

# PRENTICE COMPUTER CENTRE

UNIVERSITY OF QUEENSLAND, ST. LUCIA, QUEENSLAND, AUSTRALIA. 4067.



## NEWSLETTER

N-272

29-March-82

### CONTENTS

- 1.0 Operations Manager.
- 2.0 VAX 11/780.
- 3.0 VAX Users - Cooperation Needed.
- 4.0 Non-Conforming Use.
- 5.0 New Version of NAG for the VAX 11-780.
- 6.0 Mathematical & Statistical Packages.
- 7.0 New Version of QDATA.
- 8.0 File Access Protection Codes for the KL System.
- 9.0 Terminal Maintenance Charge Rates 1982.
- 10.0 Information Concerning Courses.

## Principle Service Centres

	Extensions
Operations Manager	3471
Contract Programming & Feasibility Studies	3944
System Status Automatic Answering	3101
General Enquiries & Course Enrolments	3018
Training & Courses	3021

## CONSULTING

PROBLEM AREA	MAIL BOX	NAME AND EXTENSION	
Operations and Programming			
Accounts	ACCOUNTS	Ms C. Walker	2188
Cobol	COBOL	Ms B. Young	3020
Commands, system use and probs	COMMANDS	Mr R. Morrison	2962
		Mr I. Otto	2923
Database (VG, 1022)	DATABASE	Ms A. Shanahan	2951
Fortran	FORTRAN	Mr J. Currie	2924
		Mr R. Armstrong	2953
Graphics	GRAPHICS	Mr M. Williams	2835
Micros	MICROS	Mr B. Peterson	2836
Network	NETWORK	Mr B. Claire	2960
Operations	OPERATIONS	Senior operator	3212
PDP-11 software	PDP11	Mr N. Richter	3022
		Ms N. Meier	3022
Program Library, Tapes	LIBRARY	Ms L. Roberts	3943
Statistics	STATISTICS	Mr B. Maher	3021
Text processing, Typesetting	TEXT	Mr A. Broughton	2837
Non-specific problems	OPERATIONS	Senior operator	3212
Engineering and maintenance			
Development and communications	ENGIN	Mr G. Rees	3288
Mini/Micro support	ENGIN	Mr C. Lythall	3942
		Mr A. Langdon	2928
Mini and terminal maintenance	MAINT	Ms M. Kenley	3938
Griffith University:			
Consulting			7561
Computer Services			7560

### 1.0 OPERATIONS MANAGER

Sandra Campbell has been appointed Operations Manager to replace Di Ball. Sandra has 19 years experience in computer operations including 3 years at the Prentice Computer Centre.

Director  
extension 2189

### 2.0 VAX 11/780

Primary memory on the VAX 780 system has now been increased to 4 million bytes. Nevertheless, users may find that they are having difficulty in accessing the system. The problem is the number of lines from the MICOM switching system rather than the VAX capacity. Within the next couple of weeks the number of lines having access to the VAX will be increased to around sixty (i.e. double the present number of lines).

Director  
extension 2189

### 3.0 VAX USERS - COOPERATION NEEDED

Despite the increase in the capacity of the VAX system, demand is expected to be high as a result of lower charge rates and movement of student work to the system.

It would be helpful to all users if jobs involving high processor utilisation could be submitted to run after 2100 hours.

Director  
extension 2189

#### 4.0 NON-CONFORMING USE

At this time of the year it is appropriate to bring to your attention that unauthorised use of the system will not be tolerated. Those who use the system improperly are disadvantaging other users. They are breaking the statutes of the University and in some cases may be breaking the criminal law.

Users are expressly directed not to:-

- use another users PPN, account number or UIC
- interfere with the work of other users of the system
- access, copy, alter, or destroy files belonging to another user
- use others' programs without permission
- abuse or improperly use hardware or public software
- use your fund allocation for a purpose other than that for which it was granted
- use encryption techniques on information held on files
- knowingly overspend the fund allocations provided to you
- use strategies to avoid charging
- use the system in any way contrary to conditions of use which may be advised from time to time by the Director, Prentice Computer Centre or your academic supervisor.

You may hear stories of how some users may have "beaten the system". It is fair to point out that we do monitor the use of the system, and that punitive action has been taken where necessary. The intention is to give specific warning that computer abuse is unacceptable to the University.

Director  
extension 2189

#### 5.0 NEW VERSION OF NAG FOR THE VAX 11-780

Mark 8 of the NAG FORTRAN library has been implemented on the VAX11/780. This is a double precision implementation, so users should be aware of the differences in the interpretation of *italicsed terms* in the Library Manual. They are:

Manual Reference : VAX Interpretation

real : DOUBLE PRECISION (REAL \*8)  
 basic precision : DOUBLE PRECISION (REAL \*8)  
 complex : DOUBLE PRECISION COMPLEX is not available  
 in this implementation. All routines  
 involved have been modified - resulting in  
 all routines being replaced with DOUBLE  
 PRECISION arrays with an extra leading  
 dimension of 2 - modifications to the  
 COMPLEX routine parameters must be made by  
 the user:-

e.g. COMPLEX CX,A(IA,N),D(N) should be  
 modified by users to be:-

DOUBLE PRECISION CX(2),A(2,IA,N),D(2,N).  
 CX(1) will hold the real part and CX(2) the  
 imaginary part.

additional precision : QUADRUPLE PRECISION

To access the NAG library:

- (1) include the appropriate call statement in your program
- (2) \$LINK program, SYS\$STA:NAG/L

NAG manuals are available for perusal at the HAWKEN BATCH STATION.

Leonie Roberts  
 extension 3943

## 6.0 MATHEMATICAL & STATISTICAL PACKAGES

The following is a brief description of 3 packages available to users:

I.M.S.L. - Edition 8 - The IMSL library consists of a collection of subroutines and function subprograms in the areas of mathematics and statistics. The package is available on both the KA and KL systems and is accessed by the command

LOAD program, STA:IMSL/SEA

N-272

**NAG - MARK 8** - The NAG library is a collection of algorithms for the solution of numerical problems. These problems fall into the general areas of analysis and statistics. This package is available on the KL and VAX systems. The access commands are:

KL - .LOAD program, STA:NAG/SEA  
VAX- \$LINK program, SYS\$STA:NAG/L

**SPSS - Version 8.1 & 9.** SPSS is a program designed to compute those statistics used by social scientists for data analysis. It is available on the KL (version 9) and the VAX (version 8.1). The commands to use SPSS are:

KL - .R STA:SPSS  
VAX - \$SPSS/OUTPUT=(SPSS control file) (SPSS output file)

For more information on the VAX version of SPSS see the previous newsletter.

Leonie Roberts  
extension 3943

## 7.0 NEW VERSION OF QDATA

QDATA is a data entry package designed to allow a user to enter data in a convenient manner via any normal computer terminal. The package allows the user to key in data in response to prompts displayed for each field and removes the need to worry about field positioning, justification, fill, etc. These types of output alignment considerations are defined in a job definition prior to entering the data, as are certain field checks designed to prevent incorrect data being entered. Facilities are available for verifying batches of data that are keyed in, and for correcting any errors that may be detected along the way. Details of how to use QDATA may be found in technical manual MNT-4 "The QDATA Data Entry Package Introductory Guide".

This document explains new features which are in addition to the Introductory Guide. (A new Introductory Guide is still some time away).

There is a new version of QDATA on NEW: which I would like some feedback on. It has several new features which are intended to make life easier, and is slightly different to the current version on STD: in a few areas which I hope will not make life too unpleasant. New general features include:

- batch headers (selectable at job definition (JOBDEF) time)

- cross checks between expected number and actual number of records (selectable at JOBDEF time).

New features in field definitions include:

- date fields
- check digit fields
- user specified special edits
- numeric range checks
- string table checks.

These developments were primarily implemented as part of a contract being fulfilled by the PCC for the Aboriginal Health Department, but I have tried to develop the enhancements in such a way that future additions of a similar nature may be added relatively easily. Other modifications are associated with:

- outputting batches
- verifying.

At times it is difficult to look at these individually, because in some cases changes have had an impact on most facets of the package. But I shall try to tie them together in a coherent manner.

## 7.1 New General Features

Two new features impact heavily on most areas of QDATA. The first of these involves crosschecking the number of records input for a batch with an initial estimate of the number of records. The second allows the generation of batch headers.

### 7.1.1 Cross Checks On Expected And Actual Number Of Records -

Each time a batch is commenced, appended to or examined in any way, the user is asked for the expected number of records for the current batch:

Expected number of records:  
or  
Expect:

The purpose of this is to allow comparison (if desired) of the

expected number of records with the number of records actually keyed in, in an attempt to avoid the situation of a record either keyed in twice or not keyed in at all. For a particular application, it may not be appropriate to bother with checks of this nature. For this purpose, a zero value implies that no checks should be performed. A zero value supplied when a batch is being examined, appended or updated causes the current expected number to remain unchanged. Once the expected number of records is non-zero, it cannot be zeroed again.

When a JOBDEF is being created, the user is asked, as part of the new job definition, if record count checks are to be enforced:

Force expected record count(Y/N):

A positive response will prevent the expected number of records ever being zero.

#### 7.1.2 Batch Headers -

At JOBDEF time, the supervisor is asked if batch headers are required:

Batch header required(Y/N):

If the answer is No, the topic is never heard of again. If answered Yes, the user will be asked two questions at the beginning of entering any batch using that job definition:

Type:

Comments:

The "Type: " question expects up to nine characters in response. QDATA makes no use of this information apart from putting it in the batch header so that the program for which the data is destined can (if need be) distinguish between different batches that may have been output to the one file, e.g. a batch of type UPDATE will probably be intended to cause actions somewhat different to a batch of type DELETE.

The "Comments: " question will accept a response of up to 34 characters. The response is intended for visual use only, e.g. date and time of experiment, who collected data, which lecture group data applies to, etc.

### 7.1.3 OUTPUTING A Batch -

If, at sometime in the life of the batch being OUTPUT, the expected number of records becomes non-zero, a comparison will be made of the expected number of records and the number of records actually keyed in. If they do not agree, the user is warned, and given the opportunity to withhold the particular batch from output similar to the procedure for other errors.

If batch headers are invoked for the batch, a batch header will be generated before the data is output. The format of the batch header is thus:

```
Batch <n1> type <x> expected <n2> actually <n3>; <comments>
```

where	<n1>	is the batch number
	<x>	is the text supplied as batch type
	<n2>	is the expected number of records
	<n3>	is the actual number of records input
	<comments>	is the supplied commentary text

## 7.2 New Field Definitions

All new field types are a consequence of the implementation of special editing features. The very concept of a special edit implies that the field must conform to special, possibly complex, rules as supplied by the supervisor (although the supervisor would have to get a programmer from PCC to write the code to do it). Date and check digit fields are considered as commonly used special edits while range checking and string table checks are flexible byproducts of the special edit implementation. Each new field type will be considered in due course, but first, let us consider how these are defined as part of the job definition.

### 7.2.1 New Field Definitions -

The possible singular character codes for defining a field are very nearly exhausted. Hence we ended up with the following additions to the existing set of field types:

D	Date field
Q	Special (?Queer?) edit field
X	Check digit field

Many different types of date formats and check digit algorithms exist and it is hoped that many different types of special edit will be called for. Therefore the open and close parentheses "()" have been

introduced. Hence we have:

Date format n	D(n)
Special edit n	Q(n)
Check digit format n	X(n)

If the parenthesized argument is omitted, type 0 is assumed, i.e. D implies D(0). Other field definition codes can be used with these codes, e.g. fill type or justification type.

**NOTE:** Two special edit definitions in the one field definition may cause unusual results.

### 7.2.2 Field Definitions For Special Edits -

To date, the following have been implemented:

#### Date Fields

D(0) field MUST be 9 characters wide  
may have any of the following formats:

dd/mmm/yy  
d/mmm/yy  
d/mmm/yy  
99/MMM/99  
0  
any number of blanks (max. 9)

#### Notes:

The field is left justified by default; this can be overridden with a definition D(0)R.

If only one column is used for the day field, only eight characters will be accepted for the field.

0 is a valid entry - 00 is not.

It even checks for leap years.

#### Check Digit Fields

X(0) field may be up to 11 characters wide  
performs MOD 11 check digit calculation with  
check digit X -> 0.

#### Special Edits

Q(0) field MUST be 7 characters wide  
has the format mmmEnnn  
where if mmm < 800 then Ennn MUST be spaces  
if mmm >= 800 then nnn >= 800

### 7.2.3 Field Definition For Numeric Range Checks -

The special edit concepts have been taken one step further. For numeric range checking, the concept N(range spec.) has been introduced. For existing job definitions, N with no argument will mean the same as it used to. However, numeric range checks are specified thus:

```
N(*-n1,n2-n3,n4, ... ,n5-*)
where *      on left of a range implies -infinity
      n2-n3  n2 <= n <= n3
      n4     n = n4
      *      on right of a range implies +infinity
```

Notes:

n4 actually implies  $n4 \leq n \leq n4$ , hence  $n = n4$ .  
 \*-n1 implies  $n \leq n1$ .  
 n5-\* implies  $n \geq n5$ .  
 \* by itself implies  $-\infty \leq n \leq +\infty$ .  
 +/-infinity is the maximum single precision number.

e.g. N(0,5-25,99)  
 will allow an integer value of 0, 99 or within the range 5 to 25 inclusive.

If a decimal point is included anywhere in the range definition, floating point numbers will be accepted for input (including E format numbers, e.g. -1.6E-2). Otherwise, only integers will be allowed as input. The range validity check is performed at the completion of the field.

### 7.2.4 Field Definition For String Table Checks -

The parentheses construct was taken one stage yet further. If the open parenthesis "(" is not prefixed with D, N, Q or X, a string table is being defined. The idea is to set up a number of character patterns that a field must EXACTLY match. So consider the construct:

```
(<delim><str1><delim><str2><delim> ... <delim><qualifiers>
```

Whatever the first character after the "(" is, it becomes the delimiter. The string list should be followed by suitable "normal" field definition codes to describe the characters in the string table, e.g.

```
(" "Y"N"*")BACS
```

where valid characters are

```
<spc>
```

Y  
N  
\*

**BEWARE!!**

- One space is NOT the same as two spaces.
- The example above will not allow a null response (i.e. no characters typed before typing <field complete>). To allow this, use (" "Y"N"\*")BACS.
- Select as a delimiter a character that does NOT appear in any string in the table.
- If ")" is to be a delimiter, there must be 2 of them at the end of the string definitions.

### 7.3 "Slight" Differences

A number of other modifications have been made in response to user pressure. These are apparent when deleting a field, when outputting a batch and when verifying.

#### 7.3.1 <Esc>Delete Field -

When issued during input of a record, the <esc>Delete Field command no longer causes the retyping of the current record. This was considered to be too costly for large records, especially when the command was given near the end of a large record. Now, only the current field prompt is retyped. This may cause some problems with job definitions with special screen layouts.

#### 7.3.2 Prematurely Terminated Fields -

Fields that are by definition required to be filled, and are accidentally terminated prematurely, previously would have had a <esc>Delete Field performed for them when they tried to delete the <field complete> character (with all that used to go with that). Now, the field complete character can be deleted, and input resumed.

### 7.3.3 Outputting A Range Of Batches -

Until now, if a supervisor specified a range of batches to be output (and only one range could be specified), only batches entered using the same job definition as the first batch in the specified range would be output. Now, ALL batches in the specified range will be output, irrespective of job definitions. Also, multiple ranges can be specified. Thus, the following commands may be given:

```
*SUPERVISOR
*OUTPUT FRED.DAT=2-10,40-90,20
```

If batches had been entered in the order

```
4, 5, 11, 12, 20, 30, 80, 10, 50
```

batches would be output to FRED.DAT in the order

```
4, 5, 10, 50, 80, 20
```

Note that if batch headers had been defined for any of the batches, they would appear in the appropriate places. Note also that upon giving commands of the type

```
*SUPERVISOR
*BATCH 2
*OUTPUT FRED
```

nothing useful would happen. What should be typed is

```
*SUPERVISOR
*BATCH 2
*OUTPUT FRED=
```

or

```
*SUPERVISOR
*OUTPUT FRED=2
```

### 7.3.4 Changes To Verify Fields -

If a record has a number of VERIFY fields in it and one is changed (i.e. an error is accepted or the field is reentered), only that field will be reverified. Previously, all verify fields in the record would be reverified.

### 7.3.5 Modifying Batches Input With Old Versions (<176) Of QDATA -

If an attempt is made to call up a batch which was input using a previous version of QDATA, the user may be told that he has 0 records out of an expected number of 2 records. This is erroneous, and will be corrected if the batch is examined or modified in any way.

If anyone has any problems with NEW:QDATA, please direct consulting mail to COMMANDS with as many details of the problem as possible.

Bryan Claire  
extension 2960

## 8.0 FILE ACCESS PROTECTION CODES FOR THE KL SYSTEM

There has been some confusion about the meaning of the file protection codes on the KL, since they are different from the KA. There are three fields in the protection code. The first refers to the owner of the file (Table 1) the second to members of the same project and the third to everyone else (Table 2).

On the KA, where File Daemon does not exist to extend the protection system, field 1 has the same codes as fields 2 and 3, except that the owner may always change the protection.

Table 1  
Protection codes for field 1 - KL System  
Owner Protection Codes

Code	Accessibility by Owner
7,6 or 3	You (the owner) may only change protection, execute and read the file.
5 or 2	You can change protection, execute, read, append to, update and supercede the file.
4,1 or 0	You have full access to the file.

The owner field should not be higher than 3 unless some special case requires allowing access by others who would otherwise be denied. The highest protection is afforded by code 377, which prevents the owner from editing, though he can always read his files.

Table 2  
Protection codes for fields 2 and 3 - KL (and field 1 on KA)  
Project Members and Other User Protections

Code	Access privileges *
7	Users other than the owner cannot access the file.
6	Only execution allowed.
5	User can execute and read the file.
4	User can execute, read and append to the file.
3	User can execute, read, append to and update the file.
2	User can execute, read, append to, update and supercede the file.
1	User can rename the file as well as the above.
0	User has all access including ability to change the protection.

- \* In addition, the user may be allowed further access if the owner protection code (Field 1) is 4 or greater and the owner has allowed it in an ACCESS.USR file and the FILDAE job is running. FILDAE runs on the KL but not on the KA.

Ian Burgess  
extension 2959

## 9.0 TERMINAL MAINTENANCE CHARGE RATES 1982

The following charge rates will apply for maintenance of terminals from 1 January, 1982 for the University of Queensland and Griffith University. This service is not available to groups external clients.

LA120	\$260 p.a.
LA36 Decwriter II	\$200 p.a.
TTY43	\$180 p.a.
ADM35	\$120 p.a.
* TI733 ASR or KSR	\$240 p.a.
* TI745	\$280 p.a.
* GE Terminet	\$280 p.a.
* VT55	\$300 p.a.
Teleray 1061	\$200 p.a.
ADM31	\$200 p.a.
VIS200	\$150 p.a.
VT100	\$200 p.a.
* VTE-5	\$200 p.a.
VIS100	\$200 p.a.

Clients requesting new or changed maintenance arrangements are asked to write to G. Rees, Engineering Analyst at the Prentice Computer Centre.

N-272

\* These terminals will not be maintained in 1983.

Graham Rees  
extension 3288

## 10.0 INFORMATION CONCERNING COURSES

### 10.1 General Notes:

1. Courses conducted by the Centre are free to all staff and post-graduate students of University of Queensland and Griffith University. All others must pay the prescribed fee.
2. Instruction in the use of terminals, file-manipulation and editing is provided in the course "Introduction to PDP-10". This knowledge is essential before enrolling in any other course.
3. All courses listed below will be held in the Client Room, Hawken Building, St Lucia.
4. The following list may be obtained by TYPEing or PRINTing the file

HLP:COURSE.HLP

### 10.2 Schedule:

Unfortunately, Barry Maher, our Education Officer, was recently taken ill and will be on sick leave for an extended period of time.

We also have severe staff shortages due to several recent resignations and consequently must cancel the following courses:

Demonstration of alternative : March 23  
version of RUNOFF 2.30-4.30pm

BASIC Programming : April 6-8  
3 full days 9-12am+2-5pm each day

Statpack : April 14-15  
2 half days 2-4pm each day

RUNOFF : April 19-23  
5 half days 9-12am each day

The courses scheduled below will depend on the availability of staff. If it becomes necessary to cancel or alter any of these courses the people enrolled will be notified. Further information will be published in future newsletters.

We apologize for any inconvenience caused.

#### April

SPSS : April 19-23  
5 half days 1.30-5pm each day

#### May

FORTRAN Programing : May 10-14  
5 full days 9-12am+2-5pm each day

VG : May 24-28  
5 half days 9-12am each day

1022 : May 24-28  
5 half days 2-5pm each day

#### June

Introduction to PDP-10 : June 7-8  
2 full days 9-12am+2-5pm each day

Statpack : June 9  
1 full day 10-12am + 2-4pm

Introduction to PDP-10 : June 10-11  
2 full days 9-12am+2-5pm each day

BASIC Programming : June 16-18  
3 full days 9-12am+2-5pm each day

RUNOFF : June 21-23  
3 full days 9-12am+2-5pm each day

SPSS : June 29-July 1  
3 full days 9-12am+2-5pm each day

#### July

VAX Conversion Course : July 2  
1 full day 9-12am + 2-5pm

FORTRAN Programming : July 5-9  
5 full days 9-12am+2-5pm each day

N-272

Plotting/Graphics

: during week July 12-16  
Details to be finalised.

### 10.3 Enrolments And Enquiries

Please note that enrolments for all courses should now be directed to extension 3018.

Allan Woodland  
extension 2921

\* \* \* \* \*