. Up to *four fields* may be searched. Up to *ten values* may be searched for in any one field. By selecting the same field several times, it is possible search for more than ten values.

11.2 Instructions

The VG multiple search option is invoked by typing a + (plus sign) to the question **Key** : in VG functions. Then you are asked for the Field Numbers to be searched (up to 4). By typing a ©**Z** to this question the option is dropped. The next question to answer is for the dependency between the fields.

Then you must decide on the dependency between the values for the first field and then enter the values to be searched for. This is repeated until all fields entered are complete.

11.2.1 *Search values*

Each value nominated to be searched for, may be up to ten characters long, and is made up of any alpha-numeric combination. Hyphenated words are acceptable. In addition you may specify a search value as a *wild* type.

11.2.2 *Wild cards*

A value ending with * will match the preceding characters, with following characters being ignored to make a match, e.g. **EX*** will match with EXPERT, EXCITE, EXCLAIM, EX..... etc. By replacing a nominated character with **?** that single character becomes *wild.* It is important to note that if there is no character in the the destination at all, then no match will be made, e.g, **WORD?** will not match with WORD, but **WORD*** will match.

11.2.3 *Dependency between values*

Each value that is searched for in a field may be made dependent in the following manner. In a nominated field the values may be made **EVERY**. This means that *every* value nominated in that field *must* be found, e.g. search for RED *and* BLUE. RED and BLUE must exist in the field for a satisfactory search.

The values may be made **ANY**. This means we will accept *any* of the nominated values as satisfactory, e.g. search for WHITE or GREEN or PINK, where *any one* will satisfy.

11.2.4 *Dependency between fields*

If searching two or more fields then the dependency between each *field* may be declared as **EVERY** or **ANY**. This allows you to select records which satisfy specific criteria in two or more fields.

11.3 Key searching

This operation is different from the above. Key Searching is performed only on the *key* and *only the first ten characters are examined.*

Wildcard keys are used to retrieve the required records. If the key was SURNAME then a key like **SMITH*** will return all SMITHS. If the key was a Library of Congress Catalogue Number then a key like **LA2102*** would return all records with that prefix (which happens to be Education Australia).

11.4 Search example

In the following example a VG file named CARS will be searched to find all those cars that are 'Ford' or 'Morris' but the owner *must* belong to department 'Geography'. This is a *conditional* search.

11.4.1 *Comment*

The above search can be used in all report and tabulation functions simply by entering + to the question Key :

Example

```
Master file is CARS.
*REPORT®                                          Use REPORT function

Interactive Enquiries

Display Table of Names ? : YES®

[ 1]    Number
[ 2]    Type
[ 3]    Color
[ 4]    Owner
[ 5]    Dept
[ 6]    Phone

Output to File or Tty (F or T) : T®
Add field names [Y/N] : YES®
Fields to be displayed [<cr> for all] : ®         All
Key : +                                           Activate the search
Multiple Field Search Option
Type <cr> to ignore
Type ↑Z to any Search question to restart.

Field Numbers to be searched [Limit 4] : 2,5®

     2 →     Type
     5 →     Dept
Enter Relationship between "Fields"
If ANY field will satisfy,              REPLY "A"
If EVERY field must be satisfied,       REPLY "E"
RELATIONSHIP IS : E®

* * EVERY "FIELD" must be satisfied * *

Relationship of values in : "TYPE"

Enter Relationship between "VALUES"
If ANY "VALUE" will satisfy, REPLY "A"
If EVERY "VALUE" must satisfy, REPLY "E"
RELATIONSHIP IS : A®
Type: ANY "VALUE" will satisfy

Now enter the Values [Limit 10] required for Type

Value [ 1] for : Type           : FORD®
Value [ 2] for : Type           : MORRIS®
Value [ 3] for : Type           : ®               To end list

Relationship of values in : "DEPT"

Enter Relationship between "VALUES"
If ANY "VALUE" will satisfy, REPLY "A"
If EVERY "VALUE" must be satisfied, REPLY "E"
RELATIONSHIP IS : A®
Dept : ANY "VALUE" will satisfy

Now enter the Values [Limit 10] required for Dept

Value [ 1] for : Dept           : GEOGRAPHY®
Value [ 2] for : Dept           : ®               To end list
```

Key: *® *All records to be included*

Key: ©Z® *End report*

Example of Output

```
Number    : ETR-777
Type      : Ford
Color     : Red
Owner     : MacFizz L
Dept      : Geography
Phone     : 4410

Number    : EEK-000
Type      : Ford
Color     : Red
Owner     : Carter
Dept      : Geography
Phone     : 4410
```

CHAPTER 12

SETS AND GROUPS

12.1 Function SETS

12.2 Purpose

Specific *sets* of records may be set up which eliminates the need to use the Search and Select feature every time the file is processed. The Set that is created is *valid only until the file is Edited or Updated.*

12.3 Instructions

Select the function SETS, which then displays the field names and positions. Then enter a name for the set (5 characters). The Search and Select routine is then invoked where you set the criteria for the set creation. For full details on the Search and Select refer to that chapter.

12.3.1 *Using a Set*

To use the created set, type **@SETNM**® to the question **Key** : , in the required function. The set is then locked on.

There is no limit to the number of SETS.

The Sets Function is particularly efficient when performing more than one function on the set, e.g. to produce a set of labels (LABELS), a column-report (COLREP) and perhaps some form-letters (LETTER).

12.4 Set-creation example

In the following example a set will be created which will be used in the REPORT function.

Remember: a set is *valid only until the Master file is modified by* EDIT *in any way!*

Example of Set Creation

```
Master file is CARS.
*SET®                                              Use SET function

Sets Function

This set is only VALID UNTIL the Master file is Edited or Updated, then the set must be
recreated. Failure to do so will result in errors.

Set files are automatically given the extension ".VST".

Name of file for the SET? [5 Chars] : FORDS®

[ 1]      Number
[ 2]      Type
[ 3]      Color
[ 4]      Owner
[ 5]      Dept
[ 6]      Phone

Multiple Field Search Option
Type <cr> to ignore
Type ↑Z to any Search question to restart.

Field Numbers to be searched [Limit 4] : 2®

      2 →         Type

Relationship of values in : "TYPE"

Enter Relationship between "VALUES"
If ANY "VALUE" will satisfy, REPLY "A"
If EVERY "Value must satisfy, REPLY "E"
RELATIONSHIP IS : ®                                ® to default to ANY
Type : ANY "VALUE" will satisfy

Now enter the Values [Limit 10] required for Type

Value [ 1] for Type : FORD®
Value [ 2] for Type : ®                            ® for end list

Key : *®                                           All keys
Processed 2 records                                i.e. 2 records with Ford
```

Example of Using a Set

```
      *** VG ***

Master file is CARS
*REPORT®

Interactive enquiries

Display Table of names ? : NO®
Output to file or tty ? : T®
Add field names ? : YES®
Fields to be displayed : 2,3®             Only fields 2 and 3

Key : @FORDS®                              Use set FORDS
FORDS Set file is now in use

Type        : Ford
Color       : Red

Type        : Ford
Color       : Red

Key : ©Z®                                  End

Processed 2 records
```

CHAPTER 13

COLUMN REPORTS

13.1 Function COLREP

13.2 Purpose

To allow a self-designed report layout with fields from individual records placed into selected columns on standard line printer stationery. Optional sub-headings and an index are available, plus a 'design' file may be stored for later repetition of the same report.

13.3 Instructions

Select the function COLREP from the menu. If this is the first creation of this particular report the answer to the first question about the existence of a design file is **NO**. The next question asks if one should be created. The answer is **YES** or **NO**. If the answer is **YES** then give the design file a name of 5 characters, plus a name of 5 characters to the listing file.

Then a *Top Of Page Heading* of up to 75 characters is typed in, which appears at the top of every page.

The columns are set up with a separator of 1 to 5 spaces in width. Optionally a symbol may be included in the separator. Two spaces are recommended with an '*' as the symbol.

Then working from the left margin, nominate the *field number* for placement into each column, and the width in characters for each column. After each declaration, information on the remaining width left is typed out, and the program asks if there is to be another column declared. For no more columns type **NO**.

13.3.1 *Sub-headings*

The next question is if a subheading is required. If so enter the number for the field to be used. If no sub-heading is required then type a carriage return. If a sub-heading field is selected an *Index* is automatically created.

Note: The file should be sorted on the field that is used for the sub-heading, causing all sub-headings to be grouped together. Refer to SORT for sorting instructions.

The next question is KEY. Type in * for all records to be processed or individual keys as required. Wild card type keys are acceptable, eg, **SMITH*** will select all keys SMITH????? etc.

By responding to the key question with the extended Search and Select option is invoked. Refer to SEARCHING.

13.4 Layouts

13.4.1 *Headings*

The top of form heading contains a user-supplied string of 75 characters plus a page number and date of report creation.

If the sub-heading option is invoked, then a sub-heading is printed at the start of each occurrence of a new value for the field selected for the sub-heading. This is why it is important to sort the file into primary order on the sub-heading field, otherwise the subheadings will be scattered and repeated. Each sub-heading is preceded on the printout with the Field Name and has a row of asterisks above and below. By using the switch **/P** to the Sub-heading question, a new page will be thrown on the occurrence of a new Field Value for the subheading field. In the example shown later in this chapter the sub-heading field is 'Type' (Motor Vehicle Make), the file was sorted with 'Type' as the major order, clustering all Fords, Holden etc together. In the report produced, all the makes appear under their respective 'Type'.

13.4.2 *Column headings*

Each column has a header taken from the field name. If the column is only 5 characters wide, then only the first 5 characters of the field name will be used for the column header.

13.4.3 *Separators*

The column separators may be made from 0 to 5 characters wide. In addition a symbol such as '*' may be inserted in the separator. The separating symbol is left justified.

13.4.4 *Columns*

The minimum width is 5 characters and the maximum is 132. The maximum number of columns is 26 columns of 5 character width with no separators. Columns should be wide enough for the text to be intelligible. Address and name fields should be about 12 characters wide with 'descriptive' text type fields about 20 characters wide. The aim is to balance the columns so that the report is not lopsided (i.e. one or two of the columns are not grossly longer than the rest).

13.4.5 *Splitting words*

Where the last word in a column is too long to fit, the word is carried over for the next line; where this is impossible (column too narrow), then the word is split. Where multi-line fields are used a new line is commenced in the column for each new line of the field (e.g., in multi-line address fields).

13.4.6 *Design files*

Design files are for reports that are to be repeated frequently. They save time in re-formatting the report layout. The design file is stored as a file with the extension '.DSN'. They must *not* be interferred with using LINED ot TECO etc. There is no limit to the number of design files.

13.4.7 *Key question*

The options to the question **Key** : are * (all records to be processed), + (invoke the Search and Select option), or **@SETFIL** (use a Set file—refer SETS).

Example of COLREP

```
Master file is TEST
*COLREP®

Column reports

Permanent file for Report design ? : N®
Create Design File ? : Y®
Name for Design File : TESTD®
Enter Main Heading (75 characters)
: ****  TESTD  ****
Name for Output file : TESTD®
Output file is : 'TESTD.DAT''
Width (1 to 5 chars.) of Separator Col ? : 3®
Enter Separator Symbol if required : *®          Type ® for no symbol
Parameters for column no. 1 position no. 1 Remains = 132
Infield        : 8®
Chtrs          : 20®
More Columns ?    : Y®

Parameters for column no. 2 position no. 23 Remains = 110
Infield        : 7®
Chtrs          : 20®
More Columns ?    : Y®

Parameters for column no. 3 position no. 45 Remains = 88
Infield        : 9®
Chtrs          : 20®
More Columns ?    : N®

Field Number for sub-heading : ®

Key : *
Running

                29 records now on output file

Key : ©Z
```

Example of Output

```
****  TESTD  ****
(Date   :    1-Nov-79)

Position      Employer                 Salary
***************************************************************************************
Storeman    *  Fine and Co.        *    $150.00 pw
Technician  *  Dept. Agriculture   *    $432.00 pw
Fireman     *  Fire Department     *    $140.00 pw
Postmen     *  Telecom             *    $260.00 pw
```

Example of Sub-headings

```
master file is CARS
*COLREP®

Column Reports

Permanent file for report design ? : NO®
Create Design file : YES®
Enter main heading (75 characters)
: ****  Car report by Type  ****
Name for output file : CARS®
Output file is CARS.DAT                          Confirmation only
Width (1 to 5 chars.) of Separator Col ? : 2®
Enter Separator Symbol if required : *®          Use *
Parameters for column no. 1 position no. 1 Remains = 132
Infield      : 1®
Chtrs        : 10®
More Columns : YES®
Parameters for column no. 2 position no. 13 Remains = 120
Infield      : 4®
Chtrs        : 12®
More columns ?  : YES®
Parameters for column no. 3 position no. 27 Remains = 106
Infield      : 6®
Chtrs        : 12®
More Columns : NO®
Field no. for Sub-Heading [type <cr> to ignore] : 2
Key : *                                          All keys

Running

           4 records now on output file

Key : ©Z
Print Output file [Y/N] : YES®
Priority : ®                                     ® for default of 4
Number of copies : ®                             ® for default of 1
Remove file from your area ? : YES®
CARS.DAT/PRIO:4/LIM:21/FOR:NORMAL/COP:1/DELETE
[LPT01:CARS=/Seq:185774/Limit:21, 1 file]

       *** VG ***

*©Z                                                  Exit
```

Example of Output

```
* * * *  Car report by Type  * * * *
(Date    :      23-Jan-80       )          (Page :    1)
Type                                           FORD

Number          Owner                    Phone
*************************************************************************************

ETR-777       * MacFizz L              * 4410
EEK-000       * Carter                 * 4410
*********** **
Type                                           HOLDEN
*********** **
VAG-000       * Vandenberg             * 3943
TTT-008       * Smythe                 * 5511
NNN-876       * Power J                * 3535
EEE-531       * Jones R                * 5555


          * * * * *  CARS INDEX  * * * * *
          Type                                        Page

            FORD                                 . . . . .  1
            HOLDEN                               . . . . .  1
```

CHAPTER 14

TABULATIONS

14.1 Function TABUL

14.2 Purpose

TABUL is a program which processes VG files, producing
(a) frequencies of occurrence of variables in one field, or
(b) a one way tabulation of variables from one field against the variables of another field, or
(c) a two way tabulation of the variables in any two fields.

14.2.1 *Variables*

A variable within a field is considered to be any unbroken string of alphanumeric characters. The first ten characters are taken as a variable and the rest of the string is truncated if required. The program will handle up to 100 variables in any one field. Hyphenated words are treated as one variable. All other punctuation symbols are regarded as being delimiters between variables (comma, brackets, period, etc). No distinction is made between upper and lower case, all being converted to upper case. The output is in the form of a file ?????.MEM, which is *not* automatically queued for printing.

14.2.2 *Instructions*

Select the function TABUL from the Menu. Then answer the question One Way or Two way Tabulation required. The next question is what field numbers are to be tabulated. Enter these as two numbers separated with a comma. If only one field is entered only the frequency of occurrence of each variable (string) is reported.

Note: In the event of a blank field being examined, * * * * * replaces the blank.

14.3 Uses

It may be convenient to obtain the frequency of occurrence of the subjects a person is enrolled in, or the frequency of these subjects broken down into areas such as postcode. Another use may be find the frequency of occurrence of keywords in a bibliographical-type file.

14.4 Warnings

Where large fields are concerned with many variables, very large results may be produced. Also single characters such as 'a' will be treated as a variable. Variables over ten characters long are truncated.

Example

```
.R VG®

        *** VG ***
*TABUL®                                     Tabulations

VG Tabulation
TWO WAY Tabulation required ? Y/N : YES®

Enter the field numbers to be cross tabulated : 2,5®

Type          [Tabulation with]     Dept

Key : *®                                    All keys
Found 6 records
Output file 'CARS.MEM' being produced

Creating CARS.MEM
RUNOFF: CARS      2 Pages
```

Example of Output

```
  Report for 'Type' : FORD [by] 'Dept'

GEOGRAPHY → 0002

  Report for 'Type' : HOLDEN [by] 'Dept'

HISTORY → 0004

                                                              Page 2

  Report for 'Dept' : GEOGRAPHY [by] 'Type'

FORD → 0002

  Report for 'Dept' : HISTORY [by] 'Type'

HOLDEN → 0004

  Total input records processed = 6
  Output records generated = 6
    END
```

CHAPTER 15

FORM LETTERS

15.1 Function LETTER

15.2 Purpose

Form letters can be used for repetitive letters, legal documents such as contracts, student memoranda, course enrolment notices and so on.

15.3 Runoff

This feature of VG uses the Typesetting Package called RUNOFF. Runoff is fully documented in manual MNT-14. Before proceeding, it is necessary to be familiar with Runoff for form letters. As well, a knowledge of a text-editor such as EDIT or SOS is required. It should be noted that EDIT (on the PDP-10 U.Q.) has a command set similar to the Edit in VG.

The principal reason for form letters is to save a typist from the labourious task of typing the same letter to *n* number of people with a small variation in each letter. The variable components in such letters are likely to be Surnames, Addresses, Subjects enrolled in, Semester Addresses, Phone Numbers and so on.

15.3.1 *Legal contracts*

In Legal Contracts there are likely to be: Special Conditions, Price, Date of agreement, Various addresses, object of the contract and so on. Once the contract wording is established, a VG file of contracts can be established which contains the varying details. Duplicate copies can be easily printed on high quality letter head stationery or whatever. To reprint the contract is a simple procedure. The same file can be used for many other purposes as well, such as the production of a register.

15.3.2 *Form letters*

Once the form letter has been laid out using RUNOFF, and *dummy fields* for the varying parts (such as name and address of recipient), as many letters as needed can be produced, as well as a name and address label for each letter with the Labels function. Also the letter can be permanently stored for later use by another group.

15.4 Instructions

Set up the Form Letter with a suitable editor with the file name having the extension .VGL. Replace the varying parts of the letter with the VG Form Letter instructions **$$n** on the next blank line following the varying part. The **n** is the field number to be inserted.

Example Input .VGL File

```
Dear Mr                                       First line of letter
$$2                                           Put surname here
.par                                          Begin new paragraph, using paragraph
                                              parameters
Re your motor vehicle:                        Constant text
$$6                                           Motor vehicle description field
     We would like you to present your vehicle
for inspection . . . . . . .                  Continue main body of letter
```

Example of Result

Dear Mr Jones

Re your motor vehicle: Holden Kingswood Sedan

We would like you to present your vehicle for inspection . . . .

CHAPTER 16

VG FILE INFORMATION

16.1 Function INFORM

16.2 Purpose

To obtain a typeout of the field names of the current VG file in use, the number of records in the file, and the degree of efficient utilisation of disk space.

16.3 Instructions

Respond to the Menu prompt with the command **INFORM**. That is all. The following example shows the type of result.

Example

```
Master file is TEST
*INFORM®

VG Master File Information

Record Description

Field       Description
  1 →       Name
  2 →       Age
  3 →       Address
  4 →       Phone
  5 →       Marital (s,m,d,w)
  6 →       Children
  7 →       Employer
  8 →       Position
  9 →       Salary
 10 →       MotorVehicle
Records on file           29

Words allocated=    506    Used=   410
Disk utilisation efficiency=  81.03%

      * * *  VG  * * *

Master file is TEST
*©Z®                                          Exit from menu
```

CHAPTER 17

REFRESHING FILES

17.1 Purpose

Where extensive editing has been performed on a VG file (i.e. where over 70% of records have been enlarged by altering fields), redundant space will be recovered with a file refresh.

Also, large processing runs will benefit speed-wise where a refresh has been done causing the Master file to be in the same physical order as the index. Where constant sorting occurs on a file, refreshing serves no purpose. The same applies to small files (i.e. less than 500 records).

Use of the INFORM function will provide the statistics necessary to decide if a file-refresh is appropriate.

17.2 Instructions

Select the REFRES function from the menu. The rest is all automatic.

17.3 Warning

Refreshing makes all *sets* redundant.

CHAPTER 18

MISCELLANEOUS FUNCTIONS

18.1 Function EXIT

Purpose of this function is to exit from the menu and return the user to monitor level. This is the same as typing ©Z to the menu.

18.2 Function FILES

Reports the names of all VG files and SET files.

18.3 Function IBUILD

In the event of the Index file being destroyed or corrupted, use the function IBUILD to re-generate this index.

18.4 Function KJOB

Logs user off the system direct from the menu.

CHAPTER 19

NEW FILES FROM OLD

19.1 Functions GENER and JOINER

19.2 Purpose GENER

To generate a new VG file, *with no records in it* but with a *structure* identical to that of an existing VG file.

Thus, if two distinct VG files are to exist simultaneously with the same fields in the same order, you need create only *one* file with the function NEW, and then *copy its structure* for the second file, using the function GENER. Where a large number of fields are involved, this is clearly very convenient. In addition, the risk of error in re-typing is removed.

Also, where a large file is to be generated by more than one person, each person can update to separate, but structurally-identical, files (created with GENER) and then these different files may be *linked* using the command JOINER, described below.

19.3 Instructions

Once the command GENER has been issued, all the user has to provide is the name of the VG file which is to be identical in structure to the existing current VG file.

[Ensure that the *current* VG file is the one from which you wish to model; otherwise, use the function SWAP to make the desired file the current file.]

19.4 Purpose JOINER

To create a new master VG file composed of all the records from a number of existing VG files.

Note:- The files do *not* need to have identical structures in order for JOINER to process the files, although clearly if the different files to be joined together into the new master file are to be meaningful, any existing fields should *correspond*, since joining is done on a field-number basis.

Below, one application involving non-identical files, but for which any *corresponding* fields *are* identical, is discussed. (See Application (3).)

19.5 Instructions

All the user is required to supply after invoking the JOINER command are (i) the name of the new master file to be generated and (ii) the name of each file to be added into this new master file (one at a time, until all are added in).

Note: Do *not* use the name of any *existing* VG file as the name for the new master.

Some applications

(1) *Simultaneous entry by more than one person:*
Operator A uses NEW to create a VG file in the form in which the final master file is to appear. Operators B, C, ... then use GENER to create other 'subfiles' of the same structure. All operators then update their own individual files until all records have been accounted for. Finally, use JOINER to generate the 'grand' master file M, incorporating each of the files updated by A, B, C, ...

Note: Do *not* attempt to join B to A—you *must* join *each* of A and B to a *new* file M.

(2) *Regular updating of a master file:*
If a substantial updating of a master file is to take place on some regular (weekly, monthly, semester, annual, etc.) basis, but the master file is to remain static between updates, then progressive updating cannot be used, and it may be too difficult for a single person to do all the updating in a short time. One solution is to make a separate file (with GENER) to which the new records may be added progressively, without disturbance to the master file, and then at the appropriate time, the existing master and the file of records to be added may be united (with JOINER) into a new master file. Some renaming of files may be desirable in this application.

(3) *Adding fields to an existing VG master file:*
It may happen that you find the need for one or more new fields in an existing VG file. The following application describes how to use JOINER to avoid having to retype all records. Use NEW to create a VG file with all the fields you desire to have in the 'revised' VG file. There is *no need* to include any records in this file! But, you should keep the fields corresponding to the old VG file in the same position as previously (i.e. same field numbers). Now use JOINER to unite the old VG file with the new (empty) file into a new master file. If necessary, the values for the 'new' fields for the 'old' records can now be inserted by normal editing techniques.

CHAPTER 20

GENERATING DATA FILES

20.1 Function MATFRM

20.2 Purpose

To generate from a VG file a corresponding record data file containing specified fields, *but of fixed length.*

While VG carries out a variety of functions, there may be tasks to be carried out on the data contained in a VG file which VG is not designed to handle. MATFRM provides a means whereby the data of the records in the VG file can be output for processing by other user programs. Since many such programs demand records whose fields are of *fixed length* (as opposed to VG which accepts variable-length records), it is necessary to be able to access the data in this form. MATFRM provides this facility.

Notes:

1. As a standard feature of this function, the output "matrix" file (i.e. the data is arranged in rows and columns) always contains *two blank columns* between the fixed-length columns of data.

   The user must be aware of this in order to describe correctly the input format for subsequent user programs.
2. The maximum length of record generated in this form (including the blank columns between columns of data) is 600 characters.

20.3 Instructions

In response to the command **MATFRM**, the system replies

```
Matrix form generator
Type H for Help:
```

If assistance is required at this stage, typing **H**® will result in the optional help-message being displayed at the terminal.

Otherwise, simply typing ® will cause the system to begin requesting the necessary information about fields to be output and desired field-lengths.

Notice that the operation is similar to that of COLREP, in that
(i) a permanent design file may exist for repeated usage, and
(ii) the user supplies the necessary fields, number of characters for each field, and whether or not all required fields have been entered.

Example

```
Master file is XXX
*MATFRM®
Matrix type file creator
Type H for Help: ®                            ® for no help-message
Permanent file for matrix design? : ®         ® is equivalent to N®
Create Matrix Design File? : ®                Reply Y® only if you wish to create a
                                              permanent design file
Name for Output MATRIX File is : AAA®
Parameters for Column No. 1 Position No. 1 Remaining 600
Infield: 1®
Characters: 3®
More Columns? : Y®
...
...
...
More Columns? : N®
Key: *®                                       * for all records, else + or @SET for
                                              selection
Running

36 records processed

Key: ©Z                                       Return to menu
```

Sample Output

```
073   Table   63.00
095   Chair   27.50
173   Desk    84.10
    ↑       ↑
  Two blank columns
```

CHAPTER 21

BIBLIOGRAPHIC MENU

21.1 Function BIBLIO

21.2 Purpose

To call up the Bibliographic Sub-menu from the main VG menu.

The bibliographic menu contains a range of additional functions, which are particularly suited to (*but not restricted to*) records of a bibliographic nature. Most of these functions are described in Chapters 22–25. For convenience, some functions are common to both the main VG menu and the bibliographic menu; these functions are described in Section 21.4 below.

21.3 Instructions

After having entered the main VG program, select the function BIBLIO. The system responds with the message

```
  VG Bibliography Menu
Master file is XXX                where XXX is master file currently in use
                                  when the Biblio command is issued
*                                 Note that * is still the prompt
```

To obtain a list of functions available within the Bibliography menu, simply type ® at this point. The system responds with a list of the functions together with a brief description of each function, and finishing with the * prompt so that the appropriate function can be selected.

21.4 Common Functions

The following functions contained within the bibliographic menu are identical in operation and usage to the functions of the same name in the main menu:

EXIT	Returns user to Monitor level
INFORM	Types the fields in the current master file, and the number of records in that file
KJOB	Logs the user off the system
SETSRT	Sorts a SET file
SWAP	Allows a different master file to replace the current master file.

22.5 Leaving the Bibliographic Menu—Function VG

In addition to the functions EXIT and KJOB described above, the user may leave the bibliographic menu by means of the command VG, in which case control is passed back to the *main VG menu.*

Example

```
. VG®
          *** VG ***

Enter Master File Name: XXX
* BIBLIO®                                     Move to Bibliographic sub-menu
* ®                                           To inspect functions available in Bibliog-
                                              raphic menu

The following functions are defined:

BATKEY        As a Batch Job, Create a Key Word Index for a VG file
CREF          Cross Reference Lister
EXIT          Exit from Biblio
EXTSRT        Extended 100 character wide sort option
FORMAT        Format Bibliographical Output
FREQ          Produce Frequency list from Keyed Index
INFORM        Types out file particulars
KEYIND        Create a Key Word Index for a VG file
KJOB          Logs user off System
QSEA          4500 limit: Search Keyed Index File for Key Words
SEARCH        17500 limit: Search Keyed Index File for Key Words
SETSRT        Set File sorter program
SWAP          Change to using a different master file
VG            Return to VG Menu

* VG®                                         Return to main VG menu

          *** VG ***

Master file is XXX
*                                             System awaits further instruction
```

CHAPTER 22

CREATING A KEYWORD INDEX

22.1 Functions KEYIND and BATKEY

22.2 Purpose

To generate a *keyword index* file for a VG file.

A keyword index file may be regarded as simply a file containing a list of all words from any fields nominated by the user (including *all* fields), e.g., if, say, one field of a VG file contained the name of the author of the book (where each record of the VG file pertains to information about a book), and a keyword index was created for this file (possibly along with other fields), then the name of the author for each record is registered in the keyword index file.

Notice that every *word* of each nominated field is recorded as a separate entity in the keyword index. Thus, in the example above, if the author had been entered in the VG file as "Arthur Brown", then the keyword index constructed will contain the entries "Arthur" and "Brown".

Such an index provides a powerful means of searching a VG file for particular individual records or groups of records satisfying conditions imposed by the user. Such applications are discussed in Chapters 23 and 24.

Both KEYIND and BATKEY perform the same task, viz. the creation of a keyword index. The difference is that
(a) KEYIND generates the keyword index immediately as an interactive operation, whereas
(b) BATKEY allows the same operation to be performed as a batch job.

22.3 Special Points to Observe

22.3.1 *Punctuation*

All punctuation symbols, *with the exception of the hyphen*, are considered as word delimiters, i.e. they are regarded as signals indicating the ends of words rather than text-characters.

Thus, referrring to the example above, if the value for author for a record had been entered as "A. Brown" (rather than "Arthur Brown"), then the words "A" and "Brown" will be entered into the keyword index file. Note that the entry will be "A", not "A.".

However, if the VG-entry had been "Arthur-Brown", then the keyword index file entry would also be "Arthur-Brown".

22.3.2 *"Whole-of-line" Option*

If desired, for particular individual fields, "whole-of-line" entries, rather than "single-word" entries may be used in the construction of keyword index files. In this case, a 21-character string is stored in the keyword index file as the "keyword".

To achieve this, enter the field number n as **-n** as the response to the prompt **Fields to be keyed:** when using the function KEYIND (or BATKEY). (See Examples pp. 51–52.)

This feature could be useful for situations in which:

(i) one field may be, say, a book-title, and subsequent searches are to be mounted on full titles
(ii) a field is multi-line, then each line can be entered as a "word"
(iii) values for date-fields have been entered in a form such as "2 Feb 79" or "2/2/79".

However, it is *not* suitable for long continuous text strings.

Note: Before deciding to use "whole-of-line" keywords, the user should consult fully Chapter 24, which describes how a search can be mounted on the basis of a combination of keywords, as well as a single keyword.

KEYIND Example

```
. R VG®
         *** VG ***
Enter Master file name: BIBMN®
*BIBLIO®                                  Move from main VG menu to bibliograph-
                                          ic sub-menu

         VG Bibliography Menu
Master file is BIBMN

* KEYIND®                                 Select KEYIND function from this menu

Keyword Index Generator
Name for Keyed Index File : TOPIC®        Enter name desired for keyword index file
                                          (max. 5 characters)
[ 1] → AUTHOR                             System automatically recalls structure
[ 2] → OTHER AUTHORS                      of VG file in use to aid selection
[ 3] → TITLE                              of appropriate fields as response
[ 4] → DATE                               to next request
[ 5] → PUBLISHER
[ 6] → SUBJECT
[ 7] → THEMES

Fields to be keyed : 7®                   Nominate required fields, separated by
                                          commas
Key : *®                                  Key all records (or + or @SET, as
                                          required)

Processed        3 records
Output           9 Keywords               *See Note below

Commencing Sort phase                     System pauses at this stage
Completed Sort phase

Keyed index file "TOPIC" Completed
       7 unique Keywords on file          *See Note below

         VG Bibliography Menu             Automatic return to Bibliographic Menu

Master file is BIBMN
*                                         Ready for next Bibliographic menu com-
                                          mand
```

Note: The system reports 9 keywords output, but *only 7* ***unique*** *keywords. This simply means that two keywords each occur twice.*

BATKEY Example

```
. R VG®

          * * *  VG  * * *

Enter Master file name : BIBMN®
* BIBLIO®

          VG Bibliography Menu

Master file is BIBMN
* BATKEY®

BATCH OPTION Key Word Index Generator
Name for batch control file [5 characters] : INDEX®
Batch file : "INDEX.CTL"

Is the master file on another PPN?
If it is, enter that PPN [<cr> to ignore] : ®      Or enter the appropriate ppn
Do we need to Mount a Disc?
If so, what is it? [<cr> to ignore] : ®
Name to be given to Keyed Index File : KEYS®       Choose a name

[ 1] → AUTHOR                                       Reminder given as for KEYIND
[ 2] → OTHER AUTHORS
[ 3] → TITLE
[ 4] → DATE
[ 5] → PUBLISHER
[ 6] → SUBJECT
[ 7] → THEMES

Fields to be Keyed : 1,6,7®                         Keywords from fields 1 (AUTHOR), 6
                                                    (SUBJECT) and 7 (THEMES)
Key : *®                                            All records

Disposition of Keyed Index after completion         Allows for possible archiving of Keyed
                                                    Index file
Respond "A" for archive                             Important for large files
<cr> for do nothing
Remember this Batch job may take you over your disk quota
Response : ®
Keyed index files will be KEYS.VKF, KEYS.VHD, KEYS.VID     *See Note below
Delete the control file on successful completion? [Y or N] : Y®
Name of JOB for the Queue                           System requests normal information
For Group Account etc,                              for batch job to run
or Exercise Name
Please ignore if not applicable
Queue Job Name: ????®                               Enter name or ® as appropriate
[????]
Default priority = 2
Priority : 4®                                       Enter priority number or ® for default
                                                    priority
Time limit [HHMM]: ®                                Or enter appropriate value
Cost limit: ®                                       As above
[INP01:????=/Seq:500/Time:0:05:00/Core:120P]
INDEX.CTL submitted

          VG Bibliography Menu                      Returns to Bibliographic menu
Master file is BIBMN
*                                                   Ready for next bibliographic menu com-
                                                    mand
```

Note: For operational purposes, the user need not be aware that these three files are generated when creating a Keyword Index File. In future work with this keyword index file, whenever the name is requested, the user need only respond with the filename KEYS*—the system does the rest!*

CHAPTER 23

USING A KEYWORD INDEX (1)

23.1 Functions FREQ and CREF

Note: In order to use either of these functions, a keyword index file (created by either KEYIND or BATKEY *must* exist.

23.2 Purpose of FREQ

To produce a listing of all the keywords occurring in a nominated keyword index file, the number of the field(s) in which each keyword exists, and the frequency of occurrence of each word in each field.

Clearly this function could find application in almost any VG file, e.g. in a student-record file, if "subject in which enrolled" is a field which has been used in the construction of a keyword index file, then FREQ gives a summary of the number of enrolments for each subject.

23.3 Purpose of CREF

To produce a cross-reference listing (cf. function TABUL from the main VG menu), but using entries from a keyword index file for selection.

Example of FREQ and CREF

Note: In order that the reader may follow more easily what is being done, the following example, and the examples in subsequent chapters, are based on the following set of records:

Structure

[Field 1] → AUTHOR
[Field 2] → OTHER-AUTHORS
[Field 3] → TITLE
[Field 4] → DATE
[Field 5] → PUBLISHING-DETAILS
[Field 6] → SUBJECT
[Field 7] → THEMES

Records

Record No.	*Contents*
1.	Fredericks, J.M. Collins, N.R., and Dickson, E. Victorian Literature 1976 Freedom Press, New York Literature literature prose victorian
2.	Blackburn, D.K. *Blank field* The Influence of Germany in the Pacific 1978 William Jones & Sons, London Colonialism colonialism missionaries Pacific
3.	Barrow, F. Jackson, I.S. Independence in the Islands 1978 Cowling Press, Sydney Politics politics Pacific colonial

Example of FREQ

Assuming a keyword index file called KEYS *has previously been established using* KEYIND *or* BATKEY.

```
          VG Bibliography Menu

Master file is BIBMN.
* FREQ®

Key Word Frequency Counter

Keyed Index File : KEYS®                    Enter name of keyword index file
>      19 Keywords on file.
>       3 Fields Keyed.
[  1] [  6] [  7]                           The keyed fields are reported as a
                                            reminder

Frequency count completed

Print Output file KEYS.DAT [Y/N] ? : Y®     Output file is always named XXX.DAT,
                                            where XXX is the name of the keyword
                                            index file.
```

Sample Output from FREQ

```
BARROW       [  1]  1; BLACKBURN    [  1]  1; COLONIAL      [  7]  1; COLONIALISM [  6]  1;
COLONIALISM  [  7]  1; D            [  1]  1; F             [  1]  1; FREDERICKS  [  1]  1;
J            [  1]  1; K            [  1]  1; LITERATURE    [  6]  1; LITERATURE  [  7]  1;
M            [  1]  1; MISSIONARIES[  7]  1; PACIFIC       [  7]  2; POLITICS    [  6]  1;
POLITICS     [  7]  1; PROSE        [  7]  1; VICTORIAN     [  7]  1;
```

Notes:

1. Punctuation is ignored.
2. Even single characters (such as D, F, J, etc. above) are included in the keyword index file, and thus appear as items in the frequency count.
3. The words POLITICS, LITERATURE and COLONIALISM appear twice in the report, since they occur as words in two separate keyed fields.

Example of CREF

```
        VG Bibliography Menu

Master file is BIBMN
* CREF®

Cross Reference Lister

Keyed Index file : KEYS®
            19 Keywords in Keyed Index

[ 1] → AUTHOR                           System reminds user which fields are
                                        keyed
[ 6] → SUBJECT
[ 7] → THEMES

Permanent file for Report design ? : N®          Cf procedure for COLREP
Create Design File ? N®
Enter Main Heading (75 characters)
: LISTING BY THEMES®
Name for Output file : THEME®           Enter a (max) 5-character name
Output file is : "THEME.DAT"
Width (1 to 5 chars.) of Separator Col ? : 3®
Enter Separator Symbol if required : :®
Parameters for column No.  1 Position No.  1 Remaining 132
Infield      : 1®
Chtrs        : 20®
More Columns ? : Y®
Parameters for column No.  2 Position No. 24 Remaining 109
Infield      : 2®
Chtrs        : 20®
More Columns ? : Y®
Parameters for column No.  3 Position No. 47 Remaining  86
Infield      : 3®
Chtrs        : 30®
More Columns ? : Y®
Parameters for column No.  4 Position No. 80 Remaining  53
Infield      : 7®
Chtrs        : 30®
More Columns ? : N®
New page for each keyword? : N®
Field(s) to cross reference on : 7®
Keyword : *®                            All keywords
            9 records processed
Field(s) to cross reference on : ©Z®    No more keywords
Print Output file THEME.DAT [Y/N] ? : N®
```

Sample Output from CREF

```
LISTING BY THEMES                                  (Date : 17-Dec-80)    (Page :   1)
     Reference Item ⟶ COLONIAL
AUTHOR          OTHER-AUTHORS    TITLE                   THEMES
***************************************************************************************
Barrow, F.      : Jackson, I.S.  : Independence in the   : politics Pacific colonial
                :                : Islands               :

*********************
     Reference Item ⟶ COLONIALISM
*********************
Blackburn, D.K. :                : The Influence of      : colonialism missionaries
                :                : Germany in the Pacific : Pacific
```

The output continues

```
*********************
     Reference Item ⟶ PACIFIC
*********************
Blackburn, D.K. :                : The Influence of      : colonialism missionaries
                :                : Germany in the Pacific : Pacific
Barrow, F.      : Jackson, I.    : Independence in the   : politics Pacific colonial
                :                : Islands               :
```

The output continues

CHAPTER 24

USING A KEYWORD INDEX (2)

24.1 Functions SEARCH and QSEA

Note: In order to use either of these functions, a keyword index file (created by either KEYIND or BATKEY *must* exist.

24.2 Purpose

Both functions allow searching of a keyword index file for keywords by which individual records or groups of records with common characteristics in terms of these keywords may be isolated and examined.

Thus, SEARCH and QSEA perform essentially the same task, viz., the location of a set of records satisfying conditions laid down by the user in a search definition based on keywords of a keyword index file. The only distinction is that whereas QSEA permits strikes of up to 5000 *live* records, SEARCH allows a maximum of 17000 "live" records—a "live" record being a record which satisfies the search criteria defined.

Where possible, for efficiency and economy, the user should prefer to use QSEA rather than SEARCH.

24.3 Method of Operation

This section contains a brief overview only. Details of use are contained in later sections and the examples.

First, invoke the command QSEA or the command SEARCH.

Next, by means of the command FIND, a *set* of live records is established on the basis of a *pool* of keywords.

Further discriminating commands [AND (signified by +) and EXCLUDE (signified by \) allow the list of live records to be modified by reduction; additional records may also be added by means of an un-named command (signified by ,).

At any stage, the set of live records and the pool of keywords used to establish the current selected group of records can be inspected by the commands TYPE and POOL.

Where such a set of records is to be examined further, these *complete* records may be defined as a usual VG set by means of a command DUMP, or simply the *keys only* of these records may be defined as a VG set via a command MDUMP.

Statistical information about any set found by FIND (and subsequently amended by use of +, \ or ,) can be obtained through use of the command COUNT.

To "kill" all live records established at any time, use the ZERO command.

Note: The commands AND, EXCLUDE, ",", DUMP, MDUMP and COUNT *must* be preceded by the command FIND.

A help-message may be typed in response to the * prompt, once SEARCH or QSEA has been entered, simply by typing **HELP**®.

24.4 Using SEARCH and QSEA

In response to either of the commands SEARCH and QSEA, the system requests the name of the Keyword Index file to be used in the search. If this keyword index file does not exist, return to the bibliographic sub-menu and generate it with either KEYIND or BATKEY.

Once the name of the keyword index file has been entered, the system reports

```
        0 Entries in pool
        0 Record(s) live
```

since at this stage no search has yet been undertaken.

Apart from the use of the command HELP, the next command which *must* be used is FIND, so that a set of live records may be established and a pool of keywords generated.

Example

```
. R VG®
...                                         Enter normal information regarding Mas-
                                            ter File name
* BIBLIO®                                   Enter bibliogaphic sub-menu

         VG Bibliography Menu

Master file is BIBMN
* QSEA®                                     Using command QSEA in Bibliographic
                                            menu

QSEA Routine

Keyed Index file : THEME®                   Enter name of keyed index file, which
                                            must exist
              7 keywords in Keyed Index

         0 Entries in pool
         0 Record(s) live
* HELP®                                     To obtain help-message
          How to use the Search Function
         QSEA          Is for Strikes of up to 5000 Live records.
         SEARCH        Is for Strikes of up to 17000 Live records.
         A Live Record is a strike that satisfies the search.
         In both QSEA and SEARCH the following commands are used:

         ...                                The help-message continues
         The following Commands are defined:
         AND           Records must contain named Keyword(s) as well.
         KEY           Type the names of Keyed Fields.
         POOL          On the TTY: Type contents of the POOL.
         DUMP          To a SET file the Live Records.
         EXCLUDE       Named Keywords from the search.
         FIND          Find the named Keywords.
         HELP          Type help command.
         TYPE          On the TTY: the result of the search, all live records.
         COUNT         Type frequencies of POOL
                       eg "C 10" frequencies for field 10
                       but a FIND must precede this command.
         ZERO          Clear the POOL and start again.
```

24.5 Simple Use of FIND

The basic form of the command FIND is issued in the following way:

```
FIND keyword®
```

where **keyword** is any keyword contained in the keyword index file.

The system responds with a report of the form

```
x Entries in pool
y Record(s) live
[    mm.nn% of total records]
*                                   Awaits next command
```

If you select a word which is not listed in the keyword index file, both x and y will naturally both be zero.

Remember that FIND will operate only if you have previously given one of the commands SEARCH or QSEA (which in turn implies that you are working with the bibliographic sub-menu and that a keyword index file has been generated with either KEYIND or BATKEY).

The two option now open are *(a)* to clear the pool in order to begin a different search or *(b)* to examine the set generated with the FIND command in a more intensive manner.

Where the same keyword exists in more than one field, and that same keyword is used as the parameter of a FIND command, then the records selected will be those for which the nominated keyword occurs in *any* field. If you wish to be more selective, you may issue the command in the form

```
FIND [n]keyword
```

where n is the field-number of a specified field (provided this field has been used in the construction of the keyword index file).

In this case, only those records will be selected which contain the nominated keyword *in the nominated field.*

Note also that *wildcarding* may be used when giving the keyword, e.g. **SM?TH*** will cause the location of records containing SMITH, SMYTH, SMYTHE, SMITHERS, etc.

24.6 Using ZERO

The command ZERO allows the user to drop a set of records established with a FIND command, returning to the basic SEARCH or QSEA level, so that either another fresh search may begin or else the searching session may terminate.

Following the issuing of the command ZERO, the system will always respond with

```
0 Entries in pool
0 Record(s) live
```

24.7 Functions POOL and TYPE

Once a set of records has been identified by means of a FIND command, then the user may arrange to have displayed at the terminal either *(a)* the keywords used to determine the set so selected, or *(b)* the complete records selected by the search. The commands for performing these tasks are POOL and TYPE, respectively.

24.7.1 *Using* POOL

After the FIND command has been issued, the system responds with a message of the form

```
x Entries in pool
y Record(s) live
[    mm.nn% of total records]
```

To inspect the pool of keywords used in the selection, type

POOL® *Command* POOL *may be abbreviated to* P

The sytem then displays a summary containing (i) the actual keywords, (ii) the field numbers of the fields in which each keyword occur—there may be more than one, and (iii) the frequency of occurrence of each keyword in each field in which it occurs.

24.7.2 *Using* TYPE

Having selected a set of records with command FIND, the user may inspect these *complete* records at the terminal by giving the command

TYPE® *Command* TYPE *may be abbreviated to* T

A report is then displayed at the terminal in the same way that would be obtained from a REPORT command from the main VG-menu, with the contents of *all* fields displayed, and concluding with

```
x Entries in pool
y Record(s) live
[    mm.nn% of total records]
```

24.8 More Extensive Searching

Once a set of records has been established by a simple FIND command, it may be necessary to work further with just this set of records.

Three discriminating commands (signified by +,\ and ,) are available for this purpose.

24.8.1 *Use of* AND (+)

The purpose of the command AND is to allow the user to narrow down a selection group identified with a previous FIND command, by imposing an additional condition on the records so selected.

The form of the command is

AND keyword® *or* **+ keyword®**

Then, only those records *from the original set selected by the* FIND *command* **and** which also meet the additional condition imposed by the AND command will form the new current selection group.

Thus, the usual report

```
x Entries in pool
y Record(s) live
[    mm.nn% of total records]
```

will generally reflect an increase in the value of x, and a decrease in the values of y and mm.nn.

24.8.2 *Use of* EXCLUDE (\)

This command again can be issued only if a selection group is in use, i.e. a previous FIND command has been given to select a set of records. Then, EXCLUDE is used to *drop* from this selection group those records which satisfy the new keyword condition imposed by the EXCLUDE command.

The form of the command is

EXCLUDE keyword® *or* **\ keyword®**

The system report appearing will generally be identical to that which follows the issuing of an AND command.

24.8.3 *Use of* ','

The purpose of this command (after a set has been selected with FIND) is to allow the addition to the selected set of records which did not satisfy the previous search criterion (or criteria), but which the user now wishes to include in the selection set.

The form of the command is

```
, keyword®
```

The system responds with a message of the form

```
x Entries in pool
y Record(s) live
[    mm.nn% of total records]
```

where the values of x, y and mm.nn will usually have increased from their previous values.

24.9 Commands DUMP and MDUMP

These commands allow the user to take a set selected by a FIND command and establish this group of records as a normal VG set, which may be operated on by any commands (such as SETSRT) appropriate to VG sets, and to use this set as a possible response to the system enquiry Key : when using other commands.

The difference in the two commands is that whereas DUMP will create a VG set with *all* fields of the records included, MDUMP uses *only* the key (i.e. field 1) of each record of the selected set.

In response to the command DUMP or MDUMP, the system will request a *name* for the set being generated.

24.10 Command COUNT

COUNT generates a simple statistical report on a *set selected by a* FIND *command*. The form of the command is

```
COUNT n
```

where the command-name COUNT may be abbreviated to C,
and n is the number of a field, *provided* this field is a keyed field, i.e. it has been used in the creation of the keyed index file.

The information provided by COUNT is:

(a) a list of all keywords in the keyword index file which have been extracted from the nominated field. (Optionally, the user may declare to have listed only those keywords from that field which are also contained in the set of records selected with the FIND command.)
(b) the frequency of occurrence of each of the listed keywords *in the set of selected records*
(c) the frequency expressed as a percentage of the number of records in the selected set which contain that keyword
(d) the frequency expressed as a percentage of the total number of (non-distinct) keywords in the keyword index file
(e) the total number of keywords contained in the records of the selected set
(f) the total number of (non-distinct) keywords in the keyword index file
(g) the number of live records
(h) "average frequency", which evaluates the value of the expression [sum of frequencies of keywords contained in selected records/number of *distinct* keywords for field n in keyword index file]
(i) "average per case", which is the value of the expression [value in item (f)/value in item (g)]
(j) a repeat of the message

```
x Entries in pool
y Record(s) live
[ mm.nn% of total records]
```

Note: Depending on the application and the distribution of keywords over the selected records, not all of these items of information may be meaningful. In different applications, different items of the reported statistical information provided by COUNT may be useful and/or significant.

24.11 Command Key

The command KEY provides a means whereby the user may be reminded of whifields are keyed, i.e. which fields have been used to build the keyword index file.

The form of the command is

KEY®

The system response is a report of the form

```
Keyed on Fields:
[ 1] → AUTHOR
[ 6] → SUBJECT
[ 7] → THEMES
      x Entries in pool
      y Record(s) live
      [ mm.nn% of total records]
```

This line does not appear if a FIND *command does not precede the* KEY *command*

24.12 More Complex Searching

(i) A *range* of records may be isolated by means of the FIND command in the form

F A>B

which locates all records in the range A→B.

e.g., in our example, **F BA*>BL*** will extract the records from BARROW to BLACKBURN.

This technique is particularly useful where fields concerning dates are involved.

(ii) The facility exists for searches involving a number of conditions to be carried out in response to a *single* command, rather than as a number of distinct steps.

To illustrate:

the *sequence* of commands

FIND POLITICS® — *Find records containing keyword* POLITICS

+ COLONIALISM® — *From the records selected by the previous* FIND *command, select only those which also contain the keyword* COLONIALISM

, LITERATURE® — *Now add to the selected records those records which contain the keyword* LITERATURE

\ PACIFIC® — *Now exclude from the current set those records which contain the keyword* PACIFIC

may be replaced by the *single* command

FIND POLITICS + COLONIALISM,LITERATURE\PACIFIC

With this technique, wildcarding is still permissible.

But, bracketed search requests are *not* supported, e.g.,

FIND (POLITICS + PACIFIC)\LITERATURE

is *not permissible.*

Example

Note: Referring to the Sample Output from command FREQ on page 55, for field 7 of the records given on page 54, we have the information contained in columns 1 and 2 of the table below. Reference to the original records (page 54) gives the information in column 3.

Keyword	*Freq. of Occurrence*	*Records in which Keyword occurs*
COLONIAL	1	BARROW
COLONIALISM	1	BLACKBURN
LITERATURE	1	FREDERICKS
MISSIONARIES	1	BLACKBURN
PACIFIC	2	BLACKBURN, BARROW
POLITICS	1	BARROW
PROSE	1	FREDERICKS
VICTORIAN	1	FREDERICKS

Thus:

total no. of (non-distinct) keywords = 9 (since 'PACIFIC' occurs in two records)
total no. of (distinct) keywords = 8.

The above summary (along with the original records) will be useful in tracing the development of the example below.

Note: More commands P and T are issued below than would normally be issued in a regular session. This has been done to allow the reader to follow more clearly what is happening.

Example

```
        VG Bibliography Menu
Master file is BIBMN
* QSEA®
QSEA Routine
Keyed Index file : KEYS®
                    19 Keywords in Keyed Index
                    3 Records in Master file
                    0 Entries in pool
                    0 Record(s) live
* f *®
```

Simple use of FIND *command, finding records based on complete pool of keywords*

```
                    19 Entries in pool
                    3 Record(s) live
                    [ 100.00% of total records]
* Z®
```

Clear the pool to begin another search

```
                    0 Entries in pool
                    0 Record(s) live
* F PACIFIC®
```

Find all records containing the keyword PACIFIC

```
                    1 Entries in pool
                    2 Record(s) live
                    [  66.67% of total records]
* POOL®
```

Display the keywords used thus far in this search

```
PACIFIC              [ 7] ;  2
```

PACIFIC *occurs twice in field 7 of records*

```
                    1 Entries in pool
                    2 Record(s) live
                    [ 66.67% of total records]
* TYPE®
```

Display the live records at the terminal

```
[  1] Blackburn, D.K.
[  3] The Influence of Germany in the Pacific
[  4] 1978
[  5] William Jones & Sons, London
[  6] COLONIALISM
[  7] COLONIALISM MISSIONARIES PACIFIC

[  1] Barrow, F.
[  2] Jackson, I.S.
[  3] Independence in the Islands
[  4] 1978
[  5] Cowling Press, Sydney
[  6] POLITICS
[  7] POLITICS PACIFIC COLONIAL
                    1 Entries in pool
                    2 Record(s) live
                    [ 66.67% of total records]
*  +POLITICS®
```

Select from the records which are currently live *only those records which* also *contain the keyword* POLITICS. *(This should leave only the* 'BARROW' *record.)*

```
                    3 Entries in pool
```

PACIFIC *from field 7 and* POLITICS *from* both *field 6* and *field 7*

```
                    1 Record(s) live
```

Should be 'BARROW'. *Check with* TYPE *command.*

```
                    [ 33.33% of total records]
```

```
* TYPE®

[ 1] Barrow, F.
[ 2] Jackson, I.S.
[ 3] Independence in the Islands
[ 4] 1978
[ 5] Cowling Press, Sydney
[ 6] POLITICS
[ 7] POLITICS PACIFIC COLONIAL

            3 Entries in pool
            1 Record(s) live
            [ 33.33% of total records]
* , LITERATURE®
```

Add to the set of live records those records containing the keyword LITERATURE

LITERATURE *occurs in two fields*

1 extra record has been added, viz. the 'FREDERICKS' *record*

```
5 Entries in pool
            2 Record(s) live

            [ 66.67% of total records]
* POOL®
```

Inspect the pool

```
PACIFIC              [ 7] ; 2
POLITICS             [ 6] ; 1
POLITICS             [ 7] ; 1
LITERATURE           [ 6] ; 1
LITERATURE           [ 7] ; 1
            5 Entries in pool
            2 Record(s) live
            [ 66.67% of total records]
* TYPE®
```

Inspect records

```
[ 1] Fredericks, J.M.
[ 2] Collins, N.R. and Dickson, E.
[ 3] Victorian Literature
[ 4] 1976
[ 5] Freedom Press, New York
[ 6] LITERATURE
[ 7] LITERATURE PROSE VICTORIAN

[ 1] Barrow, F.
[ 2] Jackson, I.S.
[ 3] Independence in the Islands
[ 4] 1978
[ 5] Cowling Press, Sydney
[ 6] POLITICS
[ 7] POLITICS PACIFIC COLONIAL

            5 Entries in pool
            2 Record(s) live
            [ 66.67% of total records]
* \ POLITICS®
```

Now exclude from the current *set those records containing* POLITICS *as a keyword, i.e. exclude the* 'BARROW' *record*

```
            7 Entries in pool
            1 Record(s) live
            [ 33.33% of total records]
```

```
* POOL®
PACIFIC             [ 7] ; 2
POLITICS            [ 6] ; 1
POLITICS            [ 7] ; 1
LITERATURE          [ 6] ; 1
LITERATURE          [ 7] ; 1
POLITICS            [ 6] ; 1
POLITICS            [ 7] ; 1
           7 Entries in pool
           1 Record(s) live
           [ 33.33% of total records]
```

Note that:

(i) *although the records containing the keyword* POLITICS *have been excluded,* POLITICS *still occurs as a member of the* pool, *since it has been* used *in the search since the* FIND *command was last issued, and*

(ii) *the keyword* POLITICS *occurs* twice *in the above list, since it has been* used twice *in the current search.*

```
* TYPE®
[ 1] Fredericks, J.M.
[ 2] Collins, N.R. and Dickson, E.
[ 3] Victorian Literature
[ 4] 1976
[ 5] Freedom Press, New York
[ 6] Literature
[ 7] Literature prose victorian
           7 Entries in pool
           1 Record(s) live
           [ 33.33% of total records]
* COUNT 7®                                   For the current selected set, use only
                                             keywords extracted from only field 7
Report Keyword Frequencies for defined set
Display ZERO frequency counts [N/Y]? : Y®    See Note below

Keyword                       Frequency      % of Set        % of Keywords
COLONIAL                          0            0.00%             0.00%
COLONIALISM                       0            0.00%             0.00%
LITERATURE                        1          100.00%            11.11%
MISSIONARIES                      0            0.00%             0.00%
PACIFIC                           0            0.00%             0.00%
POLITICS                          0            0.00%             0.00%
PROSE                             1          100.00%            11.11%
VICTORIAN                         1          100.00%            11.11%
                                             (n=     1)        (n=     9)
```

This last line reports that the set contains 1 record *and that there are* 9 individual (non-distinct) keywords *extracted from* all *records from field 7. Inspection of the list shows 8* distinct *keywords from field 7. Not all of these occur in the record(s) of the selected set.*

```
No. of Keywords            3 (for this set)      viz., LITERATURE, PROSE, VIC-
                                                 TORIAN
Keyword total for Field    9 (ie, all records)   3 from each of the 3 records
                                                 (PACIFIC occurring twice)
Live Records [cases]       1                     viz. 'FREDERICKS' record
Average Frequency      0.375                     =no. keywords this set/no. distinct
                                                 keywords from field 7 for all records
Average per Case       3.000                     =no. keywords from selected set/no.
                                                 of live records (i.e. records in select-
                                                 ed set)

           7 Entries in pool
           1 Record(s) live
           [ 33.33% of total records]
* ©Z®                                        Return to Bibliography menu
```

Note: The keyword list given after a COUNT *command has been issued may be made briefer by requesting that only those keywords which are contained in the selected set should be displayed. To do this, reply* **NO** *to the question* Display ZERO frequency counts [N/Y]? NO *is the default value.*

In the above example, the following output would have been obtained if **NO** *had been the user's response at this point:*

```
LITERATURE                    1          100.00%          11.11%
PROSE                         1          100.00%          11.11%
VICTORIAN                     1          100.00%          11.11%
```

Remainder of report identical with above

CHAPTER 25

FURTHER BIBLIOGRAPHIC-MENU FUNCTIONS

25.1 Functions EXTSRT, FORMAT, and SCIFOR

25.2 Pupose of EXTSRT

EXTSRT provides a more extensive sorting operation than does SORT.

Whereas SORT examines only the first ten characters of each of the fields (maximum of 4 fields) for sorting purposes, EXTSRT allows examination of a total of 100 characters across the fields on which the sort operation takes place.

Thus, if sorting with EXTSRT takes place on only a single field, sorting is done on a 100-character-wide string; if sorting with EXTSRT takes place on two fields, the user may nominate the number of characters for inspection for each field, provided the sum of these numbers does not exceed 100, e.g.,
field A—70 characters and field B—30 characters,
or field A—5 characters and field B—95 characters,
and so on.

Note, however, that EXTSRT demands that the number of characters for each field should be a multiple of 5,
i.e., field A—55 characters and field B—25 characters is permissible (55 + 25 < 100),
but field A—63 characters and field B—15 characters is not (since 63 is not a multiple of 5).

In practice, you won't be able to nominate a number which is not a multiple of 5, since the system will request from you "the width in five-character words",
i.e. if you respond with **4**, the sort will be performed on 20 (4 x 5) characters.

Also, you will not be able to exceed the 100-total maximum (see Example below).

25.3 Purpose FORMAT

To generate a mildly-flexible formatted listing of bibliographic-style material.
Note: In order to use FORMAT satisfactorily, it is *necessary* to have one field nominated as "date".

The format of the output produced is as follows for each record:

Line 1— *Either* AUTHOR (where the OTHER-AUTHORS field is blank)
or AUTHOR with: OTHER-AUTHORS (where OTHER-AUTHORS field is non-empty *and* the user requests this field to be included in line 1)
Line 1 is aligned with the left margin.

Line 2— DATE followed by OTHER FIELDS NOMINATED BY USER (in order nominated by user)
Line 2 is indented two positions from the left margin (*not alterable*)
Where the information to be contained in line 2 overflows to additional lines, these lines are indented from the left margin by a number of positions nominated by the user.

The user also nominates
(i) the width of the output (up to 132 characters maximum), and
(ii) whether he requires the TITLE to be underlined.

Normally, output should be arranged so that the authors are listed alphabetically, other-authors listed alphabetically within the author-listing, and books by the same author(s) listed chronologically. To achieve this result, SORT the file *before* requesting FORMAT.

The output will be written to a file named XXXXX.TXT, where XXXXX is the name nominated by the user.

Example of EXTSRT

```
            VG Bibliography Menu

Master file is BIBMN
* EXTSRT®

EXTENDED SORT

Fields for file order
Left field most significant
Separate field Numbers with a ","
Enter fields : 1,2®                          Sorting by AUTHOR by OTHER-AUTHORS

Width is is 5 character words.

Width in words for Field [ 1] : 3®           Sort on first 15 (3 x 5) characters of
                                             values in field AUTHOR
  17 Words remain                            85 (17 x 5) characters from other fields
                                             still available for sorting
Width in words for Field [ 2] : 5®           Sort now on basis of first 25 (5 x 5)
                                             characters in field OTHER-AUTHORS
  12 Words remain
File in field order    1 width: 3    2 width: 5   Sorting complete
            VG Bibliography Menu             System automatically returns to Bibliography
                                             menu
```

Example of FORMAT

```
            VG Bibliography Menu

Master file is BIBMN
* FORMAT®

Bibliography Format Output

Type H for Help :
Width of Output : 70®                        Nominating that output be 70 characters
                                             wide
Enter Indent : 6®                            Nominating that where line 2 runs over
                                             into additional lines, then these additional
                                             lines be indented 6 characters from the
                                             left margin
Display Table of Names ? : N®
Name for output file [5 characters] : LIST®
Output to Disk file LIST.TXT
Enter Field(s) for (1) Author(s) : 1,2®
Field for Date : 4®                          Will not accept a blank as response
Fields for Title, Publication etc [max of 10] : 3,5®
Underscore TITLE ? : N®

Key : *®                                     All records

Key : ©Z®                                    No more records

Processed        3 Records

Creating LIST.TXT
RUNOFF: LIST     1 page

            VG Bibliography Menu
```

Sample Output from FORMAT

```
Barrow, D.K. with: Jackson, I.S.
  1978 Independence in the Islands Cowling Press, Sydney

Blackburn, D.K.
  1978 The Influence of Germany in the Pacific William Jones & Sons,
      London

Fredericks, F. with: Collins, N.R. and Dickson, E.
  1976 Victorian Literature Freedom Press, New York
```

Note: Where punctuation is required (e.g. between the title and the publishing details), these symbols need to be included in the records themselves.

25.4 Purpose SCIFOR

A command SCIFOR, similar in application to FORMAT, but producing output in a form more appropriate to scientific-publications conventions, is also available.

At time of preparation of this manual, some details concerning the final format and possible options to be available within this command are still being developed; hence the absence of detail in this publication.

CHAPTER 26

EXAMPLE RECORD DESCRIPTIONS

26.1 Address files

Surname	*Key purposes*
Full Name and Title	*For labels and letters*
Address	*Multi-line fields*
Postcode	*For sorting*

The above example could be used as a base core for other types of special address files, such as staff records, and so on.

26.1.1 *Staff Record example*

Surname	*Key*
Full Name and Title	
Home Address	
Postcode	
Home Phone	
Work Phone	
Work Location	
Position	
Commence Date	
Motor Vehicle Number	
Motor Vehicle Type	

26.2 Registers

26.2.1 *Example of Computer Program Register*

Name of Program	*Key*
Purpose	
User Instructions	
Author	
Key words	*For searching*
Language	
Compile Instructions	
Notes	

26.2.2 *Cassette and Record Register example*

Number	*From LP or assigned*
Name	
Artist(s)	
Type (LP, 45, C60, etc.)	
Contents	
Brand	
Duration	

26.3 Bibliographies

26.3.1 *Articles published or unpublished*

Number	*Key, user assigned*
Name	

Principal Author	
Author(s)	
Abbreviated Abstract	
Full Abstract (optional)	
Location of Articles	
Date	
Published in	
Publisher	
Key Words	*For searching*
Library Ref.	

26.3.2 *Books and Reading Lists*

LCN Number	*Library of Congress Number*
Name	
Topic	
Description	
Author(s)	
Publisher	
Date	
Location	
Copies	

INDEX OF COMMANDS

In the following index, the code (V) indicates that the given command exists in the main VG-menu, the code (B) indicates that it exists in the Bibliography sub-menu, and the code (V/B) indicates that it exists in both menus.

The page reference directs the reader to the page on which the *usage* of the command is first explained.

Principal Author	
Author(s)	
Abbreviated Abstract	
Full Abstract (optional)	
Location of Articles	
Date	
Published in	
Publisher	
Key Words	*For searching*
Library Ref.	

26.3.2 *Books and Reading Lists*

LCN Number	*Library of Congress Number*
Name	
Topic	
Description	
Author(s)	
Publisher	
Date	
Location	
Copies	

INDEX OF COMMANDS

In the following index, the code (V) indicates that the given command exists in the main VG-menu, the code (B) indicates that it exists in the Bibliography sub-menu, and the code (V/B) indicates that it exists in both menus.

The page reference directs the reader to the page on which the *usage* of the command is first explained.