NATIONAL INSTITUTE OF OCEANOGRAPHY
WORMLEY, GODALMING, SURREY

N.I.O. Computer Programs 13

N. I. O. Internal Report No. N.13

JULY 1969

N.I.O. COMPUTER PROGRAMS 13

edited

by

B. J. Hinde

N.I.O. Internal Report No. N.13

National Institute of Oceanography

N.I.O. PROGRAMS 13

1) PROGRAMS

80	Approximation to a Directional Wave Spectrum	ASPE
87	Gravity Anomalies	GYAN
88	Atlas Costs	COST
88/A	Reading Atlas Costs from disk	SUMS
88/B	Initialising Atlas Costs data file	ZERO
89	Shipborne wave recorder analysis	SBWRO
89/A	Shipborne wave recorder analysis	SBWR1
113	Card sequence numbering	CIDRO
114	Error correction of Current Meters	ERC01
115	Tape Security System: File initialisation	INITO
116	Tape Security System: Query File	QUERO
118	Gulf of Aden Magnetic Reduction	
119	80/80 Card Reproduce	

2) SUB-PROGRAMS

-6/A	Find a file	FILE2
-15	Tape Security System: Find a file and record	FINDO
-16	Tape Security System: Store current tape position	STORO
-17	Tape Security System: Find next available file and record	NEXTO

3) DPG/P/7 N.I.O. Tape Security System

N.I.O. PROGRAM 80

Title	Approximation to a Directional Wave Spectrum
Program Name	ASPE
Language	1800 Fortran IV
Machine	IBM 1800
Purpose	To estimate the directional spectrum of sea waves from values of the first four angular harmonics (as obtained using NIO program 68 for example).
Input	Job and Data

Job Description	// JOB
	// *(Job No./Name/Title)
	// XEQ ASPE
	*CCEND

Data

This immediately follows *CCEND and consists of the following cards:-

1) The title of the data, commencing anywhere other than column 1. If there are several sets of data, then the title of the last set of data must be preceded by < in column 1.

2) The total number of frequencies, M, equal to the number of frequencies at which the directional spectrum is required.

Format: I3

Then follows this sequence for each of the M frequencies

3) An integer representing the frequency

Format: F5·1

4) and 5) Angular harmonics a_1 and a_2, b_1 and b_2

Format: E12·5, 2X, E12·5

6) Energy, followed by a calibration factor to bring the resulting spectrum to the required units (e.g. 3·2808 for converting metres to feet).

Format: E12·5, 2X, F8·5

The next set of data then follows, starting with the data title as specified in 1).

Output

The output commences with the words

NIO 80 OUTPUT on a new page

This is followed by the title of the first set of data, and M.

Then the angular harmonics a_1 to a_4, b_1 to b_4, and

c, s_1, s_2, and gamma are output.

Under a new heading comes the normalised angular distribution of energy, computed at 15° intervals, for each frequency. The angles are the directions of propagation of energy.

The final block of output consists of the directional spectrum of energy in physical units, at 15° intervals at each frequency, where the angles are directions of arrival of energy.

The last set of results is terminated by the words

END OF DATA

Restrictions — BM≤15, where B is the number of sets of data.

Execution Time — Approx. 60M secs.

Method — As for Atlas version of NIO 80.

Programmer — Catherine Clayson

N.I.O. PROGRAM 87

Title	Gravity Anomalies
Name	GYAN
Language	1800 FORTRAN IV
Machine	IBM 1800
Purpose	To compute the anomaly in the earth's gravitational field along a profile perpendicular to parallel horizontal prisms of infinite lateral extent and specified density.
Input	Job and Data

Job Description	// JOB
	// *(Job No./Name/Title)
	// XEQ GYAN
	*CCEND

Data

This immediately follows *CCEND and consists of the following cards:-

1) Integer representing the total number of sets of data.

Format: 3X, I3

2) Title of the first set of data commencing anywhere other than in column 1.

3) A', B', C', Z', and I';

where the anomaly has to be calculated at points on z = Z' at interval B' in x between A' and C' (B' must be positive). 0 or 1 is inserted according to whether the units of x and z are nautical miles (0) or kilometres (1).

Format: X, 4F6.1, I4

4) D', the density of the first prism in grms/cc.

Format: X, F6.1

5) X_0, Z_0; X_1, Z_1; .... X_n, Z_n; X_0, Z_0;

the coordinates of the corners of the first prism listed in clockwise direction round the prism.

N.B. Coordinates of the first corner are repeated at the end of the list.

Format: 2F7.1

The remaining prisms are also listed, commencing with their density; the density of the final prism in that set being preceded by a < in column 1.

The next set of data then follows, beginning with the title of that set of data.

The density of the final prism in the last set of data should be preceded by a / in column 1.

Output	For each set of data:- The title of the set of data The x-coordinate at the points at which the anomaly is to be calculated and the anomaly in milligals, printed in two columns. The output is terminated with the words: END OF DATA
Restrictions	No prisms may have more than 200 corners and the number of anomalies to be computed must be ≤ 200.
Failures	If a prism contains more than 200 corners the words K GREATER THAN 200 are output, and the program proceeds to the next set of data. If the initial coordinates of a prism are not repeated at the end of a list of points the words ERROR IN PRISM followed by the prism number are output.
Execution Time	Approximately 10N secs per set of data; N being the total number of coordinates in that set.
Method	As for ATLAS NIO 87.
Programmer	Catherine Clayson

N.I.O. PROGRAM 88

Title Atlas Costs

Program Name COST

Language 1800 Fortran IV

Machine IBM 1800

Purpose To evaluate the cost of each job run on Atlas 1 in pounds, shillings and pence; and also to form a cumulative total cost for each project if required, on disk.

Input Job and data on cards.

Job Description

```
// JOB
// *(Job No./Name/Title)
// XEQ COST
*FILES(4,ATCOS,0)
*CCEND
```

Data This immediately follows *CCEND and consists of the following cards:-

1) The title of the data, commencing anywhere other than in column 1. (This usually contains the two dates covering the current Atlas invoice.)

2) One digit in column 1, either 1 or 0 depending on whether it is required that the job cost be added to the cumulative total cost or not.

Then follows this sequence for each run. Data should commence in column 2 for every run except the last which must contain a '/' in column 1.

a) Job number
b) Run number
c) Data link number (0 if not run over the link).
d) Day and month on which the job was run.
e) Total number of instructions used.
f) Number of instructions used in compilation
g) Requested number of blocks for compilation store
h) Requested number of blocks for execution store.
i) Number of tape decks used.
j) Number of tape block transfers used.
k) Total number of blocks input
l) Total number of blocks of tape output.
m) Number of records of lineprinter output
n) Number of cards punched.
o) Level of job - E = express
N = normal
P = priority
S = special stationery, normal

Typical card layout:-

```
1 3       9     14    19        26        33      39    44      50 53
bbXXXbbbXXbbbXXbbbXX.XXbbXXXXXXbXXXXbbXXXbbXXXbbbXbbXXXbbb.
   a      b     c       d         e         f     g     h     i  j

59  63  67        74      80
XXbbXXbbXXXXXbbXXXbbbX
k   l   m       n     o
```

where b represents a space
X represents a 1 digit integer
XX represents a 2 digit integer, etc.
XX.XX is a convenient notation for the date
i.e. for 8th November put 08.11
a, b, c, — o represent the data items as described above.

N.B. All figures must be right-justified in their field.

Output

The output commences with the title of the job as punched on the input card. Then the following headings are output:-

JOB RUN D/L DATE INS £ S D

The appropriate information is printed in the various columns followed by the cost of the run. The last run is terminated with the words

END OF DATA

If a '1' was punched on the second card, the total cost of each run under a particular project number (job) is added to a cumulative total, on disk, for that project. (Each project has allocated to it one 'record' which contains this information)

Restrictions

Data is restricted to the format specified above.

Execution time

Approx. 1 sec per card.

Method

The program computes the following formula:-

$$\left\lceil \frac{(CS' + AN' + 4JN')}{263} + 120M \right\rceil \text{ pence + I/O charges}$$

where $S' = S$ if $S \leq 80$

$S' = 80 + \frac{S - 80}{2}$ if $80 < S \leq 120$

$S' = 100 + \frac{S - 120}{4}$ if $120 < S$

(an identical relationship connects N' and N).

C = number of instruction interrupts obeyed during compilation
S = number of store blocks requested for compilation
A = number of instruction intterupts obeyed during execution
N = number of store blocks requested for execution.
J = number of magnetic tape block transfers during execution
M = number of magnetic tape decks loaded

If a job is run at express rate the program computes the input/output charges as follows:-

1/- per block of cards or tape input
+ 2/- per block of tape output
+ 2d per card output
+ $^1/_3$d per line of lineprinter output
(or $\frac{1}{2}$d if output is on special stationery)

For a normal or priority job the charges are:-

1/6	per block of cards or tape input
+ 4/-	per block of tape output
+ 2d	per card output
+ 1/3d	per line of lineprinter output
(or ½d	if output is on special stationery)

For a normal job the formula cost, input cost and output cost are simply added.

For an express job the final cost is

[£1 + 3/2 (formula + input + output)]

For a priority job the final cost is

[£2 + 4/3 (formula + input + output)]

For a job using special stationery the final cost becomes

[£2 + (formula + input + output)]

The program evaluates the relevant cost for each job in pounds, shillings and pence. There is always a minimum charge of £1.

In order to obtain a print-out of the total cost to the present data for each project, one must execute N.I.O. PROGRAM 88/A.

Also, to initialise the Atcos data file every April 1st N.I.O. PROGRAM 88/B must be executed.

Programmer — Catherine Clayson

N.I.O. PROGRAM 88/A

Title	Reading Atlas Costs from disk
Program Name	SUMS
Language	1800 Fortran IV
Machine	IBM 1800
Purpose	To obtain a print-out of the total cost to the present date of all runs on Atlas 1, under project numbers.
Input	Job on cards

Job Description	// JOB
	// *(Job No./Name/Title)
	// XEQ SUMS
	*FILES(4,ATCOS,0)
	*CCEND

Output	The output commences with the headings:- TOTAL COST OF PROJECTS FROM APRIL 1st TO THE PRESENT DATE (AS ABOVE) followed by:- JOB £ S D The appropriate information is then printed in the various columns.
Execution Time	Approx. 2 minutes
Programmer	Catherine Clayson

N.I.O. PROGRAM 88/B

Title	Initialising Atlas Costs data file
Program Name	ZERO
Language	1800 Fortran IV
Machine	IBM 1800
Purpose	To set the contents of the ATCOS data file to zero. (Usually executed every April 1st.)
Input	Job on cards.

Job Description	// JOB
	// *(Job No./Name/Title)
	// XEQ ZERO
	*FILES(4,ATCOS,O)
	*CCEND

Output	The output consists of the words:- ATCOS DATA FILE - TOTAL COST FOR ALL PROJECTS HAS BEEN SET TO ZERO
Execution Time	Approx. 1 minute
Programmer	Catherine Clayson

N.I.O. PROGRAM 89

Title Shipborne wave recorder analysis

Name SBWRO

Machine IBM 1800

Language 1800 Fortran IV

Purpose Given values of the highest and second highest crests, the lowest and second lowest troughs, the number of zero crossings and the number of crests in a short record from the N.I.O. shipborne wave recorder, to compute the spectral width parameter and the significant wave height and also the predicted maximum height in a period of three hours, and store the results on lineprinter and disk.

Inputs Job, Parameters and Data.

Job

cc$_{18}$

//bJOB 14

//b*(Job number/Name/Title)

//bXEQbSBWRO

*FILES(1,NAMEP,1)

*CCEND

where Name is the user's name, and NAMEP is the name of a disk file of 400 sectors length.

Parameters Immediately following the *CCEND is the line number, format I5. This number, which is set to 1 at the start of a year's analysis, is raised by 1 after each record has been printed on the 1443 printer. On the next card, format F6.3, is the depth in feet of the instrument below sea surface. The third card contains the data title in format 40A2.

Data Each month's data is preceded by two cards containing the month number in format I2, and the year in format I4. Then follows the data, one card per record, containing:

a) M - the terminator < for the last record in the month, or / for the last input record (Format A1)

b) The day of the month (I2)

c) The time at which the record started (I6)

d) The length of record in minutes (1X,F6.1)

e) The height in feet of the highest crest above the mean water level (F6.1)

f) The height in feet of the second highest crest above the mean water level (F6.1)

g) The height in feet of the lowest trough below the mean water level (F6.1)

h) The height in feet of the second lowest trough below the mean water level (F6.1)

i) The number of zero crossings in the record (I7)

j) The number of crests (above and below the mean water level) in the record (I7)

k) "M", "C" or "F" to indicate record missing, record calm, or record faulty. Format A1.

Output

The lineprinter output is headed with the name of the ship on one page, and the month and year on the top of each successive page. Each page is then headed with

DATE TIME TZ H1 H2 E H1' H2' HS1 HS2 HS HMAX(3HRS.) DURATION

The month number and year are written to disk at the beginning of every month's output.

The program then computes the following, and prints the answers on the 1443 printer and stores them on disk:-

a) The day of the month

b) The time at which the record started

c) The wave period in seconds, correct to two decimal places (T_Z)

d) The height, in feet, correct to one decimal place, of the highest crest plus the lowest trough (H_1)

e) The height, in feet, correct to one decimal place, of the second highest crest plus the second lowest trough (H_2)

f) The spectral-width parameter (E) to three decimal places

g) H_1 corrected to allow for instrumental response (H_1')

h) H_2 corrected to allow for instrumental response (H_2')

i and j) The mean height of the highest one-third of the waves each correct to one decimal place calculated from H_1' and H_2' respectively (HS_1 and HS_2)

k) The average of HS_1 and HS_2 to give HS. The program computes the average of the numbers contained in the machine store and not those printed under HS_1 and HS_2

l) The most probable height of the highest wave which would occur in a period of three hours, correct to one decimal place (MAX(3HRS))

m) The duration of the record in minutes

n) If no measurements were available, the words

"RECORD FAULTY" or
"RECORD MISSING"

will be printed at the right-hand side of the page against the appropriate date and time. If the record was too small to measure (i.e. $H_1 < 1$ ft) the word "CALM" will be printed in the TZ column. In the above cases the letter "F", "M" or "C" is stored in word 1 of the disk record.

o) The line number which increases by 1 throughout the year's results, after each record or newline between days is printed on the 1443 printer or disk.

Method

The zero-crossing period is given by

$$T_z = 1200U/[2N_z] \text{ seconds}$$

then $H_1 = A + C$

$$H_2 = B + D$$

and these are corrected for instrumental and hydrodynamic factors using the formulae

$$H_1' = k H_1$$

$$H_2' = k H_2$$

where $k = 0{\cdot}83\ [1 + (8{\cdot}8\mu)^{-2}]^{3/2} \exp(2{\cdot}5d\mu^2/g)$

where $\mu = \frac{2\pi}{T_z}$ and $g = 32{\cdot}174$ ft. sec.$^{-2}$

The spectral-width parameter

$$E = \sqrt{1 - \frac{(T_c)^2}{(T_z)^2}}$$

where $T_c = 600U/N_c$ seconds.

The values of significant wave height as determined from H_1' and H_2' are

$$Hs_1 = \frac{2 H_1'}{(2\theta)^{\frac{1}{2}} (1 - 0{\cdot}2890^{-1} - 0{\cdot}2470^{-2})}$$

and $Hs_2 = \frac{2 H_2'}{(2\theta)^{\frac{1}{2}} (1 - 0{\cdot}2110^{-1} - 0{\cdot}1030^{-2})}$

where $\theta = \log_e (N_z)$

The average significant wave height is then given by

$$Hs = \frac{Hs_1 + Hs_2}{2}$$

The prediction of $H_{max(3hrs.)}$ is done by computing

$$y = \sqrt{\log_e \left(\frac{180\ N_z}{U}\right)}$$

and then

$$H_{max(3hrs.)} = \frac{\sqrt{2}}{2} (0{\cdot}006361y^4 - 0{\cdot}073968y^3 + 0{\cdot}330573y^2 + 0{\cdot}316548y + 0{\cdot}566405)\ Hs$$

The coefficients in this equation were derived from a least squares fit of the points contained in a table by Longuet-Higgins (J. Mar. Res. 11, 259) relating the most probable value of a_{max} in a given interval to the square root of the number of waves in the interval. For $10 \leq N_z \leq 100,000$ the maximum residual is less than 0.05%.

Restrictions — Each disk file is large enough to store 1 year's data. It is assumed that all the data from each station will be analysed during one run, as the program always writes to the first record in each file.

Programmer — Eileen Page.

Data Processing Group

A D D I T I O N

N.I.O. 89 SBWRO version only

Add after "Restrictions":

Note — Each year's records will be analysed from Jan 1st to Dec 31st, even if the original wave records started in the middle of a year. This disposes of several problems which may occur during further analysis.

N.I.O. PROGRAM 89/A

Title	Shipborne wave recorder analysis
Name	SBWR1
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	To analyse wave records obtained from a shipborne wave recorder using the same measurements as described in NIO 89.
Inputs	Job, parameters and data.

Job	cc 18
	//bJOB 14
	//b*(Job number/Name/Title)
	//bXEQbSBWR1
	*FILES(1,,1)
	*CCEND
	The parameters and data are exactly the same as those used in NIO 89.

Note	This program is to be used whenever four or less month's data is to be analysed, and the results are not required for further analysis. Results are stored on disk, but are destroyed by the next job.
Programmer	Eileen Page.

N.I.O. PROGRAM 113

Title	Card sequence numbering
Name	CIDRO
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	To re-identify a deck of cards given the identification letters, the first sequence number and the increment.
Inputs	Job, Parameters and Data.

Job Description	//bJOB
	//b*(Job No./Name/Title)
	//bXEQbCIDRObbbFX (b = blank)

Parameters	These immediately follow // XEQ CIDRO FX and are punched on one card:-
	FIRSTbNO.=XXXX INCREMENT=YY PROG.NAME=ZZZZ
	(cols. 1 - 14) (cols. 21-32) (cols. 41-54)
	ZZZZ are four characters used to identify the program and are punched in cols. 73 - 76 of each non-blank output card.
	XXXX is the first identification number, and is output in cols. 77 - 80.
	YY is the increment added each time to the first number.
	XXXX and YY should be punched right justified.

Data	This immediately follows the parameter card and is comprised of the program to be re-numbered without its monitor control cards. The program may contain up to three consecutive blank cards, but four blank cards are taken to be the end of the deck, and no more cards will be read in.
Output	The renumbered program is output onto cards which collect in the second output hopper. The output is also written on the 1443 printer and is followed by this message:- X CARDS OUTPUT (PUNCHED AND BLANK) IDENTIFICATION NUMBERS FROM NAME 1234 TO 5678 A message is also output for the computer operator on the 1443 (or on the 1816, dependant on the value of 'IOD' set in line 9 of the program) after the cards have been read in, but before punching:- X CARDS LOADED TO DISK PLEASE CHECK THAT THERE ARE NO PUNCHED CARDS IN THE HOPPER PLEASE LOAD AT LEAST Y BLANK CARDS, THEN PRESS - START -

Restrictions — The maximum number of cards to be renumbered on one run is 2500.

Failures — If four consecutive blank cards are found this will be taken as a terminator. CIDRO executes a 'wait' before punching so that remaining input cards could be removed from the hopper.

Execution time — 1·5 seconds per card.

Method — The subroutine STACK is used to stack the output cards in the second output stacker.

Programmer — M. H. Johnson

N.I.O. PROGRAM 114

Title Error correction of Current Meters

Name ERC01

Machine IBM 1800

Language 1800 Fortran IV

Purpose To correct current meter data (calibrated) on ½" magnetic tape according to corrections read from cards. The corrected data is to be written to a second magnetic tape in the next available file.

Input Job description parameters and correction data

```
                                  cc19
// JOB                             8
// *Job No./Name/Job Title
// XEQ ERC01
*FILES(1,N),(201,MN,O)
*CCEND
```

where 8, right-justified in cols. 15-19 of the job card, is the number of a disk pack containing at least 327,670 words of non-process working storage, and N is the number of the magnetic tape on which the corrected data is to be written.

The data consists of two parameter cards followed by NCORR correction cards, one for each record on magnetic tape that is to be corrected.

Card 1 IFIL, the file number where the data to be corrected resides, right justified in card columns 1-7.

Card 2 MN, LREAD, NCORR right justified in card columns 1-7, 8-14, 15-21 respectively. MN is the meter number for the data, LREAD is the serial number of the last required reading and NCORR is the number of records to be corrected.

Then follows the NCORR correction cards in format

I7, 1X, 3(F7·2, 1X)

each card containing reading number, current speed, direction and temperature.

```
                cc      7      15      23      31
e.g.  bbbb472bbb14·25bb252·41bbbb9·72
```

The magnetic tape data should be in similar format, as it will be if written by N.I.O. program 111, Current Meter Analysis.

Output The new magnetic tape will be in the same format as the old i.e. First record: the meter number, I7 format,

then I7, 1X, 3(F7·2, 1X)

On the 1443 printer a monitor of every correction made, preceded by the previous correct or corrected record, will be printed.

Method

The magnetic tape is first wound to the beginning of the required file, IFIL. The complete file up to and including reading LREAD is then read to disk (a total of K records). After changing the magnetic tape, the data is read back to tape, correction data being read from cards and substituted when the reading numbers agree. The output will commence at record 1 of the next available unwritten file (NFIL). At the conclusion of the program the following message is printed:

FILE IFIL ON FIRST TAPE HAS BEEN CORRECTED AND COPIED TO FILE NFIL ON SECOND TAPE

K RECORDS WERE WRITTEN

Operation

Load tape, disks and cards in the normal way. When all the data has been read from tape the following message will appear on the printer:

CHANGE MAGNETIC TAPE AND PRESS CONSOLE START

Change the magnetic tape and then press console start.

Restrictions

Number of records of data <32767

IFIL ≤ 99

If NCORR = 0 the tape will be copied with no alteration.

The program operates under the M.I.O. Tape Security System and both magnetic tapes in use must have been initialised and used under the system [see DPG/P/7].

Execution Time

Approx. (12*IFIL + K/3) seconds where K is the number of records on the file being corrected.

Programmer

Brian Hinde

N.I.O. PROGRAM 115

Title Tape Security System: File initialisation

Name INITO

Machine IBM 1800

Language 1800 Fortran IV

Purpose To initialise any one of 30 files kept on disk with the next available file number and record number on the corresponding magnetic tape. For an explanation of the N.I.O. Tape Security System see the publication DPG/P/7.

Input (e.g.)

```
// JOB
// *Job No./Name/Job Title
// XEQ INITO
*FILES(5,M5,0),(6,M6,0)
*CCEND
      CC7     14      21
       5       2       1
       6       1     100
*END
```

will initialise the disk file corresponding to tape 5 with file 2, record 1 and the disk file corresponding to tape 6 with file 1, record 100 as the next available files and records.

There should be as many *FILES items as there are data cards and all the items should be of the form

(K,MK,0)

where K is the magnetic tape number to the initialised.

The data ends with a *END card.

Output The output consists of one line for each file as follows: (e.g.) TAPE SECURITY FILE M6 INITIALISED WITH FILE NUMBER 1 AND RECORD NUMBER 100

Restrictions K≤30 and K≠12 at present.

Execution time About 4 minutes.

Programmer B. J. Hinde

N.I.O. PROGRAM 116

Title	Tape Security System: Query file
Name	QUERO
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	To examine the current status of all the disk files giving details of the next available files and records on all N.I.O. magnetic tapes. For an explanation of the N.I.O. Tape Security System see the publication DPG/P/7.

Input	// JOB
	// *Job No./Name/Job Title
	// XEQ QUERO CC FX

Output	For every magnetic tape in use: (e.g.) TAPE SECURITY SYSTEM - NEXT AVAILABLE RECORD ON TAPE M 6 IS 532 ON FILE 3.
Execution Time	About 30 seconds.
Programmer	B. J. Hinde

N.I.O. Program 118

Title	Gulf of Aden Magnetic Reduction
Machine	Atlas
Language	Fortran V
Purpose	Given navigation, daily variation and secular variation data the program converts magnetometer readings to gammas (if necessary), corrects them for daily and secular variation and calculates the anomaly value at each observation relative to the International Geomagnetic Reference Field. The program as written can only be used for observations in the Gulf of Aden using daily variation data from the Addis Ababa Geophysical Observatory.
Job Description	JOB S002....., G OF A MAGNETIC REDUCTION INPUT 1 S002 daily variation data title 2 S002 magnetometer readings data title 3 S002 navigation data title 5 S002 FIELD COEFFICIENTS OUTPUT 0 1 BLOCK 1 N BLOCKS 6 1 BLOCK TAPE 99 SLOAD2 LS261*WRITE INHIBIT COMPUTING A INSTRUCTIONS EXECUTION B MINUTES STORE 40/25 BLOCKS COMPILER LOAD 036/NIO 118 18/3/1969 ***Z As an aid to chosing the variables N, A and B it may be noted that an input of 7 blocks of magnetic data and 5 blocks of navigation data gave 33 blocks of output and required 6900 instruction interrupts.
Data	All data tapes should start with, DATA S002 data title (identical to that appearing in the job description) Title

1) daily variation.

Each month of daily variation data consists of,

Month Year (to be written on channel 0)

(Year - 1900) first day of month as day no. last day of month as day no.

D_1 D_2 D_3 ..... D_{24}

The last two lines are read in free format with

Di, i = 1, 2 .... 24 being the hourly means of daily variation.

The tape should be terminated by /
***Z

2) magnetometer data.

Each block of data should have the following layout.

Day no. Time of first observation. Base value of magnetometer in format (I3,2X,I4,5X,I6).

a_1 a_2 ..... a_6
a_7 a_8 .....
. .
. .
. .
a_{n-1} a_n >

where a_i are 3 digit numbers representing the observed magnetic field. A new block must be started each time that the day number and/or the base value changes and whenever there is a break in the sequence of observation. If only a few observations are missing zeros may be put in their place. The last block of the tape must be terminated by a /.

3) navigation data.

Each block of data consists of,

day number (format I3)

time degrees(lat.) mins.(lat.) degrees(long.) mins.(long.)

. . . . .
. . . . .
. . . . .
>

(format (I4,2(I4,I3)))

The terminator of the last block must be a /.

4) field coefficients.

This tape contains the coefficients of the spherical harmonic expansion of the geomagnetic field. Copies are held by R. B. Whitmarsh.

Output

Appears on 3 channels.

0 contains data titles of tapes read in, day number and time of magnetic blocks read and error messages.

1 contains the time, anomaly value, field value and latitude and longitude of each magnetic observation. The data are divided into the same blocks as they were read in.

6 contains a listing of the field coefficients read in.

Error messages

1) TOO MANY FIXES - occurs if more than 1000 fixes are input. Will stop reading fixes and continue execution.

2) BLOCKS OVERLAP - occurs if time of last reading of previous block of magnetic data is not less than time of first reading of next block. Warning only.

3) NAV DATA ALL BEFORE START TIME - calls EXIT.

4) NO DV DATA - occurs when no daily variation data can be found for period covered by a block of magnetic data. Calls EXIT.

5) NO DV TIME GREATER THAN M TIME - occurs when an hourly mean of daily variation cannot be found later than the time of a magnetic observation. Calls EXIT.

6) M TIME AFTER ALL FIXES - Calls EXIT.

7) FIXES MORE THAN 6 HRS. APART - occurs when fixes are more than 12 (sic) hours apart. Warning only.

Programmer

R. B. Whitmarsh.

A M E N D M E N T

N.I.O. PROGRAM 118

Title Gulf of Aden Magnetic Reduction

Page 1

Job Description The three lines following OUTPUT should read:

0 SEVEN HOLE PUNCH 1 BLOCK

1 SEVEN HOLE PUNCH N BLOCKS

6 SEVEN HOLE PUNCH 1 BLOCK

N.I.O. PROGRAM 119

Title	80/80 Card Reproduce
Name	-
Machine	1800
Language	Machine Code
Purpose	To copy cards without requirement for operating system (e.g. T.S.X.)
Operation	1. Clear Store 2. Program card and card deck to be copied in hopper 3. Press IPL button 4. Machine waits having read in cards. I register will be /0017 and S.A.R. will have /3000 in it 5. NPRO cards in reader and remove them 6. Put deck of blank cards in hopper 7. Press START 8. If more cards are to be copied, NPRO and remove spare blank cards, and go to step 1.
Restrictions	Program can only reproduce less than 200 cards per deck.
Method	Input deck is read into core storage and uses all available core. Core is then output to cards.
Speed	2 minutes per 100 cards.
Programmer	R. Merryweather (Pilkington), refer to J. Crease.

N.I.O. SUBPROGRAM -6/A

Title Find a file

Name Subroutine FILE2

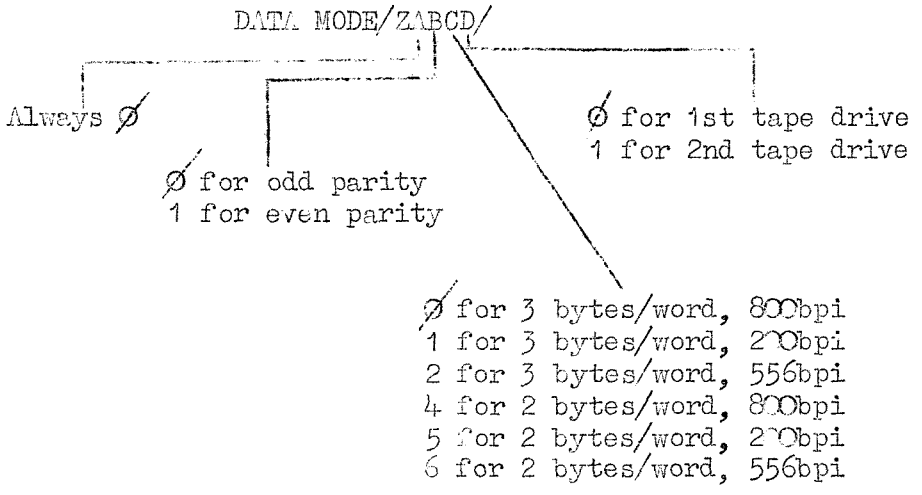
Machine IBM 1800

Language 1800 Fortran IV

Purpose The execution of CALL FILE2(N,MODE) in a Fortran program will position a magnetic tape in use so that a subsequence READ or WRITE statement will access the first record of file N.

The *IOCS record for the mainline program should contain 1443 PRINTER for error messages. It need not contain a *IOCS(MAGNETIC TAPE) record.

MODE should be set up in a DATA statement as follows:



For use with standard 1800 Fortran I/O, MODE should be set to Z0000 (or Z0001 for 2nd tape drive)

For use with BCD tapes, digit C in MODE should be set to 4, 5, or 6.

Errors Error messages are printed as follows:
'UNCORRECTABLE TAPE ERROR IN FILE NUMBER n'
and the subroutine returns control to the nonprocess supervisor.
'READ CHECKS HAVE OCCURED IN FILE NUMBER n'
and the subroutine continues.

Called subprograms MAGOP

Note This subroutine may be used in place of FILE0 (-6) in all applications. The equivalent statement to CALL FILE0(3,74) is CALL FILE2(3,MODE) where MODE=Z0000
This subroutine will also handle BCD tapes

Programmer B.J. Hinde

N.I.O. SUB-PROGRAM -15

Title	Tape Security System: Find a file and record
Name	Subroutine FINDO
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	An execution of CALL FINDO (N,M) will position the magnetic tape in use so that a READ or WRITE statement following will access record M of file N.
Restrictions	N ≤ 99 M ≤ 32767 Subroutine will only work with Fortran formatted I/O tapes written on an IBM 1800.
Notes	For an explanation of the N.I.O. Tape Security System see the publication DPG/P/7.
Programmer	B. J. Hinde

N.I.O. SUB-PROGRAM -16

Title	Tape Security System: Store current tape position.
Name	Subroutine STORO
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	An execution of CALL STORO (N,M) will store the next available record number M in file N on disk file MK. The mainline program must contain DEFINE FILE 201 (1, 2, U, IREC) and after the // XEQ card: *FILES(201, MK, D) where K is the number of the magnetic tape being used. The *IOCS record should contain MAGNETIC TAPE, DISK, 1443 PRINTER.
Output	'NEXT AVAILABLE RECORD IS M ON FILE N' on the 1443 printer.
Restrictions	$N \leq 99$ $M \leq 32767$ $K \leq 30$ and $K \neq 12$ on the present system.
Notes	For an explanation of the N.I.O. Tape Security System see the publication DPG/P/7.
Programmer	B. J. Hinde.

N.I.O. SUB-PROGRAM -17

Title	Tape Security System: Find next available file and record.
Name	Subroutine NEXTO
Machine	IBM 1800
Language	1800 Fortran IV
Purpose	An execution of CALL NEXTO (N,M) will position the magnetic tape in use so that a READ or WRITE statement following will access the next available file and record and place the number of the file and record in N and M respectively. The mainline program must contain DEFINE FILE 201 (1, 2, U, IREC) and after the // XEQ card *FILES (201, MK,) where K is the number of the magnetic tape being used The *IOCS record should contain MAGNETIC TAPE, DISK, 1443 PRINTER.
Output	'TAPE POSITIONED AT RECORD M ON FILE N' on the 1443 printer.
Restrictions	K ≤ 30, K ≠ 12 on the present system. Subroutine will only work with Fortran formatted I/O tapes written on an IBM 1800.
Notes	For an explanation of the N.I.O. Tape Security System see the publication DPG/P/7.
Programmer	B. J. Hinde

Data Processing Group

N.I.O. Tape Security System

In order that magnetic tapes may be shared between several users and that any user may have a quick and safe method of finding his way through a long magnetic tape (it is difficult to count more than 32767 records on the 1800) a suite of 3 programs and 4 subroutines have been written. They are as follows

N.I.O. 115	File initialisation	- INIT0
N.I.O. 116	Query files	- QUER0
N.I.O. 117	Pack files	- PACK0
N.I.O. -6	Find a file	- FILE0
N.I.O. -15	Find a file and record	- FIND0
N.I.O. -16	Store current tape position	- STOR0
N.I.O. -17	Find next available file and record	- NEXT0

The security system works only for formatted I/O tapes written in Fortran on the IBM 1800, although FILE0 may be used with tapes written by other systems. There is, set up on disk, a two word file for each magnetic tape owned by N.I.O. These files are named M1, M2 ..... M30 to correspond to the numbers of N.I.O. magnetic tapes. The first word of each disk file contains a magnetic file number (an integer ≤99) and the second a record number (an integer ≤32767). INIT0 is used to set up these values for a new magnetic tape to 1 and 1 respectively (or to any other values if desired). QUER0 can be used at any time to examine the contents of all the files.

If it is required to find the beginning of a particular file (N say) on a magnetic tape, a CALL FILE0 (N, L) where L is 74 for formatted I/O will result in the magnetic tape being positioned ready for a read or write operation to or from the first record on file N. If it is required, instead, to begin reading or writing to record M of file N then CALL FIND0 (N, M) should be used.

The subroutines STOR0 and NEXT0 require that the mainline program should contain

```
DEFINE FILE 201 (1, 2, U, IREC)
```

and after the // XEQ card there should be the files card:

```
*FILES (201, MK, O)
```

where K is the reel number of the magnetic tape to be used.

The program should then run as follows:

```
CALL NEXT0 (N, M)
```

- will position the tape at the next available record and file and place these numbers in M and N respectively. If the previous record written to tape was an ENDFILE then M will be equal to 1.

.
.
.
.
.
.

```
WRITE (5, 100) LIST
M=M+1          - for every record written to magnetic tape
                 increment M by 1.
                .
                .
                .
                .
                .
                .

ENDFILE 5
N=N+1
M=1            - for every ENDFILE statement, increment
                 N by 1 and reset M to 1.
                .
                .
                .
                .
                .
                .

CALL STORO (N, M)
               - at the end of the program this will
                 store on disk the next available file
                 and record number for future use.
```

If a facility is included in the program to abort a job (by testing a data switch for example) the exit routine should CALL STORO (N, M) so that a restart will continue the job in the correct place by a CALL NEXTO (N, M).

The program PACKO can be used to delete unwanted files and pack the remaining files on the tape to conserve space.

B. J. Hinde

September 1968