

Summer 1978

 Data General

ADDENDUM
to
Program Library
Catalog

USER'S  GROUP



This catalog does not replace the January 1978 Program Library Catalog. It is an addendum to the current issue.

The programs listed in this addendum will be available on magnetic tape only.

The price for the complete library, (Catalog Number 220-900) is now \$600.00



MULTIREG: Catalog Number 220-224

Source: FORTRAN IV, RDOS

Availability: \$10.00

General:

MULTIPLE REGRESSION (MULTIREG)

MULTIREG is a useful utility which fits a function of the form $b_1x_1 + b_2x_2 + \dots + b_nx_n = y$ by the method of least squares to a set of observations $(x_{1i}, x_{2i}, x_{3i}, \dots, x_{ni}, y_i)$ where $i=1, \dots, \text{IMAX}$. Currently, the maximum n is 10 and the maximum i is 50. MULTIREG can be used for polynomial and multivariable curve fitting simply by setting $x_m = x_1 ** m$ ($m = 2, 3, \dots$) or $x_3 = x_1 * x_2$, etc. MULTIREG is useful in determining the best available prediction equation for a chosen model. It is written in FORTRAN IV (Rev 5.2).

INPUT (from TTY) Filename of data file (maximum of 15 characters)

(from data file)

Line 1. # of variables (n), # of observations (i)
(uses free format)

Line 2. observation 1
 $x_{11}, x_{21}, \dots, x_{n1}, y_1$

Line 3. observation 2
 $x_{12}, x_{22}, \dots, x_{n2}, y_2$

Lines 4 to $i+1$ are subsequent observations

OUTPUT (to line printer device #12)

1. The set of observations
2. The resultant least-squares matrix
3. The determinant of the matrix
4. The solution vector ($b_k, k=1, \dots, n$)
5. The observed and calculated y_i 's, their difference, and its square.
6. The observed and calculated mean y .
7. The variance and standard deviation of the calculated y_i from the observed y_i .

BULCOW Catalog Number 220-225

Source: ALGOL

Machine

Requirements: RDOS with ALGOL, CRT

Availability: \$10.00

General:

This game is an advanced version of the very popular MASTER MIND. The idea is that you and the computer try to guess a five digit number thought up by the opposing player. A BULL is scored for each correct digit in the correct position and a COW for each correct digit out of position. The fascinating aspect of this program is the computer's ability to guess your number in a very small number of tries with seemingly unrelated guesses. The author would be very interested in hearing from a person who develops a strategy that consistently beats the computer.

BANNER Catalog Number 220-226

Source: ALGOL

Machine

Requirements: RDOS, with ALGOL, CRT and 120 column printer

Availability: \$10.00

General:

This program creates a large banner on a printer of any message you input. The type face is detailed Gothic in upper and lower case, numerals, and nineteen special characters. This is truly the most impressive character set you will ever see from a banner printer. Underlining and black for white inversion of the characters is possible. You'll want this program to create stunning banners for birthdays, parties, special events, and loved ones.

MAZE Catalog Number 220-227

Source: ALGOL

Machine

Requirements: RDOS, CRT and printer

Availability: \$10.00

General:

Remember all the fun you had as a youngster finding the path through a maze. This program will generate mazes of any size. Each maze is different because the selection of paths is done by a random number generator initiated by the clock. After struggling with some of the larger mazes, you will be fascinated by the algorithm which is used for path selection.

LIFE Catalog Number 220-228

Source: ALGOL

Machine

Requirements: RDOS, with ALGOL console CRT

Availability: \$10.00

General:

The Game of Life was originally described in *Scientific American*, October 1970, in an article by Martin Gardner. Organisms exist as characters occupying positions on the CRT screen and die or reproduce according to simple genetic rules. These rules make the behavior of the pattern unpredictable from generation to generation. Many hours of fascinating entertainment have been provided by watching the patterns progress to a steady state or annihilation.

LTYPER **Catalog Number 220-229**
Source: **ALGOL**
Machine
Requirements: **RDOS REV 6 and below (will not run with RDOS REV 5) 32K memory, 1 disk, 1 CRT terminal (6012, 6053), 1 printer if hardcopy is required.**
Availability: **\$10.00**

General:
LTYPER is a typewriter emulator that can be used to compose text easily and rapidly as on a normal typewriter. Once the text has been created, LTYPER can be used to make changes, add new material, and delete material from the text. The user can accomplish these functions without having to retype the unchanged portions of the text. With LTYPER, any combination of functions (composition, insertion, deletion, corrections) can be performed in one session at the keyboard.

Compared to conventional typewriter composition, LTYPER has several distinct advantages to offer the user in the preparation of text:

1. **SPEED**—allows rapid composition of text.
2. **STORAGE**—allows text to be saved in a convenient manner for future reference, use, modification, or printing.
3. **EDITING**—allows typing errors to be easily and quickly corrected.
4. **FORMAT**—allows text to be entered in almost any order at any point on the screen.
5. **MODIFICATION**—allows changes, insertions, and/or deletions to existing text.
6. **EASE OF USE**—LTYPER requires a minimum of training and operator experience.

ADVENTURE **Catalog Number 220-230**

Source: **FORTRAN IV**
Machine
Requirements: **Any AOS**
Availability: **\$15.00**

General:
Somewhere nearby is Colossal Cave, where others have found fortunes in treasure and gold, though it is rumored that some who enter are never seen again. Magic is said to work in the cave. I will be your eyes and hands. Direct me with commands of 1 or 2 words. I should warn you that I look at only the first five letters of each word, so you'll have to enter "northeast" as "NE" to distinguish it from "north". (Should you get stuck, type "help" for some general hints. For information on how to end your adventure, etc., type "info".)

Around you is a forest. A small stream flows out of the building and down a gully.

Load Command **RDOS LOAD/V @ MTA0:0 + /C**

TRACE **Catalog Number 220-231**
Source **Assembler**
Machine
Requirements: **ECLIPSE**
Program Size: **Requires 12K plus the size of the program being tested**
Availability: **\$10.00**

General:
TRACE is an interactive program developed for simulation of the execution of either instructions or micro-instructions. The user can at any point in the execution, view the pertinent attributes of the state of the computer. These attributes are the contents of selected memory locations, the contents of registers, and the values of binary variables such as input/output flags.

The program alleviates the increased burden which is put on the microprogrammer but requires a much deeper understanding of the architecture of the computer than would be necessary for assembly-language or machine language programming.

The main feature of TRACE that is not found in other simulators is that it is completely interactive. The simulation responds to and is activated by user commands at a console. This gives the user the power to stop at any point in the simulation to examine important parameters or to control the type and amount of output. This interactive control is an important link in the program development stage that is not available with most simulation. Another important feature of the simulator is that it runs on the machine it simulates. This allows programs to run directly or under simulation with a different switch setting on the operators console. Minor errors can be found and corrected very quickly without waiting for slow batch processing turn around times. This is due to the fact that the program under test and the simulator are both resident in the memory of the machine and the simulator has complete editing facilities.

MASH **Catalog Number 220-232**

Source: FORTRAN 5

Machine

Requirements: AOS

Availability: \$15.00

General:

AOS—FORTRAN 5 Package for large MATrices using SHared pages. The program MASH contains the following routines:

OPNM—opens the matrix file (for shared pages access) and defines the matrix structure. The matrix file may already exist and its contents are not cleared or intialized in any way.

CALL OPNM (mat-name filename, imax, nwpe)?
imax—max. Value of first subscript (array bound)
nwpe=number of words per matrix element

GETM/PUTN—access one element of the matrix.

CALL GETM (mat-name, value, i, j)

CALL PUTM (mat-name, value, i, j)

value = returned value (GETM), value to enter (PUTM)

i, j = first, second subscript

CONFM—(to be called before OPNM). Enables the reduction of the page length defined at assembly-time.

CALL CONFM (ndbpp)

ndbpp = number of disc blocks per page (must be a multiple of 4).

PUTMB—replaces the value of the matrix element last accessed. This avoids a second computation of the same offset. CALL PUTMB (mat-name, value)

Caution: MASH requires a special BINDing sequence (see source file).

FLECS **Catalog Number 220-233**

Source: ALGOL

Machine

Requirements: RDOS with ALGOL and DISC

Availability: \$10.00

General:

FLECS is a language extension of FORTRAN which has additional control mechanisms. These mechanisms make it easier to write FORTRAN by eliminating much of the clerical detail associated with constructing FORTRAN programs. FLECS is also easier to read and comprehend than FORTRAN.

Control Structures

IN FLECS, there is a uniform convention for writing control phrases and indicating their scopes.

For example:

<pre>IF (X. EQ. Y) U = V+W R = S+T FIN</pre>	<pre>DO (I = 1, N) A (I) = B (I) + C C = C* 2.14-3.14 FIN</pre>
--	---

Notice there is no need to invent statement numbers as the end of the scope is indicated by FIN. In the case of IF, the two statements are executed if the condition is true and no reverse logic is required (e.g. branch around if X. NE. Y). Similar structured statements are provided with keywords UNLESS, WHEN...ELSE, CONDITIONAL, SELECT, WHILE, UNTIL, REPEAT WHILE, and REPEAT UNTIL.

Internal Procedures

IN FLECS, a sequence of statements may be declared an internal procedure and given a name. The procedure may then be invoked from any point in the program by simply giving its name. This is extremely helpful in modularizing a program.

For example:

```
TO SUMMARIZE-FILE
  INITIALIZE-SUMMARY
  OPEN-FILE
  REPEAT UNTIL (EOF)
    ATTEMPT-TO-READ-RECORD
    WHEN (EOF) CLOSE-FILE
    ELSE UPDATE-SUMMARY
  FIN
  OUTPUT-SUMMARY
  FIN
```

Included Source

FLECS source statements may be included from the disk by using the INCLUDE statement. This will be particularly useful for inserting repetitious specifications in main programs and subroutines.

For example:

INCLUDE SPECS

SPECS is a FLECS file on disk.

For any programmer who appreciates structured programming (e.g. ALGOL, PL/I, etc.) but requires access to the features available through DG FORTRAN, FLECS is a must.

FOBOL.LB **Catalog Number 220-234**
Source: Assembly sources and documentation
 supplied
Machine
Requirements: NOVA or ECLIPSE, RDOS or DOS,
 FORTRAN IV (software floating point)
 Rev. 5.00
Availability: \$10.00

General:

FOBOL.LB is a library of FORTRAN-IV-callable sub-routines for string manipulation. From the users point of view it looks like variable-length string variables. He provides array space for the strings and a real (32-bit) variable for the description of each string or sub-string "variable" desired. Using FOBOL calls for the following types of functions can be performed:

SET UP DESCRIPTOR ("PIC")
CONCATENATE
MOVE
COMPARE
SUBSTRING
INDEX
LENGTH
STRING TO REAL CONVERSION
READ/WRITE LINE (.ROL/.WRL)
READ/WRITE SEQUENTIAL (.ROS/.WRS)

LTYPER-1 **Catalog Number 220-235**
Source: ALGOL
Machine
Requirements: RDOS REV .5 and below (will not run
 with RDOS REV .6) 32K memory, 1
 disk, 1 CRT terminal (6012, 6053), 1
 printer if hardcopy is required.
Availability: \$10.00

General:

LTYPER is a typewriter emulator that can be used to compose text easily and rapidly as on a normal typewriter. Once the text has been created, LTYPER can be used to make changes, add new material, and delete material from the text. The user can accomplish these functions without having to retype the unchanged portions of the text. With LTYPER, any combination of functions (composition, insertion, deletion, corrections) can be performed in one session at the keyboard.

Compared to conventional typewriter composition, LTYPER has several distinct advantages to offer in the preparation of text:

1. SPEED—allows rapid composition of text.
2. STORAGE—allows text to be saved in a convenient manner for future reference, use, modification, or printing.
3. EDITING—allows typing errors to be easily and quickly corrected.
4. FORMAT—allows text to be entered in almost any order at any point on the screen.
5. MODIFICATION—allows changes, insertions, and/or deletions to existing text.
6. EASE OF USE—LTYPER requires a minimum of training and operator experience.

STAKER **Catalog Number 220-238**Source **ASSEMBLER**

Machine

Requirements: memory size 6310 octal array NOVA
with hardware MUL/DIV RDOS,
magnetic tape drive FORTRAN IV run-
time library

Availability: \$20.00

General:

STAKER

This set of subroutine manages a run-time stack for assembly language programs running in NOVAS lacking a hardware stack. The stacks are configured so that subroutines can utilize them as they do in DGC FORTRAN IV using the provided linkage routines or as a fast push down stack. This allows assembly language subroutines to be written as re-entrant codes in a single or multi-task environment. The initialization routines use hardware integer MUL/DIV that could be easily modified. The package requires, 5 ZREL and 152 OCTAL NREL locations. It has been tested with RDOS Rev. 5.00.

MTBIO

This is a set of FORTRAN IV callable subroutines to perform block transfers of data between magnetic tape and the caller's program. It is written in assembly language but uses all the normal FORTRAN IV linkage. The transfers are done via RDOS free format I/O calls so that the records can be of arbitrary length and no double buffering is done. The provided routines to open and close the tape files allow for the option of labeling the tape files. These labels contain information about the file such as its name, record length and creation date. The labels also provide a means of controlling access to the file and thus prevent accidental over-write of data. The package uses the linkage routines. FARL and FRET in the DG FORTRAN IV library and requires 1021 OCTAL NREL locations most of which can be overlaid. MTBIO also requires hardware MUL/DIV.

MTCARD

MTCARD is an assembly language program to perform transfers of source code between RDOS formatted files and card image formatted files on magnetic tape. The card image tape files can be in ASCII or EBCDIC and they can have record lengths of 80 or 132 characters. The program is called into execution from the CLI as a normal utility. Global and local switches control the options. MTCARD requires hardware integer MUL/DIV and 6310 octal memory locations. It uses the support routines in the packages Staker, Stringer and MTBIOA.

STRINGER

Stringer is a set of subroutines which provides various string handling routines useful in text input and output. There are entry points to load and store a byte, to convert a binary number to a string of octal or decimal ASCII characters, and to convert a string of octal or decimal ASCII characters into a binary number. The routines are written in assembly language and are intended to be used in assembly language programs. All the entry points are re-entrant and require the linkage routines provided in the package STAKER. Stringer requires hardware MUL/DIV and uses 6 ZREL and 252 OCTAL NREL locations.

BUFMAN

Bufman is a set of routines to manage data buffers in assembly language programs. The buffers can be of arbitrary number and size. The bookkeeping is maintained in descriptor blocks which are linked together in two queues. The routines are all re-entrant. They are intended to allow a group of interrupt handlers on data accumulation tasks to fill buffers and then pass them on to another task for analysis or storage. Bufman uses the linkage routines provided in the package STAKER. It uses 13 OCTAL ZREL locations and 142 OCTAL NREL locations. It has been tested with RDOS REV 5.00 but should run with other revisions as well.

MTBIOA

This is a set of assembly language subroutines to perform block transfers of data between magnetic tape and the callers program. They are intended to be called from assembly language programs. The transfers are done via RDOS free format I/O calls so that the records can be of arbitrary length and no double buffering is done. The provided routines to open and close the tape files allow the option of labeling the tape files. These labels contain information about the file such as its name, record length and creation date. The labels also provide a means of controlling access to the file and thus prevent accidental over-write of data. The package requires 1077 OCTAL NRSL memory locations most of which can be overlaid. MTBIO also requires hardware integer MUL/DIV.

SICOS **Catalog Number 220-239**

Source: FORTRAN IV

Machine

Requirements: NOVA 2/10

Availability: \$10.00

General:

SICOS (Simulation of Continuous System) is a simulation program for continuous systems. The basic idea of the system is how far a comfortable simulation program could be realized on a minicomputer. These continuous systems could be mechanical or electrical as therodynamical systems, which can be described by common differential and algebraic equations with sufficient accuracy. SICOS requires three classes of information: a control instruction, a description of the model to be simulated (configuration) and certain values as parameters.

Related to these classes SICOS provides for three statement types:

Control instruction:

They affect the simulation and determine the information input (via console/card reader/paper tape reader) respectively the simulation output (via console/line printer/plotter).

Configuration statements:

They define the mathematics of the model, divided into several block-functions.

Parameter statements:

They provide each block with numeric constants or with the starting conditions.

Block types:

SICOS offers 26 different block types which are defined by the user with symbolic characters. The program is divided into three different components: the translator, the function library and the simulator.

EUCLIDS **Catalog Number 220-240**

Source: TS BASIC

Machine

Requirements: RDOS

Availability: \$10.00

General:

Euclid's Algorithm. The program illustrates Euclid's algorithm for finding the greatest common divisor of two positive integers by successive subtractions and comparisons.

PSGRAM **Catalog Number 220-241**

Source: BASIC

Machine

Requirements: RDOS

Availability: \$10.00

General:

Phrase Structure Grammar. The program deals with non-recursive phrase structure grammar. The computer will generate some random sentence structures. The user determines how many strings he wants and is then required to develop the phrase structure trees that produce the strings.

PALAUNG **Catalog Number 220-242**

Source: BASIC

Machine

Requirements: RDOS

Availability: \$10.00

General:

Palaung Pronouns. The program presents the pronouns of Palaung as a three dimensional model. The dimensions are number, inclusion or exclusion of the speaker, inclusion or exclusion of the person spoken to. The operator attempts to guess the appropriate pronouns for dual number and then identifies the ending for dual. Number of guesses etc. are tabulated and printed out after presenting the four dual pronouns.

HAIKU **Catalog Number 220-243**

Source: BASIC

Machine

Requirements: RDOS

Availability: \$10.00

General:

Haiku Generator. Haiku Generator combines strings of nouns, verbs, adjectives, and adverbs into 3 lines of 5 syllables, 7 syllables, and 5 syllables. Two inputs are required: (1) a random number generator seed and (2) the order for occurrence of the words. Words are selected at random from lists stored in the program. Syllables are counted by identifying a, e, i, o, u, and terminal y. Terminal e does not count as a syllable. Individual lexical items must be 26 characters or less but can be phrases or clauses as well as individual words.

**FOURIER SERIES
ANALYSIS (FFT) Catalog Number 220-244**

Source: BASIC
Machine
Requirements: NOVA, TTY, H.S. Reader/Punch
Line printer
Availability: \$10.00

General:
This program is designed to calculate the Fourier coefficients to the Mth harmonic given one cycle of N equally spaced data points.

$$a(0) + \sum_{p=1}^M \{ a(p) \cos(2\pi np/N) + b(p) \sin(2\pi np/N) \}$$

where n is one of N points
and the (Nth)th point would be the
same as the zeroth point.

The program request an even number of points from 4 to 360 and accepts data from either paper tape or keyboard. The coefficients are printed on the terminal and may be also printed on the line printer along with the raw data and series summation for each point allowing for curve fit comparison.

Algorithm

The algorithm is a recursive technique modified slightly from that described in **Mathematical Methods for Digital Computers**, Volume 1, Chapter 24, pp 258-262. The first change eliminates the use of the (N+1)th point which should equal the zeroth point and the second change requires trigonometric look up for each harmonic. This second change has two advantages. First, it eliminates cumulative error in the trig calculation by going directly to the sin/cos routines each time. Second, it allows the routine to calculate each harmonic independently of all others. For example the routine could easily be modified to calculate the 5th harmonic without ever calculating the 1st, 2nd, 3rd or 4th.

LETTERS Catalog Number 220-245

Source: BASIC
Machine
Requirements: DOS or RDOS, LPT or TTO
Availability: \$10.00

General:

LETTERS/LETTERS.DA

Written in BASIC, "LETTERS" produces full page size letters composed of either a user specified text or by the letter that the large letter represents. Currently available characters are A-Z, 0-9, and some punctuation (?:'"+). Full instructions are given for creating new characters e.g. lower case. About 5kb of BASIC user space is required, and a line printer or TTY.

BCILW Catalog Number 220-246

Source: Assembler
Machine
Requirements: magnetic tape device
Availability: \$10.00

General:

BCILW writes core images to magnetic tape. This program creates a save file which, when booted from magnetic tape, loads both the 'binary loader' and the 'mag. tape core image loader/writer' into computer memory. These two loaders can be positioned anywhere in memory.

The CILW is 'core efficient' in that it will load core up to and including the location preceding itself. The bin loader has been modified to allow loading of data above itself (the standard version doesn't allow this).

TREAD Catalog Number 220-247

Source: Assembly
Machine
Requirements: paper tape reader or teletype reader
Disk
Availability: \$10.00

General:

TREAD.SV ABSTRACT:

Tread performs the opposite functions of "PUNCH" and "BPUNCH" reading paper tape from \$PTR (easily changed for \$TTR). The program offers:

1. Optional Parity check on ASCII transfer (Supressed by default)
2. Binary or ASCII transfer
3. Arrange destination file randomly or sequentially
4. Append to existing file if any

Tread is written in assembly language for the NOVA/ECLIPSE, DOS/RDOS.

CARDTRANS Catalog Number 220-248

Source: FORTRAN
Machine
Requirements: RDOS Rev 5.0, Console and disk
Availability: \$10.00

General:

CARDTRANS is a FORTRAN utility program which may be used to transform and compact card image files by removal of sequence numbers in column 73-80 and by extraction of trailing blanks from text in columns 1 to 72. Carriage returns are inserted after each logical record. CARDTRANS may be used in conjunction with TAPE-COPY to input fixed and blocked card image files from 1/2 inch magnetic tape (written by a main-frame computer) to the NOVA or ECLIPSE. It may also be used to chop off trailing blanks and sequence numbers from RDOS files produced by reading EBCDIC coded cards with XFER/A command.

TAPECOPY **Catalog Number 220-249**
Source: **FORTRAN**
Machine
Requirements: **RDOS Rev. 5.0, disk, Magnetic tape
drive, console**
Availability: **\$10.00**

General:

TAPECOPY is a FORTRAN utility program which will read most 1/2 inch magnetic tapes which are written to one of the industry standard exchange formats and will write that information to any RDOS file or device. The program reads the tape and writes its output file in binary mode, permitting code independent transfer of data. Optionally, the input data bytes can be transformed from any external character set to any other desired character set. (The program is supplied with a control file for translation of EBCDIC code characters to ASCII code characters. This allows immediate use for reading of tapes written on IBM systems.)

**BASIC
INTERPRETER
SYSTEM** **Catalog Number 220-250**

Source: **Assembler**
Machine
Requirements: **6K minimum for one user, CALL
statement from BASIC**
Availability: **\$20.00**

General:

This is a sample data acquisition program to be used with Single User Basic. The program contains 7 subroutines.

1. ADGET—reads the A/D channel specifier and returns the value read.
2. DAC—outputs the value of the D-A channel specified
3. DIGIN—reads the A-register of the general purpose interface and returns the value of the bit requested.
4. DIGOT—outputs the value of the bit # of the A-register of the general purpose interface.
5. CLKST—starts the real time clock with the interval desired, 0-3.
6. CLOCK—returns the number of the clock ticks since the set command was given.
7. An interrupt service routine for the clock.

PAD **Catalog Number 220-251**
Source: **BASIC**
Machine
Requirements: **RDOS**
Availability: **\$10.00**

General:

PAD is a program to calculate the register values required for the T-PAD, PI-PAD or minimum-loss matching PAD.

LEADER **Catalog Number 220-252**
Source: **FORTRAN IV, Version 5.21**
Machine
Requirements: **RDOS**
Availability: **\$10.00**

General:

Leader produces a human readable punch pattern on paper tape corresponding to an input from the console. Input characters may be alpha (a-z), Numeric (0-9), or special purpose !, ", \$, %, &, ', (,), *, +, ,, -, ., /, :, ;, <, =, >, ?, @, [,], ^, -

BLACKJACK **Catalog Number 220-253**
Source: **FORTRAN 5**
Machine
Requirements: **RDOS**
Availability: **\$10.00**

General:

This program simulates the game of Blackjack. The machine is the dealer. Player starts with a purse of \$100, maximum bet is \$20 and minimum bet is \$1.00.

TEKPACK **Catalog Number 220-255**

Source: Assembler subroutines for Extended BASIC

Machine
Requirements: Any system capable of supporting Extended BASIC with, 32K memory running RDOS Rev. 6

Availability: \$25.00

Peripherals: Tektronix 401x graphics display

General:

Tekpack is a small set of assembler subroutines for Extended BASIC organized as an SBRTB for revision 4.3 which implements a simple set of graphic primitives for the Tektronix T-401x series of graphic displays. The subroutines have been in use for several years in single user, multiuser, standalone, and disk based versions of BASIC. The routines implement ERASE, MOVE, DRAW, and graphic input screen coordinate primitives.

DRAFTSYS

Source: DGC Extended BASIC

Subroutine
Required: TEKPACK

General:

DRAFTSYS is a program written in DGC Extended BASIC which provides the user with a simple computer-aided sketchpad. DRAFTSYS enables the user to create drawings composed of vectors, alphanumerics, and up to 20 user definable symbols. Drawings may be saved on disk for later display or editing, and are composed of from one to several logical parts called pages. DRAFTSYS requires DGC Extended BASIC with graphic primitives supplied by TEKPACK and a tektronix T-401x graphic terminal.

MOONLANDER

Source: Extended BASIC

Subroutine
Required: TEKPACK

Peripherals: Tektronix 401x graphics display

General:

MOONLANDER is a BASIC game which makes use of the routines in TEKPACK to simulate a lunar lander. The user has control over vehicle thrust and 2-D rotation via the graphic input device on a Tektronix T-401x graphic terminal. The display shows the terrain, fuel consumption, the vehicle position, and thrust magnitude.

ATCS

Source: ALGOL and ASM

Peripherals: Tektronix 401x graphics display

General:

ATCS (Algol Terminal Control System) is a comprehensive set of graphic primitives written in DGC ALGOL for the Tektronix T-401x series of graphic terminals. ATCS allows the user to draw in either "page" (logical) or "screen" (absolute) coordinates by simply calling EXTERNAL ALGOL procedures. ATCS automatically clips vectors which lie outside the viewing area, and provides a graphic input routine for reading the position of the graphic cursor. A DGC minicomputer capable of supporting ALGOL and a Tektronix T-401x graphics display are required. Does not support T-4014 Enhanced Graphic Module, 1023 x 1023 resolution only.

TANK

Source: FORTRAN IV

Peripherals: Tektronix T-4014 graphics display

General:

TANK is a DGC Multitasking Fortran IV program which simulates in real-time the motion and fire of two user controlled tank-like images on a Tektronix T-4014 graphics display. TANK makes use of the storage and refresh capabilities of the T-4014. Users maneuver and fire their tank via single keystroke commands from the T-4014 terminal. A 32K NOVA or Eclipse with RDOS and a T-4014 configured as the master console is required.

**PLAN and
FILCTL**

Catalog Number 220-256

Source: FORTRAN IV

Machine

Requirements: D.G. Computer with 32K words
Running RDOS

Availability: \$15.00

General:

PLAN is a modeling system designed to aid the business planner by performing the row matrix manipulation calculations needed in spread sheet preparation. PLAN enables the user to work with the system in an interactive mode. The PLAN system is designed for users with no programming experience and requires only a knowledge of the PLAN User's Manual.

The user issues commands which are used to create a set of simple English-language type instructions. These instructions dictate how numeric information will be computed or obtained as well as specifying the format in which reports will be printing. A Profit and Loss Statement is an example of a typical report.

The PLAN system provides the means to specify a wide range of planning applications and to generate reports. All specifications for a particular planning application, including all instructions for calculations, printing and the resulting data values, are kept in a user's data base which is maintained on a disk file. This data base is filled with specifications and data values which are manipulated by user commands.

The user data base is created by an interactive routine FILCTL. FILCTL consists of a set of commands to create, analyze, extend or reorganize a user data base.

PLAN and FILCTL are written in FORTRAN IV and run on any Data General Computer with 32K words or more under RDOS.

**VIRTIAL ARRAY
PROGRAM (AOS**

FORTRAN 5) Catalog Number 220-257

Source: FORTRAN 5

Machine

Requirements: AOS

Availability: \$10.00

General:

Allows user to have more or larger arrays than will fit in his address space (uses shared page files to extend address space).

Supplied are the assembler programs to effect the system. Fortran 5 set up and test programs, Macros and instructions for use. Note that it is not user-transparent. In particular two different names must be used for the same array when found to the left or to the right of "=" in an assignment statement. All sources are provided so users can modify and update for any changes in the language and operating system.

Currently it supports real and integral vartapies. But an assembler language programmer could change it easily to support other data types, or customize it to run faster for specific requirements.

There are several restrictions on use. Arrays can be passed to subroutines but the subroutine must know it is getting a virtual array, and the dimensions of that array. The naming issue is mentioned above. Virtual array accessing will be much (10-100 times) slower than normal arrays. Individual arrays must be no greater than 32565 ELEMENTS APTFCF and can only have one dimension (Two or more dimensions can be obtained with slightly more complexity in the users array definition). Arrays can be equivalenced. Used as strings rather than elements (F.G. sequential record and block I/O) will not work in general.

0

0

0

