HEWLETT PACKARD

DATA SYSTEMS • 11000 Wolfe Road, Cupertino, California 95014, Telephone 408-257-7000, TWX 910-338-0221

Dear User:

Attached is the newest update to your HP BASIC Program Library Handbooks. This single package contains both the HP BASIC Addenda and the Educational Users Group Addenda. The order number for this package is HP 36000-90673.

CONTENTS:

- HP BASIC Program Library Handbook
  a) New Introduction
  b) New Complete Indices
  c) New and Modified Documentation, RUNs, and LISTings
  d) Note: LETTER (HP 36124A has been replaced by a new program, PRINT (HP 36299A). LETTER should be removed from your Handbook.

- Volume II Huntington Computer Project
  a) New Introduction
  b) New Complete Indices
  c) New Documentation, RUNs, and LISTings

- Volume III Application Programs
  a) New Introduction
  b) New Complete Indices
  c) New Documentation, RUNs, and LISTings

- Volume IV Stanford Graduate School of Business Handbook
  a) New Introduction
  b) New Complete Indices

This material should be merged with your original Handbooks to comprise single volumes.

HP SOFTWARE CENTER
Users' Library

36000-90673 June 1973

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

HP BASIC PROGRAM LIBRARY HANDBOOK

HEWLETT-PACKARD, SOFTWARE CENTER, 11000 Wolfe Road, Cupertino, California 95014 Area Code 408 257-7000 TWX 910-338-0221

36000-90001	6/72	Handbook
36000-90002	10/72	Addenda
36000-90003	2/73	Addenda
36000-90673	6/73	Addenda

Printed in U.S.A.

CLASSIFICATION CODE CATEGORY

(Not all categories have programs. Please refer to Classification Code Listing for available programs in HP BASIC)

100 DATA HANDLING

101 EDITING
102 INFORMATION STORAGE AND RETRIEVAL
103 TABLE HANDLING
104 CHARACTER/SYMBOL MANIPULATION
105 CODE/RADIX CONVERSION
106 DUPLICATION
107 SORTING AND MERGING
108 DATA HANDLING UTILITIES
109 MEDIA CONVERSION
110 FILE MANAGEMENT
112 SPECIAL FORMAT DATA TRANSFER

200 TESTING, DEBUGGING AND PROGRAMMING AIDS

201 TRACING
202 INSTRUMENT TEST
203 DISC/DRUM EQUIPMENT TEST
204 MAGNETIC TAPE EQUIPMENT TEST
205 GRAPHIC EQUIPMENT TEST
206 MEMORY SEARCH AND DISPLAY
207 DUMPING
208 CORE STORAGE TEST
209 CENTRAL PROCESSING UNIT TEST
210 BREAK POINTS
211 DEBUGGING AIDS
212 PROGRAMMING AIDS
213 PAPER TAPE EQUIPMENT TEST
214 PUNCH CARD EQUIPMENT TEST
215 PRINTER EQUIPMENT TEST
216 A/D - D/A EQUIPMENT TEST
217 TELECOMMUNICATIONS EQUIPMENT TEST
218 SPECIAL DEVICE EQUIPMENT TEST
219 DATA ACQUISITION SYSTEMS TEST

300 MATH AND NUMERICAL ANALYSIS

301 MATHEMATICS, GENERAL
302 EXTENDED-PRECISION ARITHMETIC
303 COMPLEX ARITHMETIC
304 BCD/ASCII ARITHMETIC
305 BOOLEAN ALGEBRA
306 FUNCTIONS, COMPUTATION OF
307 INTERPOLATION/EXTRAPOLATION
309 CURVE FITTING
310 NUMERICAL INTEGRATION
311 POLYNOMIALS AND POLYNOMIAL EQUATIONS
312 MATRIX OPERATIONS
313 EIGENVALUES AND EIGENVECTORS
314 SYSTEMS OF LINEAR EQUATIONS
315 SYSTEMS OF NON-LINEAR EQUATIONS
316 INTEGRAL TRANSFORMS
317 NUMERICAL DIFFERENTIATION
318 ORDINARY DIFFERENTIAL EQUATIONS
319 PARTIAL DIFFERENTIAL EQUATIONS

400 PROBABILITY AND STATISTICS

401 UNIVARIATE AND MULTIVARIATE PARAMETRIC STATISTICS
402 TIME SERIES ANALYSIS
403 DISCRIMINANT ANALYSIS
404 REGRESSION ANALYSIS
405 RANDOM NUMBER GENERATORS
406 PROBABILITY DISTRIBUTION SAMPLING
407 NON-PARAMETRIC STATISTICS
408 STATISTICS, GENERAL
409 CORRELATION ANALYSIS
410 ANALYSIS OF VARIANCE AND COVARIANCE
411 FACTOR ANALYSIS
412 SCALING
413 GENERAL PROBABILITY

500 SCIENTIFIC AND ENGINEERING APPLICATIONS

501 SOCIAL AND BEHAVIORAL SCIENCES
502 GEOPHYSICS
503 GEOLOGY
504 OCEANOGRAPHY
505 PHYSICS
506 MEDICAL SCIENCES
507 CHEMISTRY
508 BIOLOGY
509 ASTRONOMY AND CELESTIAL NAVIGATION
510 PETROLEUM ENGINEERING
511 HYDRAULIC ENGINEERING
512 NUCLEAR ENGINEERING
513 ELECTRICAL ENGINEERING
514 MECHANICAL ENGINEERING
515 CIVIL ENGINEERING
516 CHEMICAL ENGINEERING
517 AERONAUTICAL ENGINEERING
518 STRUCTURAL ENGINEERING
519 SYSTEM THEORY

600 MANAGEMENT SCIENCES AND OPERATIONS RESEARCH

602 PERT
603 CRITICAL PATH ANALYSIS
604 OPTIMIZATION PROGRAMS
605 LINEAR PROGRAMMING
606 DISCRETE SYSTEMS SIMULATION
607 CONTINUOUS SYSTEMS SIMULATION
608 FORECASTING TECHNIQUES
610 DYNAMIC PROGRAMMING

700 BUSINESS AND MANUFACTURING APPLICATIONS

701 JOB REPORTING
702 QUALITY ASSURANCE PERFORMANCE ANALYSIS
703 QUALITY ASSURANCE TESTING
704 NUMERICAL CONTROL
705 BILL OF MATERIALS
706 PAYROLL ACCOUNTING
707 WORK-IN-PROCESS CONTROL
708 INVENTORY ANALYSIS
709 ACCOUNTS PAYABLE
710 SALES FORECASTING
711 ACCOUNTS RECEIVABLE
712 FINANCIAL ANALYSIS
713 INVESTMENT ANALYSIS
714 ECONOMIC ANALYSIS
716 BUDGETING PROGRAMS
717 BUSINESS INFORMATION SYSTEMS
718 BUSINESS SERVICES
720 EDUCATIONAL ADMINISTRATION

800 EDUCATION

801 MATHEMATICS (EDUCATION)
810 PROGRAMMING AND COMPUTER SCIENCE (EDUCATION)
820 ENGINEERING (EDUCATION)
830 ECONOMICS (EDUCATION)
833 SCIENCE (EDUCATION)
850 FINE ARTS (EDUCATION)
860 SOCIAL SCIENCE (EDUCATION)
863 HISTORY (EDUCATION)
870 ENGLISH (EDUCATION)
871 FOREIGN LANGUAGES (EDUCATION)
872 READING (EDUCATION)
880 BUSINESS (EDUCATION)
890 VOCATIONAL (EDUCATION)

900 UNCLASSIFIED

901 DEMONSTRATIONS
903 GAMES
904 PLOTTING ROUTINES
905 MICROCODE

INTRODUCTION

What is Hewlett-Packard?

From a two-man beginning some thirty years ago, Hewlett-Packard has developed into a major designer and manufacturer of electronics for measurement, analysis, and computation. HP customers in science, industry, medicine, and education know and appreciate Hewlett-Packard's reputation for technical excellence, quality, and reliability.

Over 100 world-wide offices sell and service the products of sixteen manufacturing facilities located in the United States, Europe, and the Far East. Specialized customer support (such as training, systems analysis and equipment rental) is available at domestic regional Data Centers in New Jersey, Georgia, Illinois, and California and international data centers in Italy, England, Germany, France, Canada, and Australia.

What is the HP BASIC Library?

One important facet of customer support is the collection and distribution of contributed software, from and for HP computer users. The HP BASIC Library contains contributed programs covering a variety of subjects, all written in the HP BASIC language. You can use these programs directly or as a starting point for developing your own special-purpose software. Many of the programs run on a "single terminal" BASIC system; however, some use sophisticated features available only to time-share system users.

How is the HP BASIC Library Organized?

Hewlett-Packard has categorized all programs within the following major areas of user interest:

1 – DATA HANDLING
2 – TESTING, DEBUGGING AND PROGRAMMING AIDS
3 – MATH AND NUMERICAL ANALYSIS
4 – PROBABILITY AND STATISTICS
5 – SCIENTIFIC AND ENGINEERING APPLICATIONS
6 – MANAGEMENT SCIENCES AND OPERATIONS RESEARCH
7 – BUSINESS AND MANUFACTURING APPLICATIONS
8 – EDUCATION
9 – UNCLASSIFIED

The four indices appearing at the beginning of the Handbook contain **all** BASIC programs available through the HP BASIC Program Library. Four separate handbooks are now available. The indices indicate in which of the four volumes the programs appear.

T = HP BASIC
E = EDUCATION
H = HUNTINGTON
S = STANFORD

The programs which appear in this Handbook are designated in the indices by a "T" indicating general programs in HP BASIC.

Program indices are by:

- Classification Code (In the Handbooks, programs are sequenced by classification code)
- Cross-reference Index
- Program Name (In alphabetical order)
- Order Number

Who Prepares These Programs?

Hewlett-Packard **supported programs** written in HP BASIC are now listed in the Educational Users Group Handbook, Volume III, Applications Programs, and the HP BASIC Handbook or general programs. They are distinguished by a stripe. **Contributed programs** in HP BASIC which have been submitted by users of HP computer systems throughout the world are listed in the HP BASIC Handbook, The Educational Users Group Handbooks, including the Stanford Graduate School of Business Handbook. The Educational Handbooks contain programs which are oriented for instructional purposes.

Contributions to our library are from many disciplines and various applications. HP personnel at the Software Center run and check the documentation for each program, but because it is impractical to test these programs under all possible conditions, Hewlett-Packard cannot assume responsibility for errors. Therefore, please report any errors you might find to the Software Center for forwarding to the program contributors.

How Can I Make Use of These Programs?

This handbook contains operating instructions, a sample RUN and Listings for each program. If the time-share system your terminal accesses has these programs in its system library, you merely GET the program preceding the name by a "$." Otherwise, you may enter the

program from your terminal by duplicating the information in the listing.

The system on which the program was tested by Software Center personnel is listed under "Systems Specifications" in the Handbooks, and it is indicated in the Indices as the leading alpha of the classification code. Its versatility to RUN on other time-sharing systems can be determined from its program features. The following chart was designed to be used in conjunction with the program listings from the BASIC Handbooks.

Program Features	2000A	2000B	2000C	2000E	2000C High-Speed 2000F
Maximum Program Size	5100 Words	5100 Words	10000 Words	4180 Words	10000 Words
Maximum Number of Files	8	8	16	4	16
Maximum Number of Records/File	128	128	32767	48	32767
Maximum Number or Words/Record	64	64	256	128	256
Programmable Functions:					
TIME		x	x	x	x
ENTER		x	x		x
COMMON		x	x	x	x
CHAIN		Chain-"Name"	Chain-$Name, Statement No.	Chain-$Name	Chain-$Name, Statement No.
		x	x	x	x
PRINT USING (IMAGE)			x		x
BRK					x
ASSIGN			x		x
RESTARTABLE RND			x	x	x
SPACE			x		x
LINE			x		x

How Can I Order These Programs?

Contact your local HP sales office to order any of the material from the library, (except HP subscriptions).

To order magnetic tapes of all contributed BASIC software, contact your local HP sales office and ask for:

HP Part No. 02000-90029 2000C Mag Tape $25.00
HP Part No. 02000-90060 2000C'/F Mag Tape 25.00
HP Part No. 02000-90059 2000E Disc Cartridge 150.00*

* Price includes disc. If you return your user cartridge with your order, the price is $25.

Individual Programs On Paper Tape and Handbooks

Programs include punched paper tape and documented user procedures. Price, $10 per program or otherwise noted on program documentaion to cover the cost of reproduction and handling. **Licensed HP-supported programs are available only with the purchase of an HP time-share system.**

Note: The June 1973 Addenda to the HP BASIC Library Handbooks is a single package, including both the HP BASIC Handbook Addenda and the Educational Users Group Addenda. The order number assigned to this addenda (HP 36000-90673) reflects the issue date in the last 3 digits.

To order individual Handbooks from your local sales office, use the following:

Title	Date	Part Order No.	Price
HP BASIC Program Library Handbook	June '72	36000-90001	$10
HP BASIC Program Library Handbook Addenda	Oct. '72	36000-90002	$ 3
HP BASIC Program Library Handbook Addenda	Feb. '73	36000-90003	$ 3
Educational Users Group Program Library—Vol. II Huntington Computer Project	Feb. '72	5951-3041	$ 5
Educational Users Group Program Library—Vol. III Application Programs	Feb. '72	5951-3042	$ 5
Educational Users Group Program Library—Vol. IV Stanford Graduata School of Business Handbook	Oct. '72	36500-90001	$ 5
Educational Users Group Program Library—Addenda	June '72	36400-90002	$ 3
Educational User Group Program Library—Addenda	Oct. '72	36400-90003	$ 3
Educational Users Group Program Library—Addenda	Feb. '73	36400-90004	$ 3
HP BASIC & Educational Users Group Program Library Addenda	June '73	36000-90673	$ 6

Note: Educational Users Group Program Library Volume I, Guide to Educational Support (HP 5951-3040) is obsolete and has been discontinued.

OTHER LIBRARY PUBLICATIONS

The following is a list of documentation not included in the Library Handbooks. They should be ordered to fully document the programs referenced.

Title	Date	Order No.	Price
FINDIT Users Manual	Feb. '73	36250A, Option D00	$ 5
TIDE Text Editor	Oct. '71	36200 Option D00	$ 5
CTC1 Documentation	Feb. '73	36210B	$ 5
CTC2 Documentation	Feb. '73	36211B	$ 5
CTC3 Documentation	June '73	36212B	$ 5
CTC4 Documentation	June '73	36213B	$ 5
CTC5 Documentation	Feb. '73	36214B	$ 5
CTC6 Documentation	June '73	36638A	$ 5

How to Update These Volumes?

This handbook was produced, three-hole punched, and stapled so that users can unbind them and keep them in a loose-leaf notebook.

A 2" capacity binder is available from Hewlett-Packard. Order HP Part No. 9282-0069 (to match your other HP System manuals.)

Updates will be issued once every four months. The updates will contain revisions and additional programs. The updates should be merged with the original to produce single volumes.

HP BASIC SUBSCRIPTION SERVICES

(available only to users in the North American countries)

PAPER TAPE SUBSCRIPTIONS

To subscribe, mail Hewlett-Packard a check for $250 (plus applicable state and local taxes). When we receive a paper tape subscription, we send the subscriber a booklet containing 50 coupons. Discounted punched paper tapes may then be procured via coupon or coupons as follows: (Check the documentation—if no price appears, the program is $10.)

NORMAL PROGRAM PRICE	COUPONS REQUIRED
$10 to $40	1
50 to 90	2
100 to 140	3
150 to 190	4
200 to 240	5

MAGNETIC TAPE SUBSCRIPTIONS

A "two-year" magnetic tape subscription service of contributed BASIC programs is available to 2000C System owners for $100 (plus applicable state and local taxes). When we receive a magnetic tape subscription, we send the subscriber a current magnetic tape plus a coupon book for five additional magnetic tapes (released periodically). The coupon book consists of five self-addressed prepaid mailing cards. The subscriber simply mails a card each time a new addendum is released. Each new index announces a new addendum.

HP BASIC HANDBOOK SUBSCRIPTIONS

A "two-year" handbook subscription service for a choice of any three of the handbooks is available to HP users for $25 (plus applicable state and local taxes). When we receive a handbook subscription, we send the subscriber the current set of the three volumes he has requested, plus a coupon book to handle future addenda when they are announced in the Index. The user will receive addenda released between June 1972 and June 1974.

To order any of the above subscriptions, make your check payable to Hewlett-Packard Company and mail to:

Hewlett-Packard Company
Software Distribution
11000 Wolfe Road
Cupertino, California 95014

When Will New Programs be Added?

New programs are continually added to the library. A new "INDEX TO HP BASIC PROGRAM LIBRARY" is published every February, June, and October. The Index announces the release of new addenda, lists all the programs to date in the BASIC Library (by cross reference words); and flags new, revised, and HP-supported programs. Using the Index (5952-4639) you can learn about new additions, contribute your own programs and report any software bugs. Copies of the Index are complimentary and may be obtained from your local HP Sales and Service office.

How Do I Interpret the Listings?

The ASCII characters "Control N" and "Control O" when used in a PRINT statement, generate linefeed and carriage return on the TTY terminal during a RUN. During a LIST these characters are not printed by a TTY. On HP 2778 and the HP 2610 line printer listing, these characters are printed as "N" and "O" respectively. Listings in some Handbooks were prepared on these line printers.

What Related Information is Available?

There are a number of manuals and documents relating to HP 2000 series Time-Sharing Systems that may be of value to users of the Handbook. These include:

Language Manuals:

2000C: A GUIDE TO TIME-SHARED BASIC
(02000-90016) — June, 1972

A GUIDE TO HP EDUCATIONAL BASIC
(02116-91773) — September, 1971

HP BASIC (02116-9077) — April, 1970

2000A: A GUIDE TO TIME-SHARED BASIC
(02000-90002) — August, 1970

2000F: A GUIDE TO TIME-SHARED BASIC
(02000-90044) — August, 1972

Operating System Manuals:

2000C: TIME-SHARED BASIC OPERATOR'S GUIDE
(02000-900017) — June, 1972

2000B TO 2000C CONVERSION PROGRAM GUIDE
(02000-90031) — April, 1971

2000F: TIME-SHARED BASIC SYSTEM
OPERATOR'S GUIDE
(02000-90045) — August, 1972

2000A: TIME-SHARED BASIC SYSTEM
OPERATOR'S GUIDE
(02000-90001) — September, 1970

2000A TO 2000B CONVERSION GUIDE
(02000-90011) — August, 1970

Educational Applications Manuals:

2000C/2000F SYSTEM OPERATOR INSTRUCTIONS FOR EDUCATIONAL APPLICATION (02000-90046) – December, 1972

2000C/2000F INSTRUCTIONAL MANAGEMENT FACILITY AND INSTRUCTIONAL DIALOGUE FACILITY–PROCTORS MANUAL (02000-90047) – December, 1972

2000C/2000F MATHEMATICS DRILL AND PRACTICE PROGRAM–PROCTORS MANUAL (02000-90051) – December, 1972

2000C/2000F INSTRUCTIONAL DIALOGUE FACILITY–AUTHORS MANUAL (02000-90055) – January, 1973

2000C/2000F MATHEMATICS DRILL AND PRACTICE PROGRAM–TEACHERS HANDBOOK (02000-90052) – December, 1972

COPYFL (02000-90032) – September, 1971

EDCALC (02000-90033) – September, 1971

INITIALIZATION (02000-90034) – September, 1971

INTEGER TO STRING (02000-90035) – September, 1971

DATE AND TIME (02000-90036) – September, 1971

2000C/2000F INTRODUCTION TO MATHEMATICS DRILL AND PRACTICE (02000-90050) – August, 1972

2000C/2000F MATHEMATICS DRILL AND PRACTICE CURRICULUM GUIDE (02000-90053) – August, 1972

COURSE DEVELOPERS' MANUAL FOR IDF-1 AND IMF-1 (02000-90061) – February, 1973

UPSHIFT (02000-90037) – September, 1971

CHARACTER REMOVAL (02000-90038) – September, 1971

KEY WORD SEARCH (02000-90039) – September, 1971

DOWNSHIFT (02000-90040) – September, 1971

STRING MATCH WITH "DON'T CARES" (02000-90041) – September, 1971

STRING TO NUMBER (02000-90042) – September, 1971

STUDENT RESPONSE ANALYSIS (02000-90043) – September, 1971

The preceding publications are available at nominal cost through your local HP sales office.

In addition, both 35mm slides and video tape instructional courses on programming in the BASIC language have been prepared. Again, contact the nearest HP sales office for more information.

The HP Program Catalog lists abstracts for over 700 programs written in FORTRAN, ALGOL, and HP Assembly Language suitable for various configurations of non BASIC language Systems. This catalog is available at no charge from your local HP sales and support office. 5950-9226

An active educational group for BASIC language computer users also invites inquiries. For more information on this activity, contact the Executive Secretary; Educational Users' Group, Hewlett-Packard Company; 11000 Wolfe Road, Cupertino, California 95014.

CLASSIFICATION CODE LISTING

LEGEND
Volumes: T = HP BASIC, E = Education, H = Huntington, S = Stanford
System Specifications: A = 2000A, B = 2000B, C = 2000C, F = 2000F, S = Single Terminal

Volume	System Specification	Class Code	Order No.	Name	Title
T	(A101)		36003A	MACRO	: A TEXT AND FILE PROCESSING SYSTEM
T	(C101)		36200B	TIDE	: TEXT EDITOR FOR THE 2000C
T	(A101)		36204A	TIOEX	: SYMBOLIC FILE EDITOR
T	(A102)		36004A	SLSMN	: SALES MANAGER'S INFORMATION RETRIEVAL SYSTEM
T	(A102)		36231A	ADDRES	: ADDRESS LABELS
*T	(C102)		36250B	FINDIT	: INFORMATION RETRIEVAL SYSTEM
*T	(F102)		36284A	MESSAG	: INTERTERMINAL COMMUNICATOR
T	(A104)		36005B	FORMAT	: ALLOWS SPECIAL FORMATTING OF DATA PRINTOUT
T	(B104)		36117A	DATES	: COMPUTES DATE FROM SYSTEM CLOCK
T	(B104)		36125B	HELLO	: TYPES DATE, TIME, AND PORT NUMBER ON TERMINAL
T	(A104)		36176A	STGINT	: STRING-INTEGER CONVERSIONS
T	(C104)		36220A	CHARS	: ASCII CHARACTER SET
*T	(C104)		36256B	ASCII*	: CREATES AN ASCII FILE CONTAINING ALL 256 ASCII CHARACTERS
T	(C104)		36257A	-ASCII	: ASCII CODE GENERATOR
T	(A104)		36258A	KEYSIG	: GIVES MAJOR SCALES
T	(A104)		36259A	MUSIC5	: FINDS DOMINANT SEVENTHS
*T	(F104)		36296A	ALFTOV	: ALPHA TO VARIABLE CONVERSION
*T	(F104)		36297A	TIMER	: TIME OF THE DAY
*T	(F104)		36298A	DATER	: DATE AND DAY OF THE WEEK
T	(C104)		36548A	GTAPID	: PAPER TAPE TITLER
T	(C107)		36122A	SORT	: FILE SERIAL STRING SORT
T	(C107)		36145A	STGSRT	: SORTS STRINGS FROM FILES
T	(A107)		36206A	FILSOR	: FILE POINTER SORT/POINTER SORT
T	(A107)		36206A	POISOR	: FILE POINTER SORT/POINTER SORT
T	(C107)		36232A	IRV	: FILE SORT ROUTINE
*T	(A107)		36292A	ALPHA	: ALPHANUMERIC SORT
T	(A108)		36197A	JULIAN	: JULIAN CALENDAR FOR THE CURRENT YEAR
T	(A108)		36218A	HPMLUT	: LIST/DUMP HP ASSEMBLER FILES
T	(A108)		36218A	HPMLIT	: LIST/DUMP HP ASSEMBLER FILES
T	(A108)		36222A	TALK	: TIME SHARING SYSTEM COMMUNICATION
*T	(F108)		36248A	NEWIDS	: CREATES NEW USER NUMBERS
T	(C108)		36272A	FILIS	: FILE LISTING PROGRAM/INSTRUCTIONS
T	(C108)		36272A	FILIS1	: FILE LISTING PROGRAM/INSTRUCTIONS
*T	(F108)		36279A	JULI	: ONE PAGE JULIAN CALENDAR
*T	(F108)		36288A	CALNDR	: PRINTS A CALENDAR
T	(A110)		36006A	FILMAN	: FILE MANAGER
T	(A110)		36007A	FILIN	: KEYBOARD FILE LOADING PROGRAM
T	(A110)		36008C	FILDUM	: PAPER TAPE FILE DUMP
T	(A110)		36009C	FILIST	: LISTS FILE CONTENTS BY RECORD NUMBER
T	(A110)		36010C	FILOAD	: LOADS A FILE FROM THE TELETYPE
T	(A110)		36011A	FILREA	: REENTERS THE DATA TAPE DUMPED BY FILDUM
T	(A110)		36012B	FLCOPY	: COPIES ONE FILE INTO ANOTHER
T	(B110)		36221A	XTRACT	: FILE UTILITY PROGRAM
T	(B110)		36221A	TAPDUM	: FILE UTILITY PROGRAM
*T	(F110)		36247A	FILRPT	: REPORTS FILE CONTENTS AND STRUCTURE
T	(C205)		36185A	TERTES	: TELETYPE-CRT DIAGNOSTIC
T	(A211)		36116A	BASTES	: BASIC TEST PROGRAM
T	(C211)		36143C	XREF	: BASIC LANGUAGE PROGRAM CROSS-REFERENCE GENERATOR
*T	(F212)		36287A	DATA	: DUMPS FILE TO DATA STATEMENTS
T	(A217)		36013A	TESTTY	: TESTS THE OPERATION OF A REMOTE TELETYPE
T	(A301)		36034A	SPHERE	: SOLVES SPHERICAL TRIANGLES
T	(A301)		36035B	EUCLID	: COMPUTES LARGEST COMMON FACTOR OF TWO INTEGERS
T	(A301)		36036A	SQRS	: AN INTEGER AS THE SUM OF FOUR SQUARES
T	(A301)		36037A	FACTOR	: FINDS PRIME FACTORS OF POSITIVE INTEGERS
T	(B301)		36131A	CALCOM	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 1 OF 2)
T	(B301)		36131A	CALPLT	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 2 OF 2)
T	(A301)		36168A	CALC2	: THREE REGISTER CALCULATOR PROGRAM
T	(A302)		36015A	CALC	: EXTENDED PRECISION CALCULATOR
*T	(A302)		36016B	LRGNUM	: LARGE NUMBER ADDITION & MULTIPLICATION
T	(A302)		36144A	EXTPRE	: 40-DIGIT PRECISION MATHEMATICS
T	(A303)		36017A	FNCTS	: COMPUTES TRIG FUNCTIONS FOR COMPLEX ARGUMENTS
T	(A303)		36018A	SQR(Z)	: COMPUTES SQUARE ROOT OF A COMPLEX NUMBER
T	(A303)		36118A	CXARTH	: VECTOR ARITHMETIC
T	(A303)		36119A	CXEXP	: VECTOR EXPONENTIATION
T	(A306)		36019A	BESSEL	: CALCULATES BESSEL FUNCTION OF FIRST KIND
T	(S309)		36188A	POLY	: POLYNOMIAL APPROXIMATION
T	(A309)		36246A	POLFTE	: FITS LEAST-SQUARES POLYNOMIALS
T	(A310)		36021A	INTGRT	: COMPUTES NUMERIC INTEGRAL OF A FUNCTION
*T	(A310)		36022B	ROMINT	: INTEGRATES A FUNCTION (ROMBERG METHOD)
T	(A311)		36024A	ROOTER	: FINDS THE ROOTS OF POLYNOMIALS
T	(A312)		36025A	CDETER	: COMPUTES VALUE OF COMPLEX DETERMINANT
T	(B312)		36263A	DETER4	: DETERMINANTS, CHARACTERISTIC POLYNOMIALS AND INVERSES OF MATRICES
T	(A314)		36026A	SIMEQN	: SOLVES SIMULTANEOUS LINEAR EQUATIONS
T	(A314)		36027A	CROUT1	: SOLVES SIMULTANEOUS LINEAR EQUATIONS
T	(A314)		36196A	SOLVIT	: SIMULTANEOUS LINEAR EQUATIONS USING GAUSSIAN REDUCTION
S	(C314)		36547A	GSIMEQ	: SIMULTANEOUS LINEAR EQUATIONS
T	(A316)		36028A	CTRFFT	: COMPLEX TO REAL FAST FOURIER TRANSFORM
T	(A316)		36029A	RTCFFT	: REAL TO COMPLEX FAST FOURIER TRANSFORM
T	(A316)		36030A	GFFT	: GENERAL FAST FOURIER TRANSFORM
T	(A317)		36031A	DERIV	: DERIVATIVE OF A FUNCTION AT A POINT
T	(A318)		36032A	DE-1OR	: 1ST ORDER DIFFERENTIAL EQUATION
T	(A318)		36033A	DE-2OR	: 2ND ORDER DIFFERENTIAL EQUATION
T	(A401)		36170A	T-TEST	: TEST OF HYPOTHESES USING STUDENTS T DISTRIBUTION
T	(A404)		36023B	POLFIT	: FITS LEAST-SQUARES POLYNOMIALS
T	(A404)		36038C	CURFIT	: PERFORMS LEAST-SQUARES FIT
T	(A404)		36054B	REGCOR	: REGRESSION/CORRELATION
T	(A404)		36178A	MULREG	: MULTIPLE REGRESSION/CORRELATION
T	(A404)		36186A	MULTX	: LEAST-SQUARES FIT, MULTIPLE Y'S PER X
S	(C404)		36542A	GRGPLT	: SIMPLE REGRESSION AND PLOT
T	(A405)		36050B	RANDEV	: GENERATES RANDOM DEVIATES-SUBROUTINE
T	(A405)		36203A	RANDOM	: RANDOM NUMBER GENERATOR
T	(B405)		36264A	RNDORD	: PLACING INTEGERS IN RANDOM ORDER
S	(C405)		36551A	GWRULL	: SUBJECTIVE PROBABILITY - RANDOM VALUES
T	(A406)		36041A	BINOPO	: PROBABILITY DISTRIBUTION COMPARISONS
S	(C406)		36549A	GTASPD	: SUBJECTIVE PROBABILITY DISTRIBUTION
T	(A407)		36042A	CHISQ	: COMPUTES PROBABILITY OF CHI-SQUARE VALUES
T	(A407)		36043B	CHISQS	: CHI-SQUARE STATISTICS FOR M*N CONTINGENCY TABLES
T	(A407)		36052A	STAT2	: MANN-WHITNEY 2 SAMPLE RANK TEST
T	(A407)		36053A	STAT3	: SPEARMAN RANK CORRELATION COEFFICIENTS
*T	(F407)		36606A	FISHER	: FISHER'S EXACT PROBABILITY TEST
T	(A408)		36001A	LOGRAM	: LOG-ON TAPE ANALYZER
T	(A408)		36045A	GEOMEN	: STATISTICS OF GEOMETRIC DISTRIBUTION
T	(A408)		36055B	HISTOG	: A HISTOGRAM FORMED FROM A SET OF NUMBERS
T	(A408)		36120A	FC	: ANALYSIS OF LOG TAPE
T	(A408)		36136A	SCORES	: COMPUTES MEAN, STANDARD DEVIATION AND STANDARD SCORES FOR TEST SCORES
T	(A408)		36137A	KR20	: ITEM ANALYSIS AND KUDER-RICHARDSON FORMULA 20 RELIABILITY
T	(A408)		36191B	FRQ	: FREQUENCY BETWEEN BOUNDRIES
S	(C408)		36541A	GRANK	: RANKING STATISTICS
T	(A410)		36039B	ANVAR1	: ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN
T	(A410)		36040B	ANVAR2	: ANALYSIS OF VARIANCE (LATIN SQUARE DESIGN)
T	(A410)		36172A	ANVAR3	: ANALYSIS OF VARIANCE FOR A TWO VARIABLES OF CLASSIFICATION DESIGN
T	(A410)		36173A	ANVAR4	: TWO-WAY ANALYSIS OF VARIANCE FOR A TWO-WAY EXPERIMENT
*T	(F410)		36271A	ANOVA3	: THREE FACTORIAL ANALYSIS OF VARIANCE
*T	(A410)		36294A	ANCOV	: ANALYSIS OF COVARIANCE
S	(C410)		36501A	GANOVA	: ANALYSIS OF VARIANCE (2-WAY)
*T	(F410)		36607A	STAT19	: KRUSKAL-WALLIS ONE WAY ANALYSIS OF VARIANCE
T	(S509)		36180A	SUNSET	: SUNRISE-SUNSET PREDICTOR
T	(A511)		36061A	LOVALV	: COMPUTES LIQUID CONTROL VALVE COEFFICIENTS
T	(A513)		36056A	ANALAD	: CIRCUIT ANALYSIS
T	(A513)		36057A	ACNODE	: AC CIRCUIT ANALYSIS PROGRAM
T	(A513)		36060A	LPFLTR	: DESIGNS LOW-PASS FILTERS
T	(A513)		36062A	MICRO	: MICROWAVE PARAMETERS CONVERSION
T	(A513)		36064A	MIXSPR	: MIXER SPURIOUS RESPONSE PROGRAM
*T	(F513)		36293A	ACTFIL	: ACTIVE FILTER DESIGN
T	(A514)		36058A	HTXFR	: TWO DIMENSIONAL HEAT TRANSFER
*T	(F514)		36635A	METRIC	: CONVERTS ENGLISH TO METRIC
T	(A515)		36109A	REMDES	: RECOMMENDS CORRECT STEEL BEAM USE
T	(A516)		36059A	DEBYE	: COMPUTES DEBYE OR EINSTEIN FUNCTION
T	(A519)		36063A	TMFCEV	: TIME FUNCTION EVALUATOR
T	(A603)		36171B	CPATH	: CRITICAL PATH EVALUATION
S	(C603)		36504A	GCPATH	: CRITICAL PATH ANALYSIS
S	(C603)		36505A	GCPM1	: CRITICAL PATH ANALYSIS
S	(C604)		36529A	GNETFL	: NETWORK FLOW
T	(A605)		36068B	LINPRO	: LINEAR PROGRAMMING MODEL
T	(A605)		36230A	TRANSP	: TRANSPORTATION PROBLEM
S	(C605)		36512A	GINTLP	: LINEAR PROGRAMMING-VARIABLES RESTRICTED TO VALUES OF ONE OR ZERO
S	(C605)		36516A	GLP	: LINEAR PROGRAMMING
S	(C605)		36517A	GLPSA1	: LINEAR PROGRAMMING TWO-PHASE SIMPLEX METHOD
T	(A606)		36065C	DECSN	: TOP MANAGEMENT DECISION GAME
S	(C606)		36550A	GVOTE	: COMMITTEE CHOICE ANALYSIS
S	(C607)		36552A	GSSS	: SMALL SYSTEMS SIMULATOR
T	(A609)		36069A	LNTRND	: LINEAR TREND FORECASTING
T	(A610)		36067A	DYNPRO	: DYNAMIC PROGRAMMING MODEL
*T	(A706)		36213B	CTC4	: CTC PAYROLL PROGRAMS
T	(A708)		36177A	REP	: DATA CENTER INVENTORY REPORT GENERATOR
T	(A708)		36177A	IN	: DATA CENTER INVENTORY REPORT GENERATOR
T	(A708)		36210B	CTC1	: CTC MANUFACTURING PARTS CONTROL
T	(A708)		36211B	CTC2	: CTC INVENTORY CONTROL FOR FINISHED PRODUCTS
*T	(F709)		36638A	CTC6	: CTC ACCOUNTS PAYABLE
T	(A710)		36095A	SALES	: SALES COMMISSION REPORT
T	(A711)		36214B	CTC5	: CTC ACCOUNTS RECEIVABLE
T	(A712)		36074A	ANNUIT	: ANNUITY ANALYSIS
T	(A712)		36075A	BALSHT	: PROFORMA INCOME STATEMENT AND BALANCE SHEET
T	(A712)		36082A	DEPCOM	: DEPRECIATION METHOD COMPARISON
T	(A712)		36083A	EQUITY	: COST OF EQUITY CAPITAL
T	(A712)		36089B	INSTMT	: INCOME STATEMENT
T	(A712)		36090A	LENDER	: SIMPLE LOAN ANALYSIS
T	(A712)		36091A	LESSEE	: LEASE ANALYSIS AS DETERMINED BY THE LESSEE
T	(A712)		36093A	MKBUY	: MAKE-BUY DECISION ANALYSIS
T	(A712)		36094B	MORGAG	: MORTGAGE ANALYSIS
T	(A712)		36142B	CSHFL	: CASH FLOW ANALYSIS
T	(C712)		36194A	LEASIN	: LEASE INCOME
T	(C712)		36226A	LOAN	: LOAN AMORTIZATION
S	(C712)		36510A	GFNRAT	: FINANCIAL RATIOS
S	(C712)		36513A	GIRRPV	: INVESTMENT RETURN (CASH FLOW)
S	(C712)		36543A	GRISKA	: RISK ANALYSIS IN CAPITAL INVESTMENT
T	(A713)		36076B	BNDPRC	: BOND PRICE ANALYSIS
T	(A713)		36077A	BNDSWH	: BOND SWITCH ANALYSIS
T	(A713)		36078B	BNDYLD	: BOND YIELD ANALYSIS
T	(A713)		36079A	BNKRSV	: BANK RESERVE CALCULATIONS
T	(A713)		36080A	CAPINV	: CAPITAL INVESTMENT ANALYSIS
T	(A713)		36084A	EXDRSK	: EXTENDED RISK ANALYSIS
T	(A713)		36085A	EXSMOO	: EXPONENTIAL SMOOTHING ON PRICE DATA
T	(A713)		36092A	MARKOW	: SECURITIES PORTFOLIO USING MARKOWITZ MODEL
T	(A713)		36096A	STKINC	: STOCK MERGER INCENTIVE PROGRAM
T	(A713)		36098A	STKRTN	: STOCK RETURNS REPORT
T	(A713)		36099A	STKSMO	: EXPONENTIAL SMOOTHING AS A STOCK GUIDE
T	(A713)		36100A	STKVAL	: STOCK VALUE & EVALUATION REPORT
T	(A713)		36101A	TRUINT	: TRUE ANNUAL INTEREST RATE ANALYSIS
T	(C713)		36179A	DROIPB	: DISCOUNTED RETURN ON INVESTMENT AND PAYBACK
S	(C713)		36503A	GCHLIN	: RATING INVESTMENT FUNDS
S	(C713)		36507A	GDAPI	: ABNORMAL PERFORMANCE INDEX
S	(C713)		36508A	GDPA	: EFFICIENT "CORNER" PORTFOLIOS

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CLASSIFICATION CODE LISTING

```
 S (C713) 36514A    GKASSF : WARRANT PRICE CALCULATION
 S (C713) 36515A    GKCOST : PRICE/EARNINGS RATIO CALCULATION
 S (C713) 36530A    GNMRVB : SECURITIES PORTFOLIO ANALYSIS AND
                             DETERMINATION
 S (C713) 36530A    GMRGB  : SECURITIES PORTFOLIO ANALYSIS AND
                             DETERMINATION
 S (C713) 36545A    GSTKVL : STOCK VALUATION
 S (C713) 36553A    GTHOR  : SECURITIES EPS GROWTH
 S (C713) 36554A    GVPDQT : PLOTTING DATA
 S (C713) 36554A    VPDQ   : PLOTTING DATA
 S (C713) 36557A    VRRC   : INVESTMENT STRATEGY ANALYSIS
 T (A714) 36086A    GNPSUM : GROSS NATIONAL PRODUCT SUMMARY
 T (A714) 36087A    INACNT : NATIONAL INCOME & PRODUCT ACCOUNTS
 T (A714) 36088A    IN/OUT : INPUT/OUTPUT ANALYSIS ON ECONOMIC FLOWS
 T (A716) 36073A    BUDGET : DEPARTMENTAL MANAGER'S BUDGETING PROGRAM
 T (A716) 36111A    EXPEND : BUDGET EXPENDITURES VS. TARGETS MONITOR
 T (A717) 36113B    RECRUT : RECRUTING INFORMATION SYSTEM
*T (A717) 362128    CTC3   : CTC PROJECTION PROGRAMS
 T (C718) 36164B    IATA.1 : CALCULATE AIR FREIGHT RATES
 T (C718) 36174A    TRCK.1 : CALCULATE TRUCK FREIGHT RATES
 T (C718) 36241A    IATA.C : CALCULATES BREAKPOINT OF IATA CONTAINERS
 T (A720) 36072A    SALSIM : SALARY SCHEDULE COST SIMULATOR
 E (C720) 36202A    ATTEND : ABSENTEE LISTING
*T (F720) 36282A    COLREG : COLLEGE REGISTRATION DEMO
 H (S720) 36340A    AVERG1 : AVERAGES AND CURVES GRADES
 H (S720) 36341A    FREQ   : HAR GRAPHS OF GRADES
 H (S720) 36342A    GRADE  : TEST GRADE FOR NUMBER OF QUESTIONS MISSED
 H (S720) 36343A    ITEM1  : NUMBER FREQUENCIES
 H (S720) 36344A    STAT   : STATISTICAL ANALYSIS OF LAB DATA
 T (C720) 36518A    GMARKS : GSB GRADING PROGRAM
 T (C720) 36518A    GMEDIT : GSB GRADING PROGRAM
 E (B801) 24241B    TOF    : INSTRUCTIONAL DIALOGUE FACILITY
 E (B801) 24242B    EXEC   : CAI EXECUTIVE PACKAGE
 E (B801) 24244A    SUMAIN : CAI SUBROUTINES
 E (B801) 208796    MATHB  : CAI MATH PACKAGE (2000B VERSION)
 E (B801) 243438    MATHC  : CAI MATH PACKAGE (2000C VERSION)
 E (B801) 36128A    TENS   : DRILL IN MULTIPLICATION OF NUMBERS WITH
                             TRAILING ZEROS
 E (B801) 36129A    DRILL  : BASIC ARITHMETIC DRILL
 E (B801) 36130A    TWQUES : MATHEMATICAL GAME OF TWENTY QUESTIONS
 E (A801) 36133A    WKSHT  : GENERATES MATH WORKSHEETS
 E (A801) 36159A    DIFFEQ : CAI IN SOLUTION OF LINEAR FIRST-ORDER
                             DIFFERENTIAL EQUATIONS
 E (A801) 36182A    SAT    : TRIGONOMETRIC SOLUTIONS OF TRIANGLES
 E (A801) 36205A    CADAII : COMPUTER ASSISTED ARITHMETIC DRILL
 E (A801) 36205A    CADAIM : COMPUTER ASSISTED ARITHMETIC DRILL
 E (A801) 36205A    CADAIP : COMPUTER ASSISTED ARITHMETIC DRILL
 E (A801) 36227A    FACTRL : FACTORIALS TO 1000 PLACES
 E (C801) 36237A    TRIFAC : FACTORING QUADRATIC TRINOMIALS
 E (C801) 36238A    SIPRAC : PRACTICE WITH SIGNED NUMBERS
 E (C801) 36239A    POLSUB : POLYNOMIAL SUBTRACTION
 E (C801) 36240A    BISQAR : SQUARING BINOMIALS
 E (B801) 36251A    LOGIC  : LOGIC EXAMINATION PROGRAMS
 E (C801) 36253A    EXTEND : INFINITE PRECISION MATH UTILITY PROGRAM
 E (B801) 36262A    CXSYSS : SOLVING SYSTEMS OF COMPLEX LINEAR
                             EQUATIONS
 E (A801) 36265A    GAME   : CLASSIC MATRIX OF GAME THEORY
 E (C801) 36277A    DIMIS  : X-Y AXIS SEGMENT PROGRAM
*E (C801) 36278A    SYSSOL : SOLVING SYSTEMS OF LINEAR EQUATIONS
 H (S801) 36333A    CRVLEN : COMPUTES LENGTH OF ANY CURVE
 H (S801) 36336A    RATIO  : SOLVES PROPORTIONS
 H (S801) 36337A    SETS   : UNION AND INTERSECTION OF SETS
 H (S801) 36338A    SURFAC : AREA OF SURFACE OF REVOLUTION
 H (S801) 36339A    VOLSOL : VOLUME OF SOLID OF REVOLUTION
 E (S801) 36360A    BAGELS : THREE-DIGIT NUMBER GUESSING
*E (F801) 36602A    TRUTH  : TRUTH TABLES FOR BOOLEAN EXPRESSIONS
*H (A801) 36620A    CVAREA : AREA UNDER CURVE
*H (A801) 36622A    LIMSIN : LIMIT OF (SIN X)/X
*H (A801) 36623A    PI     : CALCULATES PI
*H (A801) 36625A    ROOTS2 : QUADRATIC EQUATION SOLVER
*H (A801) 36626A    SLOPE  : FINDS DERIVATIVES
*H (A801) 36627A    SORT   : FINDS SQUARE ROOT
*H (A801) 36628A    STATAL : ARITHMETIC MEAN
 E (A810) 36014A    STOP   : TIES MATH CALCULATOR PACKAGE
 E (A810) 36139A    TUTOR  : TUTOR SERIES BASIC LANGUAGE PROGRAMMING
                             COURSE. TUT01 - TUT25
 E (A810) 36208A    STOPB  : MATH CALCULATOR PACKAGE
 E (C810) 36219A    BRAIN  : BRAIN SIMULATOR PROGRAM
 E (C810) 36254A    EXPRES : EXPRESSION SOLVING PROGRAM
 H (S810) 36334B    PLOTTR : FUNCTION PLOTTER
 H (S810) 36335A    QUADT  : NATURE OF GRAPH OF A 2ND ORDER EQUATION
                             IN TWO VARIABLES
 E (A820) 36148A    EXPNTL : CAI IN SIMPLE EXPONENTIAL FUNCTIONS OF
                             TIME
 E (A820) 36149A    SOLVER : SOLVES COMPLEX SIMULTANEOUS EQUATIONS
 E (A820) 36150A    NET-3  : COMPLEX NUMBER OPERATIONS
 E (A820) 36152A    AC-1   : COMPUTER AIDED PRACTICE IN EE AC ANALYSIS
 E (A820) 36153A    COMPLX : CAI IN ALGEBRA OF COMPLEX NUMBERS
 E (A820) 36154A    INDMTR : ANALYSIS OF A BALANCED POLYPHASE
                             INDUCTION MOTOR
 E (A820) 36155A    POLAR  : PLOTS SINGLE VARIABLE IN POLAR FORM
 E (A820) 36156A    DVDRS  : CAI IN VOLTAGE AND CURRENT DIVIDERS
 E (A820) 36157A    UNITS  : CAI IN INTERPRETATION OF EE UNITS
 E (A820) 36158A    RNET   : NETWORK SIMULTANEOUS EQUATIONS
 E (A820) 36166A    DFTERM : MATRIX DETERMINANT USING GAUSSIAN
                             ELIMINATION
 E (A820) 36167A    JACOBI : EIGENVALUES AND EIGENVECTORS OF A REAL
                             SYMMETRIC MATRIX
 S (C830) 36519A    GMCR01 : FISCAL POLICY GAME
 S (C830) 36519A    GMCR02 : FISCAL POLICY GAME
 S (C830) 36521A    GMCR06 : ECONOMIC POLICY GAME
 S (C830) 36521A    GMCR05 : ECONOMIC POLICY GAME
*H (A830) 36632A    CIRFLW : CIRCULAR FLOW MODEL
 E (A833) 36121A    ORBIT  : INTEGRATES EQUATIONS OF MOTION
 E (A833) 36134A    H-LIFE : HALF LIFE SIMULATION
 E (A833) 36184A    SPCTRA : OPTICAL ABSORPTION SPECTRA SIMULATION, 2-
                             SPECIES EQUILIBRIUM MIXTURES
 E (B833) 36260A    IONIC  : DRILL ON FORMULAS AND CHARGES OF IONS
 E (B833) 36261A    ISOMER : DRILL ON NAMING ALKANES
 E (B833) 36266A    IONIC1 : DRILL ON FORMULAS OF IONIC COMPOUNDS
 E (B833) 36267A    ELMENT : DRILL ON SYMBOLS FOR CHEMICAL ELEMENTS
*E (F833) 36295A    CHEM   : SELF-CORRECTING CHEMISTRY TEST
 H (S833) 36300A    DROS   : GENETIC CHARACTERISTICS
 H (S833) 36301A    EVOLU  : NATURAL SELECTION EXPERIMENT
 H (S833) 36302A    GAMGN  : PROCESS OF GAMETOGENESIS
 H (S833) 36303A    NZYMC  : ENZYMATIC REACTION RATES
 H (S833) 36304A    NZYM2  : ENZYME REACTION RATE
 H (S833) 36305A    PHOSYN : PHOTOSYNTHESIS EXPERIMENT
 H (S833) 36306A    DECAY1 : RADIOACTIVE DECAY GAME
 H (S833) 36307A    DECAY2 : NUCLEAR DECAY
 H (S833) 36308A    EQUIL1 : EQUILIBRIUM SYSTEMS
 H (S833) 36308A    EQUIL2 : EQUILIBRIUM SYSTEMS
 H (S833) 36309A    KINET  : KINETIC REACTION
 H (A833) 36310A    MASSD  : MASS DEFECT
 H (S833) 36311A    PRCNT  : PERCENT COMPOSITION
 H (S833) 36312A    BFIELD : MAGNETIC FIELD PICTURE
 H (S833) 36313A    BOHR   : ENERGY LEVEL DIAGRAM
 H (S833) 36314A    CALORI : CALORIMETRY EXPERIMENT
 H (S833) 36315A    KINERV : REVIEW OF KINEMATICS
 H (S833) 36316A    NEWTN2 : NEWTON'S 2ND LAW
 H (S833) 36317A    PHOTEL : PHOTOELECTRIC EFFECT
 H (S833) 36318A    PHOTON : ENERGY LEVEL PROBLEM
 H (S833) 36319A    REFLCT : LEAST TIME PRINCIPLE AND LIGHT
 H (S833) 36320A    SNELL  : SNELL'S LAW
 H (S833) 36321A    SPACE  : SPACECRAFT ORBITS
 H (S833) 36322A    VFIELD : POTENTIAL FIELD PICTURE
 H (S833) 36323A    VLOCTY : INSTANTANEOUS VELOCITY
 H (S833) 36324B    WAVES  : SUM OF TWO WAVES
 H (S833) 36325A    CLOUDS : CLOUD FORMATION
 H (S833) 36326A    WATER1 : WATER BUDGET PROBLEM CHECK
 H (S833) 36327A    WATER2 : WATER BUDGET
 H (S833) 36350A    PLANK  : A PHOTOELECTRIC SIMULATION
 H (A833) 36351A    SLITS  : YOUNG'S DOUBLE SLIT EXPERIMENT
 E (S833) 36353A    AFLIFE : JOHN CONWAY'S GAME OF LIFE
*E (F833) 36603A    ALEKA  : PROPAGATION OF ERROR
*H (A833) 36611A    MEMBR  : DIFFUSION EXPERIMENT
*H (A833) 36613A    ATWT   : CALCULATES ATOMIC WEIGHT
*H (A833) 36614A    AVOGA  : AVOGADRA'S NUMBER
*H (A833) 36615A    EMPIR  : CALCULATES EMPIRICAL FORMULAS
*H (A833) 36616A    MOLAR  : ACID-BASE TITRATION
*H (A833) 36617A    PHPOH  : PH, POH, PCT, DISSOCIATION
*H (A833) 36618A    STOICH : MASS VOLUME
*H (A833) 36629A    EFIELD : ELECTRIC FIELD STRENGTH
*H (A833) 36630A    LENSES : SOLVES LENS PROBLEMS
*H (A833) 36631A    PRJTL  : PROJECTILE MOTION
*H (F833) 36639A    POLUT  : WATER POLLUTION SIMULATION
*H (F833) 36641A    STERL  : FLY POPULATION CONTROL
*H (F833) 36642A    GENE1  : GENETICS SIMULATION
 E (A850) 36276A    MUSIC2 : TRIAD SOLVING PROGRAM
 E (B860) 36187A    WHEELS : AUTO PURCHASE AND MAINTENANCE SYSTEM
 H (S860) 36328A    BALANC : TRADE AND PAYMENT BALANCES
 H (S860) 36329A    BANK   : SOLVES FINANCIAL PROBLEMS
 H (S860) 36330A    CONSMP : DEPRESSION/EQUILIBRIUM
 H (S860) 36331A    STOCK  : STOCK MARKET SIMULATION
*E (F860) 36604A    CHARGE : CHARGE ACCOUNT SIMULATION
*H (F860) 36640A    POLSYS : SIMULATION OF CITY COUNCIL
 E (C871) 36175A    CARLOS : COMPUTER-ASSISTED REVIEW LESSONS ON
                             SYNTAX FOR SPANISH II
 E (A880) 36002B    SETUP  : MANGEMENT SIMULATION GAME FOR THE HP 2000A
 E (A880) 36002B    MANAG  : MANGEMENT SIMULATION GAME FOR THE HP 2000A
 E (A880) 36002B    RITE   : MANAGEMENT SIMULATION GAME FOR THE HP
                             2000A
 E (B880) 36193A    SETUPB : MANAGEMENT SIMULATION GAME FOR THE 2000B
 E (B880) 36193A    MANAGB : MANAGEMENT SIMULATION GAME FOR THE 2000B
 E (B880) 36193A    MANAGU : MANAGEMENT SIMULATION GAME FOR THE 2000B
 E (B880) 36193A    RITET  : MANAGEMENT SIMULATION GAME FOR THE 2000B
 E (A880) 36199A    MRKSIM : MARKETING SIMULATION
 E (C880) 36209A    SETUPC : MANAGEMENT SIMULATION GAME FOR THE 2000C
 E (C880) 36209A    MANAGC : MANAGEMENT SIMULATION GAME FOR THE 2000C
 E (A880) 36233A    LABOR  : LABOR/MANAGEMENT BARGAINING
 S (C880) 36502A    GSPMG  : STANFORD PORTFOLIO MANAGEMENT GAME
 T (A901) 36110A    CANNON : HP 7200A GAME DEMO FOR TIME-SHARED BASIC
 T (A901) 36169A    FIFTY! : EXTENDED PRECISION FACTORIAL
 T (A903) 36102A    GOLF   : A GOLF GAME
 T (A903) 36103A    BLJACK : GAME OF TWENTY-ONE
 T (A903) 36105A    BATNUM : PLAYS THE BATTLE OF NUMBERS GAME
 T (A903) 36106A    QUBIC  : PLAYS 3-DIMENSIONAL TIC-TAC-TOE
 T (A903) 36107A    SLOT   : SIMULATES A SLOT MACHINE (USE IN NEVADA
                             ONLY)
 T (A903) 36108A    SNOOPY : DRAWS SNOOPY ON THE TELEPRINTER
 T (A903) 36138A    LIFE   : A GAME OF CELLULAR GENERATION GROWTH
 T (A903) 36181A    FTBL   : FOOTBALL
 T (A903) 36236A    BATTLE : BATTLESHIP GAME
 T (C903) 36243A    STTR1  : STAR TREK
 T (A903) 36244A    KENO   : $25,000.00 KENO GAME
*T (A903) 3660IA    DRAG   : SIMULATES A DRAG RACE
 T (A904) 36104A    PLOT   : PLOTS A GIVEN FUNCTION ON THE TELETYPE
 T (A904) 36112A    FPLOT  : FUNCTION PLOT
 T (A904) 36114A    TITLE  : CHARACTER GENERATION
 T (A904) 36115A    GRAPHS : DEMO PLOT PROGRAM FOR HP 7200 PLOTTER
 T (A904) 36165A    FGRAPH : SIMULTANEOUS FUNCTION GRAPHER
 T (A904) 36228A    PLOTWD : WORD PLOTTER
 T (C904) 36235A    HISS   : SAMPLE STATISTICS AND HISTOGRAM FORMED
                             FROM A SET OF NUMBERS
*T (F904) 36299A    PRINT  : GENERATES LARGE LETTERS
 S (C904) 36555A    VCHART : INVESTMENT DECISIONS USING TEKTRONIX 4010
 S (C904) 36556A    VREGPL : PLOTTING X AND Y VARIABLES USING
                             TEKTRONIX 4010
 S (C904) 36558A    VSUB   : DISPLAY ROUTINE USING TEKTRONIX 4010
 T (C904) 36559A    VTTT   : TIC-TAC-TOE ON THE TEKTRONIX 4010 DISPLAY
                             TERMINAL
 T (B904) 72050A    HPLOTB : AUTOMATIC PLOTTING PROGRAM FOR THE HP
                             2000B
 T (C904) 72051A    HPPLOT : AUTOMATIC PLOTTING PROGRAM
```

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

LEGEND
Volumes: T = HP BASIC, E = Education, H = Huntington, S = Stanford
System Specifications: A = 2000A, B = 2000B, C = 2000C, F = 2000F, S = Single Terminal

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973
Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

*, Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

CROSS REFERENCE INDEX

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

LEGEND
Volumes: T = HP BASIC, E = Education, H = Huntington, S = Stanford
System Specifications: A = 2000A, B = 2000B, C = 2000C, F = 2000F, S = Single Terminal

ALPHABETICAL SUMMARY

Volumes	Name	Title	System Specifications	Class Code	Order No.
T	-ASCII	: ASCII CODE GENERATOR		(C104)	36257A
E	AC-1	: COMPUTER AIDED PRACTICE IN EE AC ANALYSIS		(A820)	36152A
T	ACNODE	: AC CIRCUIT ANALYSIS PROGRAM		(A513)	36057A
*T	ACTFIL	: ACTIVE FILTER DESIGN		(F513)	36293A
T	ADDRES	: ADDRESS LABELS		(A102)	36231A
E	AFLIFE	: JOHN CONWAY'S GAME OF LIFE		(S833)	36353A
*E	ALERA	: PROPAGATION OF ERROR		(F833)	36603A
*T	ALFTOV	: ALPHA TO VARIABLE CONVERSION		(F104)	36296A
*T	ALPHA	: ALPHANUMERIC SORT		(A107)	36292A
T	ANALAD	: CIRCUIT ANALYSIS		(A513)	36056A
*T	ANCOV	: ANALYSIS OF COVARIANCE		(A410)	36294A
T	ANNUIT	: ANNUITY ANALYSIS		(A712)	36074A
*T	ANOVA3	: THREE FACTORIAL ANALYSIS OF VARIANCE		(F410)	36271A
T	ANVAR1	: ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN		(A410)	36039B
T	ANVAR2	: ANALYSIS OF VARIANCE (LATIN SQUARE DESIGN)		(A410)	36040B
T	ANVAR3	: ANALYSIS OF VARIANCE FOR A TWO VARIABLES OF CLASSIFICATION DESIGN		(A410)	36172A
T	ANVAR4	: TWO-WAY ANALYSIS OF VARIANCE FOR A TWO-WAY EXPERIMENT		(A410)	36173A
*T	ASCII*	: CREATES AN ASCII FILE CONTAINING ALL 256 ASCII CHARACTERS		(C104)	36256B
E	ATTEND	: ABSENTEE LISTING		(C720)	36202A
*H	ATWT	: CALCULATES ATOMIC WEIGHT		(A833)	36613A
H	AVERGI	: AVERAGES AND CURVES GRADES		(S720)	36340A
*H	AVOGA	: AVOGADRO'S NUMBER		(A833)	36614A
E	BAGELS	: THREE-DIGIT NUMBER GUESSING		(SA01)	36360A
H	BALANC	: TRADE AND PAYMENT BALANCES		(SB60)	36328A
T	BALSHT	: PROFORMA INCOME STATEMENT AND BALANCE SHEET		(A712)	36075A
H	BANK	: SOLVES FINANCIAL PROBLEMS		(SB60)	36329A
T	BASTES	: BASIC TEST PROGRAM		(A211)	36116A
T	BATNUM	: PLAYS THE BATTLE OF NUMBERS GAME		(A903)	36105A
T	BATTLE	: BATTLESHIP GAME		(A903)	36236A
T	BEMDES	: RECOMMENDS CORRECT STEEL BEAM USE		(A515)	36109A
T	BESSEL	: CALCULATES BESSEL FUNCTION OF FIRST KIND		(A306)	36019A
H	BFIELD	: MAGNETIC FIELD PICTURE		(SH33)	36312A
T	BINOPO	: PROBABILITY DISTRIBUTION COMPARISONS		(A406)	36041A
E	BISQAR	: SQUARING BINOMIALS		(C801)	36240A
T	BLJACK	: GAME OF TWENTY-ONE		(A903)	36103A
T	BNDPRC	: BOND PRICE ANALYSIS		(A713)	36076B
T	BNDSWH	: BOND SWITCH ANALYSIS		(A713)	36077A
T	BNDYLD	: BOND YIELD ANALYSIS		(A713)	36078B
T	BNKRSV	: BANK RESERVE CALCULATIONS		(A713)	36079A
H	BOHR	: ENERGY LEVEL DIAGRAM		(S833)	36313A
E	BRAIN	: BRAIN SIMULATOR PROGRAM		(C810)	36219A
T	BUDGET	: DEPARTMENTAL MANAGER'S BUDGETING PROGRAM		(A716)	36073A
E	CADA1I	: COMPUTER ASSISTED ARITHMETIC DRILL		(AH01)	36205A
E	CADA1M	: COMPUTER ASSISTED ARITHMETIC DRILL		(AH01)	36205A
E	CADA1P	: COMPUTER ASSISTED ARITHMETIC DRILL		(AH01)	36205A
T	CALC	: EXTENDED PRECISION CALCULATOR		(A302)	36015A
T	CALC2	: THREE REGISTER CALCULATOR PROGRAM		(A301)	36168A
T	CALCOM	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 1 OF 2)		(B301)	36131A
*T	CALNDR	: PRINTS A CALENDAR		(F108)	36288A
H	CALORI	: CALORIMETRY EXPERIMENT		(SH33)	36314A
T	CALPLT	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 2 OF 2)		(H301)	36131A
T	CANNON	: HP 7200A GAME DEMO FOR TIME-SHARED BASIC		(A901)	36110A
T	CAPINV	: CAPITAL INVESTMENT ANALYSIS		(A713)	36080A
E	CARLOS	: COMPUTER-ASSISTED REVIEW LESSONS ON SYNTAX FOR SPANISH II		(CH71)	36175A
T	CDETER	: COMPUTES VALUE OF COMPLEX DETERMINANT		(A312)	36025A
*E	CHARGE	: CHARGE ACCOUNT SIMULATION		(FB60)	36604A
T	CHARS	: ASCII CHARACTER SET		(C104)	36220A
*E	CHEM	: SELF-CORRECTING CHEMISTRY TEST		(F833)	36295A
T	CHISQ	: COMPUTES PROBABILITY OF CHI-SQUARE VALUES		(A407)	36042A
T	CHISQS	: CHI-SQUARE STATISTICS FOR M*N CONTINGENCY TABLES		(A407)	36043B
*H	CIRFLW	: CIRCULAR FLOW MODEL		(A830)	36632A
H	CLOUDS	: CLOUD FORMATION		(SH33)	36325A
*T	COLREG	: COLLEGE REGISTRATION DEMO		(F720)	36282A
E	COMPLX	: CAI IN ALGEBRA OF COMPLEX NUMBERS		(A820)	36153A
H	CONSMP	: DEPRESSION/EQUILIBRIUM		(SB60)	36330A
T	CPATH	: CRITICAL PATH EVALUATION		(A603)	36171B
T	CROUT1	: SOLVES SIMULTANEOUS LINEAR EQUATIONS		(A314)	36027A
H	CRVLEN	: COMPUTES LENGTH OF ANY CURVE		(SB01)	36333A
T	CSHFL	: CASH FLOW ANALYSIS		(A712)	36142B
T	CTC1	: CTC MANUFACTURING PARTS CONTROL		(A708)	36210B
T	CTC2	: CTC INVENTORY CONTROL FOR FINISHED PRODUCTS		(A708)	36211B
*T	CTC3	: CTC PROJECTION PROGRAMS		(A717)	36212B
*T	CTC4	: CTC PAYROLL PROGRAMS		(A706)	36213B
T	CTC5	: CTC ACCOUNTS RECEIVABLE		(A711)	36214B
*T	CTC6	: CTC ACCOUNTS PAYABLE		(F709)	36638A
T	CTRFFT	: COMPLEX TO REAL FAST FOURIER TRANSFORM		(A316)	36028A
T	CURFIT	: PERFORMS LEAST-SQUARES FIT		(A404)	36038C
*H	CVAREA	: AREA UNDER CURVE		(A801)	36620A
T	CXARTH	: VECTOR ARITHMETIC		(A303)	36118A
T	CXEXP	: VECTOR EXPONENTIATION		(A303)	36119A
E	CXSYSS	: SOLVING SYSTEMS OF COMPLEX LINEAR EQUATIONS		(BB01)	36262A
*T	DATA	: DUMPS FILE TO DATA STATEMENTS		(F212)	36287A
*T	DATER	: DATE AND DAY OF THE WEEK		(F104)	36298A
T	DATES	: COMPUTES DATE FROM SYSTEM CLOCK		(B104)	36117A
T	DE-1OR	: 1ST ORDER DIFFERENTIAL EQUATION		(A318)	36032A
T	DE-2OR	: 2ND ORDER DIFFERENTIAL EQUATION		(A318)	36033A
T	DEBYE	: COMPUTES DEBYE OR EINSTEIN FUNCTION		(A516)	36059A
H	DECAY1	: RADIOACTIVE DECAY GAME		(S833)	36306A
H	DECAY2	: NUCLEAR DECAY		(S833)	36307A
T	DECSN	: TOP MANAGEMENT DECISION GAME		(A606)	36065C
T	DEPCOM	: DEPRECIATION METHOD COMPARISON		(A712)	36082A
T	DERIV	: DERIVATIVE OF A FUNCTION AT A POINT		(A317)	36031A
T	DETER4	: DETERMINANTS, CHARACTERISTIC POLYNOMIALS AND INVERSES OF MATRICES		(B312)	36263A
E	DETERM	: MATRIX DETERMINANT USING GAUSSIAN ELIMINATION		(A820)	36166A
E	DIFFEQ	: CAI IN SOLUTION OF LINEAR FIRST-ORDER DIFFERENTIAL EQUATIONS		(A801)	36159A
E	DIMIS	: X-Y AXIS SEGMENT PROGRAM		(CB01)	36277A
*T	DRAG	: SIMULATES A DRAG RACE		(A903)	36601A
E	DRILL	: BASIC ARITHMETIC DRILL		(HB01)	36129A
T	DROIPR	: DISCOUNTED RETURN ON INVESTMENT AND PAYBACK		(C713)	36179A
H	DROS	: GENETIC CHARACTERISTICS		(S833)	36300A
E	DVDRS	: CAI IN VOLTAGE AND CURRENT DIVIDERS		(A820)	36156A
T	DYNPRO	: DYNAMIC PROGRAMMING MODEL		(A610)	36067A
*H	EFIELD	: ELECTRIC FIELD STRENGTH		(A833)	36629A
E	ELMENT	: DRILL ON SYMBOLS FOR CHEMICAL ELEMENTS		(HB33)	36267A
*H	EMPIR	: CALCULATES EMPIRICAL FORMULAS		(AH33)	36615A
H	EQUIL1	: EQUILIBRIUM SYSTEMS		(S833)	36308A
H	EQUIL2	: EQUILIBRIUM SYSTEMS		(SH33)	36308A
T	EQUITY	: COST OF EQUITY CAPITAL		(A712)	36083A
T	EUCLID	: COMPUTES LARGEST COMMON FACTOR OF TWO INTEGERS		(A301)	36035B
H	EVOLU	: NATURAL SELECTION EXPERIMENT		(SH33)	36301A
T	EXDRSK	: EXTENDED RISK ANALYSIS		(A713)	36084A
E	EXEC	: CAI EXECUTIVE PACKAGE		(CH00)	24242B
T	EXPEND	: BUDGET EXPENDITURES VS. TARGETS MONITOR		(A716)	36111A
E	EXPNTL	: CAI IN SIMPLE EXPONENTIAL FUNCTIONS OF TIME		(AH20)	36148A
E	EXPRES	: EXPRESSION SOLVING PROGRAM		(CB10)	36254A
T	EXSMOO	: EXPONENTIAL SMOOTHING ON PRICE DATA		(A713)	36085A
E	EXTEND	: INFINITE PRECISION MATH UTILITY PROGRAM		(CB01)	36253A
T	EXTPRE	: 40-DIGIT PRECISION MATHEMATICS		(A302)	36144A
T	FACTOR	: FINDS PRIME FACTORS OF POSITIVE INTEGERS		(A301)	36037A
F	FACTRL	: FACTORIALS TO 1000 PLACES		(AH01)	36227A
T	FC	: ANALYSIS OF LOG TAPE		(A408)	36120A
T	FGRAPH	: SIMULTANEOUS FUNCTION GRAPHER		(A904)	36165A
T	FIFTY!	: EXTENDED PRECISION FACTORIAL		(A901)	36169A
T	FILDUM	: PAPER TAPE FILE DUMP		(A110)	36008C
T	FILIN	: KEYBOARD FILE LOADING PROGRAM		(A110)	36007A
T	FILIS	: FILE LISTING PROGRAM/INSTRUCTIONS		(C108)	36272A
T	FILISI	: FILE LISTING PROGRAM/INSTRUCTIONS		(C108)	36272A
T	FILIST	: LISTS FILE CONTENTS BY RECORD NUMBER		(A110)	36009C
T	FILMAN	: FILE MANAGER		(A110)	36006A
T	FILOAD	: LOADS A FILE FROM THE TELETYPE		(A110)	36010C
T	FILREA	: REENTERS THE DATA TAPE DUMPED BY FILDUM		(A110)	36011A
*T	FILRPT	: REPORTS FILE CONTENTS AND STRUCTURE		(F110)	36247A
T	FILSOR	: FILE POINTER SORT/POINTER SORT		(A107)	36206A
*T	FINDIT	: INFORMATION RETRIEVAL SYSTEM		(C102)	36250B
*T	FISHER	: FISHER'S EXACT PROBABILITY TEST		(F407)	36606A
T	FLCOPY	: COPIES ONE FILE INTO ANOTHER		(A110)	36012
T	FNCTS	: COMPUTES TRIG FUNCTIONS FOR COMPLEX ARGUMENTS		(A303)	36017
T	FORMAT	: ALLOWS SPECIAL FORMATTING OF DATA PRINTOUT		(A104)	36005C
T	FPLOT	: FUNCTION PLOT		(A904)	36112A
H	FREQ	: BAR GRAPHS OF GRADES		(S720)	36341A
T	FRQ	: FREQUENCY BETWEEN BOUNDARIES		(A408)	36191B
T	FTBL	: FOOTBALL		(A903)	36181A
E	GAME	: CLASSIC MATRIX OF GAME THEORY		(AH01)	36265A
H	GAMGN	: PROCESS OF GAMETOGENESIS		(S833)	36302A
S	GANOVA	: ANALYSIS OF VARIANCE (2-WAY)		(C410)	36501A
S	GCHLIN	: RATING INVESTMENT FUNDS		(C713)	36503A
S	GCPATH	: CRITICAL PATH ANALYSIS		(C603)	36504A
S	GCPM1	: CRITICAL PATH ANALYSIS		(C603)	36505A
S	GDAPI	: ABNORMAL PERFORMANCE INDEX		(C713)	36507A
S	GDPA	: EFFICIENT "CORNER" PORTFOLIOS		(C713)	36508A
*H	GENE1	: GENETICS SIMULATION		(F833)	36642A
T	GEOMEN	: STATISTICS OF GEOMETRIC DISTRIBUTION		(A408)	36045A
T	GFFT	: GENERAL FAST FOURIER TRANSFORM		(A316)	36030A
S	GFNRAT	: FINANCIAL RATIOS		(C712)	36510A
S	GINTLP	: LINEAR PROGRAMMING-VARIABLES RESTRICTED TO VALUES OF ONE OR ZERO		(C605)	36512A
S	GIPRPV	: INVESTMENT RETURN (CASH FLOW)		(C712)	36513A
S	GKASSF	: WARRANT PRICE CALCULATION		(C713)	36514A
S	GKCOST	: PRICE/EARNINGS RATIO CALCULATION		(C713)	36515A
S	GLP	: LINEAR PROGRAMMING		(C605)	36516A
S	GLPSA1	: LINEAR PROGRAMMING TWO-PHASE SIMPLEX METHOD		(C605)	36517A
T	GMARKS	: GSB GRADING PROGRAM		(C720)	36518A
S	GMCR01	: FISCAL POLICY GAME		(C830)	36519A
S	GMCR02	: FISCAL POLICY GAME		(C830)	36519A
S	GMCR05	: ECONOMIC POLICY GAME		(C830)	36521A
S	GMCR06	: ECONOMIC POLICY GAME		(C830)	36521A
T	GMEDIT	: GSB GRADING PROGRAM		(C720)	36518A
S	GMRGR	: SECURITIES PORTFOLIO ANALYSIS AND DETERMINATION		(C713)	36530A
S	GNETFL	: NETWORK FLOW		(C604)	36529A
S	GNMRV8	: SECURITIES PORTFOLIO ANALYSIS AND DETERMINATION		(C713)	36530A
T	GNPSUM	: GROSS NATIONAL PRODUCT SUMMARY		(A714)	36086A
T	GOLF	: A GOLF GAME		(A903)	36102A
H	GRADE	: TEST GRADE FOR NUMBER OF QUESTIONS MISSED		(S720)	36342A
S	GRANK	: RANKING STATISTICS		(C408)	36541A
T	GRAPHS	: DEMO PLOT PROGRAM FOR HP 7200 PLOTTER		(A904)	36115A
S	GRGPLT	: SIMPLE REGRESSION AND PLOT		(C404)	36542A
S	GRISKA	: RISK ANALYSIS IN CAPITAL INVESTMENT		(C712)	36543A
S	GSIMEQ	: SIMULTANEOUS LINEAR EQUATIONS		(C314)	36547A
S	GSPMG	: STANFORD PORTFOLIO MANAGEMENT GAME		(C880)	36502A
S	GSSS	: SMALL SYSTEMS SIMULATOR		(C607)	36552A
S	GSTKVL	: STOCK VALUATION		(C713)	36545A
T	GTAPID	: PAPER TAPE TITLER		(C104)	36548A
S	GTASPD	: SUBJECTIVE PROBABILITY DISTRIBUTION		(C406)	36549A
S	GTHOR	: SECURITIES EPS GROWTH		(C713)	36553A
S	GVOTE	: COMMITTEE CHOICE ANALYSTS		(C606)	36550
S	GVPOQT	: PLOTTING DATA		(C713)	3655
S	GWBULL	: SUBJECTIVE PROBABILITY - RANDOM VALUES		(C405)	3655
E	H-LIFE	: HALF LIFE SIMULATION		(A833)	36134A
T	HELLO	: TYPES DATE, TIME, AND PORT NUMBER ON TERMINAL		(H104)	36125B
T	HISS	: SAMPLE STATISTICS AND HISTOGRAM FORMED FROM A SET OF NUMBERS		(C904)	36235A
T	HISTOG	: A HISTOGRAM FORMED FROM A SET OF NUMBERS		(A408)	36055B
T	HPLOTR	: AUTOMATIC PLOTTING PROGRAM FOR THE HP 2000B		(B904)	72055A
T	HPMLIT	: LIST/DUMP HP ASSEMBLER FILES		(A108)	36218A

* Indicates NEW or REVISED since February 1973
Stripe indicates SUPPORTED program

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

ORDER NUMBER LISTING

LEGEND
Volumes: T = HP BASIC, E = Education, H = Huntington, S = Stanford
System Specifications: A = 2000A, B = 2000B, C = 2000C, F = 2000F, S = Single Terminal

Volume	Order No.	System Specification	Class Code	Name	Title
E	20879B	(A801)		MATHB	: CAI MATH PACKAGE (2000B VERSION)
E	24241B	(C800)		IDF	: INSTRUCTIONAL DIALOGUE FACILITY
E	24242B	(C800)		EXEC	: CAI EXECUTIVE PACKAGE
E	24243B	(C801)		MATHC	: CAI MATH PACKAGE (2000C VERSION)
E	24244A	(A800)		SUBRTN	: CAI SUBROUTINES
T	36001A	(A408)		LOGRAM	: LOG-ON TAPE ANALYZER
E	36002B	(A880)		SETUP	: MANGEMENT SIMULATION GAME FOR THE HP 2000A
E	36002B	(A880)		MANAG	: MANGEMENT SIMULATION GAME FOR THE HP 2000A
E	36002B	(A880)		RITE	: MANAGEMENT STMULATION GAME FOR THE HP 2000A
T	36003A	(A101)		MACRO	: A TEXT AND FILE PROCESSING SYSTEM
T	36004A	(A102)		SLSMN	: SALES MANAGER'S INFORMATION RETRIEVAL SYSTEM
T	36005B	(A104)		FORMAT	: ALLOWS SPECIAL FORMATTING OF DATA PRINTOUT
T	36006A	(A110)		FILMAN	: FILE MANAGER
T	36007A	(A110)		FILIN	: KEYBOARD FILE LOADING PROGRAM
T	36008C	(A110)		FILDUM	: PAPER TAPE FILE DUMP
T	36009C	(A110)		FILIST	: LISTS FILE CONTENTS BY RECORD NUMBER
T	36010C	(A110)		FILOAD	: LOADS A FILE FROM THE TELETYPE
T	36011A	(A110)		FILREA	: REENTERS THE DATA TAPE DUMPED BY FILDUM
T	36012B	(A110)		FLCOPY	: COPIES ONE FILE INTO ANOTHER
T	36013A	(A217)		TESTTY	: TESTS THE OPERATION OF A REMOTE TELETYPE
E	36014A	(A810)		STOP	: TIES MATH CALCULAIOR PACKAGE
T	36015A	(A302)		CALC	: EXTENDED PRECISION CALCULATOR
*T	36016B	(A302)		LRGNUM	: LARGE NUMBER ADDITION & MULTIPLICATION
T	36017A	(A303)		FNCTS	: COMPUTES TRIG FUNCTIONS FOR COMPLEX ARGUMENTS
T	36018A	(A303)		SQR(Z)	: COMPUTES SQUARE ROOI OF A COMPLEX NUMBER
T	36019A	(A306)		BESSEL	: CALCULATES BESSEL FUNCTION OF FIRST KIND
T	36021A	(A310)		INTGRT	: COMPUTES NUMERIC INTEGRAL OF A FUNCTION
*T	36022B	(A310)		ROMINT	: INTEGRATES A FUNCTION (ROMBERG METHOD)
T	36023B	(A404)		POLFIT	: FITS LEAST-SQUARES POLYNOMIALS
T	36024A	(A311)		ROOTER	: FINDS THE ROOTS OF POLYNOMIALS
T	36025A	(A312)		CDETER	: COMPUTES VALUE OF COMPLEX DETERMINANT
T	36026A	(A314)		SIMEQN	: SOLVES SIMULTANEOUS LINEAR EQUATIONS
T	36027A	(A314)		CPOUTI	: SOLVES SIMULTANEOUS LINEAR EQUATIONS
T	36028A	(A316)		CTRFFT	: COMPLEX TO REAL FAST FOURIER TRANSFORM
T	36029A	(A316)		RTCFFT	: REAL TO COMPLEX FAST FOURIER TRANSFORM
T	36030A	(A316)		GFFT	: GENERAL FAST FOURIER TRANSFORM
T	36031A	(A317)		DERIV	: DERIVATIVE OF A FUNCTION AT A POINT
T	36032A	(A318)		DE-1OR	: 1ST ORDER DIFFERENTIAL EQUATION
T	36033A	(A318)		DE-2OR	: 2ND ORDER DIFFERENTIAL EQUATION
T	36034A	(A301)		SPHERE	: SOLVES SPHERICAL TRIANGLES
T	36035B	(A301)		EUCLID	: COMPUTES LARGEST COMMON FACTOR OF TWO INTEGERS
T	36036A	(A301)		SQPS	: AN INTEGER AS THE SUM OF FOUR SQUARES
T	36037A	(A301)		FACTOR	: FINDS PRIME FACTORS OF POSITIVE INTEGERS
T	36038C	(A404)		CURFIT	: PERFORMS LEAST-SQUARES FIT
T	36039B	(A410)		ANVAR1	: ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN
T	36040B	(A410)		ANVAR2	: ANALYSIS OF VARIANCE (LATIN SQUARE DESIGN)
T	36041A	(A406)		BINOPO	: PROBABILITY DISTRIBUTION COMPARISONS
T	36042A	(A407)		CHISQ	: COMPUTES PROBABILITY OF CHI-SQUARE VALUES
T	36043B	(A407)		CHISQS	: CHI-SQUARE STATISTICS FOR M*N CONTINGENCY TABLES
T	36045A	(A408)		GEOMEN	: STATISTICS OF GEOMETRIC DISTRIBUTION
T	36050B	(A405)		RANDEV	: GENERATES RANDOM DEVIATES-SUBROUTINE
T	36052A	(A407)		STAT2	: MANN-WHITNEY 2 SAMPLE RANK TEST
I	36053A	(A407)		STAT3	: SPEARMAN RANK CORRELATION COEFFICIENTS
T	36054B	(A404)		REGCOR	: REGRESSION/CORRELATION
T	36055B	(A408)		HISTOG	: A HISTOGRAM FORMED FROM A SET OF NUMBERS
T	36056A	(A513)		ANALAD	: CIRCUIT ANALYSIS
T	36057A	(A513)		ACNODE	: AC CIRCUIT ANALYSIS PROGRAM
T	36058A	(A514)		HTXFR	: TWO DIMENSIONAL HEAT TRANSFER
T	36059A	(A516)		DEBYE	: COMPUTES DEBYE OR EINSTEIN FUNCTION
T	36060A	(A513)		LPFLTR	: DESIGNS LOW-PASS FILTERS
T	36061A	(A511)		LQVALV	: COMPUTES LIQUID CONTROL VALVE COEFFICIENTS
T	36062A	(A513)		MICRO	: MICROWAVE PARAMETERS CONVERSION
T	36063A	(A519)		TMFCEV	: TIME FUNCTION EVALUATOR
T	36064A	(A513)		MIXSPR	: MIXER SPURIOUS RESPONSE PROGRAM
T	36065C	(A606)		DECSN	: IOP MANAGEMENT DECISION GAME
T	36067A	(A610)		DYNPRO	: DYNAMIC PROGRAMMING MODEL
T	36068B	(A605)		LINPRO	: LINEAR PROGRAMMING MODEL
T	36069A	(A608)		LNTRND	: LINEAR TREND FORECASTING
T	36072A	(A720)		SALSIM	: SALARY SCHEDULE COST SIMULATOR
T	36073A	(A716)		BUDGET	: DEPARTMENTAL MANAGER'S BUDGETING PROGRAM
T	36074A	(A712)		ANNUIT	: ANNUITY ANALYSIS
T	36075A	(A712)		BALSHT	: PROFORMA INCOME STATEMENT AND BALANCE SHEET
T	36076B	(A713)		BNDPRC	: BOND PRICE ANALYSIS
T	36077A	(A713)		BNDSWH	: BOND SWITCH ANALYSIS
T	36078B	(A713)		BNDYLD	: BOND YIELD ANALYSIS
T	36079A	(A713)		BNKRSV	: BANK RESERVE CALCULATIONS
T	36080A	(A713)		CAPINV	: CAPITAL INVESTMENT ANALYSIS
T	36082A	(A712)		DEPCOM	: DEPRECIATION METHOD COMPARISON
T	36083A	(A712)		EQUITY	: COST OF EQUITY CAPITAL
T	36084A	(A713)		EXDRSK	: EXTENDED RISK ANALYSIS
T	36085A	(A713)		EXSMOO	: EXPONENTIAL SMOOTHING ON PRICE DATA
T	36086A	(A714)		GNPSUM	: GROSS NATIONAL PRODUCT SUMMARY
T	36087A	(A714)		INACNT	: NATIONAL INCOME & PRODUCT ACCOUNTS
T	36088A	(A714)		IN/OUT	: INPUT/OUTPUT ANALYSIS ON ECONOMIC FLOWS
T	36089B	(A712)		INSTMT	: INCOME STATEMENT
T	36090A	(A712)		LENDER	: SIMPLE LOAN ANALYSIS
T	36091A	(A712)		LESSEE	: LEASE ANALYSIS AS DETERMINED BY THE LESSEE
T	36092A	(A713)		MARKOW	: SECURITIES PORTFOLIO USING MARKOWITZ MODEL
T	36093A	(A712)		MKBUY	: MAKE-BUY DECISION ANALYSIS
T	36094B	(A712)		MORGAG	: MORTGAGE ANALYSIS
T	36095A	(A710)		SALES	: SALES COMMISSION REPORT
T	36096A	(A713)		STKINC	: STOCK MERGER INCENTIVE PROGRAM
T	36098A	(A713)		STKRTN	: STOCK RETURNS REPORT
T	36099A	(A713)		STKSMO	: EXPONENTIAL SMOOTHING AS A STOCK GUIDE
T	36100A	(A713)		STKVAL	: STOCK VALUE & EVALUATION REPORT
T	36101A	(A713)		TRUINT	: TRUE ANNUAL INTEREST RATE ANALYSIS
T	36102A	(A903)		GOLF	: A GOLF GAME
T	36103A	(A903)		BLJACK	: GAME OF TWENTY-ONE
T	36104A	(A904)		PLOT	: PLOTS A GIVEN FUNCTION ON THE TELETYPE
T	36105A	(A903)		BATNUM	: PLAYS THE BATTLE OF NUMBERS GAME
T	36106A	(A903)		QUBIC	: PLAYS 3-DIMENSIONAL TIC-TAC-TOE
T	36107A	(A903)		SLOT	: SIMULATES A SLOT MACHINE (USE IN NEVADA ONLY)
T	36108A	(A903)		SNOOPY	: DRAWS SNOOPY ON THE TELEPRINTER
T	36109A	(A515)		BEMDES	: RECOMMENDS CORRECT STEEL BEAM USE
T	36110A	(A901)		CANNON	: HP 7200A GAME DEMO FOR TIME-SHARED BASIC
T	36111A	(A716)		EXPEND	: BUDGET EXPENDITURES VS. TARGETS MONITOR
T	36112A	(A904)		FPLOT	: FUNCTION PLOT
T	36113B	(A717)		RECRUT	: RECRUTING INFORMATION SYSTEM
T	36114A	(A904)		TITLE	: CHARACTER GENERATION
T	36115A	(A904)		GRAPHS	: DEMO PLOT PROGRAM FOR HP 7200 PLOTTER
T	36116A	(A211)		BASTES	: BASIC TEST PROGRAM
T	36117A	(B104)		DATES	: COMPUTES DATE FROM SYSTEM CLOCK
T	36118A	(A303)		CXARTH	: VECTOR ARITHMETIC
T	36119A	(A303)		CXEXP	: VECTOR EXPONENTIATION
T	36120A	(A408)		FC	: ANALYSIS OF LOG TAPE
E	36121A	(A833)		ORBIT	: INTEGRATES EQUATIONS OF MOTION
T	36122A	(C107)		SORT	: FILE SERIAL STRING SORT
T	36125B	(B104)		HELLO	: TYPES DATE, TIME, AND PORT NUMBER ON TERMINAL
E	36128A	(B801)		TENS	: DRILL IN MULTIPLICATION OF NUMBERS WITH TRAILING ZEROS
E	36129A	(B801)		DRILL	: BASIC ARITHMETIC DRILL
E	36130A	(B801)		TWQUES	: MATHEMATICAL GAME OF TWENTY QUESTIONS
T	36131A	(B301)		CALCOM	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 1 OF 2)
T	36131A	(B301)		CALPLT	: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT (PART 2 OF 2)
E	36133A	(A801)		WKSHT	: GENERATES MATH WORKSHEETS
E	36134A	(A833)		H-LIFE	: HALF LIFE SIMULATION
T	36136A	(A408)		SCORES	: COMPUTES MEAN, STANDARD DEVIATION AND STANDARD SCORES FOR TEST SCORES
I	36137A	(A408)		KR20	: ITEM ANALYSIS AND KUDER-RICHARDSON FORMULA 20 RELIABILITY
I	36138A	(A903)		LIFE	: A GAME OF CELLULAR GENERATION GROWTH
E	36139A	(A810)		TUTOR	: TUTOR SERIES BASIC LANGUAGE PROGRAMMING COURSE, IUT01 - TUT25
I	36142B	(A712)		CSHFL	: CASH FLOW ANALYSIS
I	36143C	(C211)		XREF	: BASIC LANGUAGE PROGRAM CROSS-REFERENCE GENERAIOR
T	36144A	(A302)		EXTPRE	: 40-DIGIT PRECISION MATHEMATICS
I	36145A	(C107)		STGSRT	: SORTS STRINGS FROM FILES
E	36148A	(A820)		EXPNTL	: CAI IN SIMPLE EXPONENTIAL FUNCTIONS OF TIME
E	36149A	(A820)		SOLVER	: SOLVES COMPLEX SIMULTANEOUS EQUATIONS
E	36150A	(A820)		NET-3	: COMPLEX NUMBER OPERATIONS
E	36152A	(A820)		AC-1	: COMPUTER AIDED PRACTICE IN EE AC ANALYSIS
E	36153A	(A820)		COMPLX	: CAI IN ALGEBRA OF COMPLEX NUMBERS
E	36154A	(A820)		INDMTR	: ANALYSIS OF A BALANCED POLYPHASE INDUCTION MOTOR
E	36155A	(A820)		POLAR	: PLOTS SINGLE VARIABLE IN POLAR FORM
E	36156A	(A820)		DVDRS	: CAI IN VOLTAGE AND CURRENT DIVIDERS
E	36157A	(A820)		UNITS	: CAI IN INTERPRETATION OF EE UNITS
E	36158A	(A820)		RNET	: NETWORK SIMULTANEOUS EQUATIONS
E	36159A	(A801)		DIFFEQ	: CAI IN SOLUTION OF LINEAR FIRST-ORDER DIFFERENTIAL EQUATIONS
T	36164B	(C718)		IATA.1	: CALCULATE AIR FREIGHT RATES
T	36165A	(A904)		FGRAPH	: SIMULTANEOUS FUNCTION GRAPHER
E	36166A	(A820)		DETERM	: MATRIX DETERMINANT USING GAUSSIAN ELIMINATION
E	36167A	(A820)		JACOBI	: EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX
T	36168A	(A301)		CALC2	: THREE REGISTER CALCULATOR PROGRAM
T	36169A	(A901)		FIFTY!	: EXTENDED PRECISION FACTORIAL
T	36170A	(A401)		T-TEST	: TEST OF HYPOTHESES USING STUDENTS T DISTRIBUTION
T	36171B	(A603)		CPATH	: CRITICAL PATH EVALUATION
T	36172A	(A410)		ANVAR3	: ANALYSIS OF VARIANCE FOR A TWO VARIABLES OF CLASSIFICATION DESIGN
T	36173A	(A410)		ANVAR4	: TWO-WAY ANALYSIS OF VARIANCE FOR A TWO-WAY EXPERIMENT
T	36174A	(C718)		TRCK.1	: CALCULATE TRUCK FREIGHT RATES
E	36175A	(C871)		CARLOS	: COMPUTER-ASSISTED REVIEW LESSONS ON SYNTAX FOR SPANISH II
T	36176A	(A104)		STGINT	: STRING-INTEGER CONVERSIONS
T	36177A	(A708)		REP	: DATA CENTER INVENTORY REPORT GENERATOR
I	36177A	(A708)		IN	: DATA CENTER INVENTORY REPORT GENERATOR
T	36178A	(A404)		MULREG	: MULTIPLE REGRESSION/CORRELATION
T	36179A	(C713)		DROIPB	: DISCOUNTED RETURN ON INVESTMENT AND PAYBACK
I	36180A	(S509)		SUNSET	: SUNRISE-SUNSET PREDICTOR
T	36181A	(A903)		FTBL	: FOOTBALL
E	36182A	(A801)		SAT	: TRIGONOMETRIC SOLUTIONS OF TRIANGLES
E	36184A	(A833)		SPCTRA	: OPTICAL ABSORPTION SPECTRA SIMULATION, 2-SPECIES EQUILIBRIUM MIXTURES
T	36185A	(C205)		TERTES	: TELETYPE-CRT DIAGNOSTIC
T	36186A	(A404)		MULTX	: LEAST-SQUARES FIT, MULTIPLE Y'S PER X
E	36187A	(B860)		WHEELS	: AUTO PURCHASE AND MAINTENANCE SYSTEM
I	36188A	(S309)		POLY	: POLYNOMIAL APPROXIMATION
T	36191B	(A408)		FRQ	: FREQUENCY BETWEEN BOUNDRIES
E	36193A	(B880)		SETUPB	: MANAGEMENT SIMULATION GAME FOR THE 2000B
E	36193A	(B880)		MANAGB	: MANAGEMENT SIMULATION GAME FOR THE 2000B
E	36193A	(B880)		MANAGU	: MANAGEMENT SIMULATION GAME FOR THE 2000B
E	36193A	(B880)		RITET	: MANAGEMENT SIMULATION GAME FOR THE 2000B
T	36194A	(C712)		LEASIN	: LEASE INCOME
T	36196A	(A314)		SOLVIT	: SIMULTANEOUS LINEAR EQUATIONS USING GAUSSIAN REDUCTION
T	36197A	(A108)		JULIAN	: JULIAN CALENDAR FOR THE CURRENT YEAR
E	36199A	(A880)		MRKSIM	: MARKETING SIMULATION

* Indicates NEW or REVISED since February 1973

Stripe indicates SUPPORTED program

ORDER NUMBER LISTING

T 36200A (C101) TIDE : TEXT EDITOR FOR THE 2000C
E 36202A (C720) ATTEND : ARSENTEE LISTING
T 36203A (A405) RANDOM : RANDOM NUMBER GENERATOR
T 36204A (A101) TIDEX : SYMBOLIC FILE EDITOR
E 36205A (A801) CADA1I : COMPUTER ASSISTED ARITHMETIC DRILL
E 36205A (A801) CADA1M : COMPUTER ASSISTED ARITHMETIC DRILL
E 36205A (A801) CADA1P : COMPUTER ASSISTED ARITHMETIC DRILL
T 36206A (A107) FILSOR : FILE POINTER SORT/POINTER SORT
T 36206A (A107) POISOR : FILE POINTER SORT/POINTER SORT
E 36208A (A810) STOPH : MATH CALCULATOR PACKAGE
E 36209A (C880) SFTUPC : MANAGEMENT SIMULATION GAME FOR THE 2000C
E 36209A (C880) MANAGC : MANAGEMENT SIMULATION GAME FOR THE 2000C
T 36210A (A708) CTC1 : CTC MANUFACTURING PARTS CONTROL
T 36211A (A708) CTC2 : CTC INVENTORY CONTROL FOR FINISHED PRODUCTS
*T 36212A (A717) CTC3 : CTC PROJECTION PROGRAMS
*T 36213A (A706) CTC4 : CTC PAYROLL PROGRAMS
T 36214A (A711) CTC5 : CTC ACCOUNTS RECEIVABLE
T 36216A (A108) HPMLOT : LIST/DUMP HP ASSEMBLER FILES
T 36218A (A108) HPMLTT : LIST/DUMP HP ASSEMBLER FILES
E 36219A (C810) BRAIN : BRAIN SIMULATOR PROGRAM
T 36220A (C104) CHARS : ASCII CHARACTER SET
T 36221A (B110) XTRACT : FILE UTILITY PROGRAM
T 36221A (B110) TAPDUM : FILE UTILITY PROGRAM
T 36222A (A108) TALK : TIME SHARING SYSTEM COMMUNICATION
T 36226A (C712) LOAN : LOAN AMORTIZATION
E 36227A (A801) FACTRL : FACTORIALS TO 1000 PLACES
T 36228A (A904) PLOTWD : WORD PLOTTER
T 36230A (A605) TRANSP : TRANSPORTATION PROBLEM
T 36231A (A102) ADDRES : ADDRESS LABELS
T 36232A (C107) IRV : FILE SORT ROUTINE
E 36233A (A880) LABOR : LABOR/MANAGEMENT BARGAINING
T 36235A (C904) HISS : SAMPLE STATISTICS AND HISTOGRAM FORMED FROM A SET OF NUMBERS
T 36236A (A903) HATTLE : BATTLESHIP GAME
E 36237A (C801) TRIFAC : FACTORING QUADRATIC TRINOMIALS
E 36238A (C801) SIMPAC : PRACTICE WITH SIGNED NUMBERS
E 36239A (C801) POLSUB : POLYNOMIAL SUBTRACTION
E 36240A (C801) BISQAR : SQUARING BINOMIALS
E 36241A (C710) IATA.C : CALCULATES BREAKPOINT OF IATA CONTAINERS
T 36243A (C903) STTHI : STAR TREK
T 36244A (A903) KENO : $25,000.00 KENO GAME
T 36246A (A309) POLFTE : FITS LEAST-SQUARES POLYNOMIALS
*T 36247A (F110) FILRPT : REPORTS FILE CONTENTS AND STRUCTURE
*T 36248A (F108) NEWIDS : CREATES NEW USER NUMBERS
*T 36250B (C102) FINDIT : INFORMATION RETRIEVAL SYSTEM
E 36251A (A801) LOGIC : LOGIC EXAMINATION PROGRAMS
E 36253A (C801) EXTEND : INFINITE PRECISION MATH UTILITY PROGRAM
E 36254A (C810) EXPRES : EXPRESSION SOLVING PROGRAM
T 36256B (C104) ASCII : CREATES AN ASCII FILE CONTAINING ALL 256 ASCII CHARACTERS
T 36257A (C104) -ASCII : ASCII CODE GENERATOR
T 36258A (A104) KEYSIG : GIVES MAJOR SCALES
T 36259A (A104) MUSICS : FINDS DOMINANT SEVENTHS
E 36260A (A833) IONIC : DRILL ON FORMULAS AND CHARGES OF IONS
E 36261A (A833) ISOMER : DRILL ON NAMING ALKANES
E 36262A (A801) CXSYSS : SOLVING SYSTEMS OF COMPLEX LINEAR EQUATIONS
T 36263A (A312) DETER4 : DETERMINANTS, CHARACTERISTIC POLYNOMIALS AND INVERSES OF MATRICES
T 36264A (B405) RNDORD : PLACING INTEGERS IN RANDOM ORDER
E 36265A (A801) GAME : CLASSIC MATRIX OF GAME THEORY
E 36266A (P833) IONIC1 : DRILL ON FORMULAS OF IONIC COMPOUNDS
E 36267A (A833) ELMENT : DRILL ON SYMBOLS FOR CHEMICAL ELEMENTS
*T 36271A (F410) ANOVA3 : THREE FACTORIAL ANALYSIS OF VARIANCE
T 36272A (C108) FILIS : FILE LISTING PROGRAM/INSTRUCTIONS
T 36272A (C108) FILIS1 : FILE LISTING PROGRAM/INSTRUCTIONS
E 36276A (A850) MUSIC2 : TRIAD SOLVING PROGRAM
E 36277A (C801) DIMIS : X-Y AXIS SEGMENT PROGRAM
*E 36278A (C801) SYSSOL : SOLVING SYSTEMS OF LINEAR EQUATIONS
*T 36279A (F108) JULI : ONE PAGE JULIAN CALENDAR
*T 36282A (F720) COLREG : COLLEGE REGISTRATION DEMO
*T 36284A (F102) MESSAG : INTERTERMINAL COMMUNICATOR
*T 36287A (F212) DATA : DUMPS FILE TO DATA STATEMENTS
*T 36288A (F108) CALNDR : PRINTS A CALENDAR
*T 36292A (A107) ALPHA : ALPHANUMERIC SORT
*T 36293A (F513) ACTFIL : ACTIVE FILTER DESIGN
*T 36294A (A410) ANCOV : ANALYSIS OF COVARIANCE
*E 36295A (F833) CHEM : SELF-CORRECTING CHEMISTRY TEST
*T 36296A (F104) ALFTOV : ALPHA TO VARIABLE CONVERSION
*T 36297A (F104) TIMER : TIME OF THE DAY
*T 36298A (F104) DATER : DATE AND DAY OF THE WEEK
*T 36299A (F904) PRINT : GENERATES LARGE LETTERS
H 36300A (S833) DROS : GENETIC CHARACTERISTICS
H 36301A (S833) EVOLU : NATURAL SELECTION EXPERIMENT
H 36302A (S833) GAMGN : PROCESS OF GAMETOGENESIS
H 36303A (S833) NZYMC : ENZYMATIC REACTION RATES
H 36304A (S833) NZYM2 : ENZYME REACTION RATE
H 36305A (S833) PHOSYN : PHOTOSYNTHESIS EXPERIMENT
H 36306A (S833) DECAY1 : RADIOACTIVE DECAY GAME
H 36307A (S833) DECAY2 : NUCLEAR DECAY
H 36308A (S833) EQUIL1 : EQUILIBRIUM SYSTEMS
H 36308A (S833) EQUIL2 : EQUILIBRIUM SYSTEMS
H 36309A (S833) KINET : KINETIC REACTION
H 36310A (A833) MASSD : MASS DEFECT
H 36311A (S833) PRCNT : PERCENT COMPOSITION
H 36312A (S833) BFIELD : MAGNETIC FIELD PICTURE
H 36313A (S833) BOHR : ENERGY LEVEL DIAGRAM
H 36314A (S833) CALORI : CALORIMETRY EXPERIMENT
H 36315A (S833) KINERV : REVIEW OF KINEMATICS
H 36316A (S833) NEWTN2 : NEWTON'S 2ND LAW
H 36317A (S833) PHOTEL : PHOTOELECTRIC EFFECT
H 36318A (S833) PHOTON : ENERGY LEVEL PROBLEM
H 36319A (S833) REFLCT : LEAST TIME PRINCIPLE AND LIGHT
H 36320A (S833) SNELL : SNELL'S LAW
H 36321A (S833) SPACE : SPACECRAFT ORBITS
H 36322A (S833) VFIELD : POTENTIAL FIELD PICTURE
H 36323A (S833) VLOCTY : INSTANTANEOUS VELOCITY
H 36324B (S833) WAVES : SUM OF TWO WAVES
H 36325A (S833) CLOUDS : CLOUD FORMATION
H 36326A (S833) WATER1 : WATER BUDGET PROBLEM CHECK
H 36327A (S833) WATER2 : WATER BUDGET
H 36328A (S860) BALANC : TRADE AND PAYMENT BALANCES
H 36329A (S860) BANK : SOLVES FINANCIAL PROBLEMS
H 36330A (S860) CONSMR : DEPRESSION/EQUILIBRIUM
H 36331A (S860) STOCK : STOCK MARKET SIMULATION
H 36333A (S801) CRVLEN : COMPUTES LENGTH OF ANY CURVE
H 36334A (S810) PLOTTP : FUNCTION PLOTTER
H 36335A (S810) QUADT : NATURE OF GRAPH OF A 2ND ORDER EQUATION IN TWO VARIABLES
H 36336A (S801) RATIO : SOLVES PROPORTIONS
H 36337A (S801) SETS : UNION AND INTERSECTION OF SETS
H 36338A (S801) SURFAC : AREA OF SURFACE OF REVOLUTION
H 36339A (S801) VOLSOL : VOLUME OF SOLID OF REVOLUTION
H 36340A (S720) AVERG1 : AVERAGES AND CURVES GRADES
H 36341A (S720) FREQ : BAR GRAPHS OF GRADES
H 36342A (S720) GRADE : TEST GRADE FOR NUMBER OF QUESTIONS MISSED
H 36343A (S720) ITEM1 : NUMBER FREQUENCIES
H 36344A (S720) STAT : STATISTICAL ANALYSIS OF LAB DATA
H 36350A (S833) PLANK : A PHOTOELECTRIC SIMULATION
H 36351A (A833) SLITS : YOUNG'S DOUBLE SLIT EXPERIMENT
E 36353A (S833) AFLIFE : JOHN CONWAY'S GAME OF LIFE
E 36360A (S801) HAGELS : THREE-DIGIT NUMBER GUESSING
S 36501A (C410) GANOVA : ANALYSIS OF VARIANCE (2-WAY)
S 36502A (C880) GSPMG : STANFORD PORTFOLIO MANAGEMENT GAME
S 36503A (C713) GCHLIN : RATING INVESTMENT FUNDS
S 36504A (C603) GCPATH : CRITICAL PATH ANALYSIS
S 36505A (C603) GCPM1 : CRITICAL PATH ANALYSIS
S 36507A (C713) GDAPI : ABNORMAL PERFORMANCE INDEX
S 36508A (C713) GDPA : EFFICIENT "CORNER" PORTFOLIOS
S 36510A (C712) GFNRAT : FINANCIAL RATIOS
S 36512A (C605) GINTLP : LINEAR PROGRAMMING-VARIABLES RESTRICTED TO VALUES OF ONE OR ZERO
S 36513A (C712) GIRRPV : INVESTMENT RETURN (CASH FLOW)
S 36514A (C713) GKASSF : WARRANT PRICE CALCULATION
S 36515A (C713) GKCOST : PRICE/EARNINGS RATIO CALCULATION
S 36516A (C605) GLP : LINEAR PROGRAMMING
S 36517A (C605) GLPSAI : LINEAR PROGRAMMING TWO-PHASE SIMPLEX METHOD
T 36518A (C720) GMARKS : GSB GRADING PROGRAM
T 36518A (C720) GMEDIT : GSB GRADING PROGRAM
S 36519A (C830) GMCR01 : FISCAL POLICY GAME
S 36519A (C830) GMCR02 : FISCAL POLICY GAME
S 36521A (C830) GMCR06 : ECONOMIC POLICY GAME
S 36521A (C830) GMCR05 : ECONOMIC POLICY GAME
S 36529A (C604) GNETFL : NETWORK FLOW
S 36530A (C713) GNPVR : SECURITIES PORTFOLIO ANALYSIS AND DETERMINATION
S 36530A (C713) GNPGB : SECURITIES PORTFOLIO ANALYSIS AND DETERMINATION
S 36541A (C408) GRANK : RANKING STATISTICS
S 36542A (C404) GRGPLT : SIMPLE REGRESSION AND PLOT
S 36543A (C712) GRISKA : RISK ANALYSIS IN CAPITAL INVESTMENT
S 36545A (C713) GSTKVL : STOCK VALUATION
S 36547A (C314) GSIMEQ : SIMULTANEOUS LINEAR EQUATIONS
T 36548A (C104) GTAPID : PAPER TAPE TITLER
S 36549A (C406) GTASPD : SUBJECTIVE PROBABILITY DISTRIBUTION
S 36550A (C606) GVOTE : COMMITTEE CHOICE ANALYSIS
S 36551A (C405) GWHULL : SUBJECTIVE PROBABILITY - RANDOM VALUES
S 36552A (C607) GSSS : SMALL SYSTEMS SIMULATOR
S 36553A (C713) GTHOR : SECURITIES EPS GROWTH
S 36554A (C713) GVPOUT : PLOTTING DATA
S 36554A (C713) VPDU : PLOTTING DATA
S 36555A (C904) VCHART : INVESTMENT DECISIONS USING TEKTRONIX 4010
S 36556A (C904) VREGPL : PLOTTING X AND Y VARIABLES USING TEKTRONIX 4010
S 36557A (C713) VRRC : INVESTMENT STRATEGY ANALYSIS
S 36558A (C904) VSUR : DISPLAY ROUTINE USING TEKTRONIX 4010
T 36559A (C904) VTTT : TIC-TAC-TOE ON THE TEKTRONIX 4010 DISPLAY TERMINAL
*T 36601A (A903) DRAG : SIMULATES A DRAG RACE
*E 36602A (F801) TRUTH : TRUTH TABLES FOR BOOLEAN EXPRESSIONS
*E 36603A (F833) ALERA : PROPAGATION OF ERROR
*E 36604A (F860) CHARGE : CHARGE ACCOUNT SIMULATION
*T 36606A (F407) FISHER : FISHER'S EXACT PROBABILITY TEST
*T 36607A (F410) STAT19 : KRUSKAL-WALLIS ONE WAY ANALYSIS OF VARIANCE
*H 36611A (A833) MEMBR : DIFFUSION EXPERIMENT
*H 36613A (A833) ATWT : CALCULATES ATOMIC WEIGHT
*H 36614A (A833) AVOGA : AVOGADRO'S NUMBER
*H 36615A (A833) EMPIR : CALCULATES EMPIRICAL FORMULAS
*H 36616A (A833) MOLAR : ACID-BASE TITRATION
*H 36617A (A833) PHPOH : PH, POH, PCT. DISSOCIATION
*H 36618A (A833) STOICH : MASS VOLUME
*H 36620A (A801) CVAREA : AREA UNDER CURVE
*H 36622A (A801) LIMSIN : LIMIT OF (SIN X)/X
*H 36623A (A801) PI : CALCULATES PI
*H 36625A (A801) ROOTS2 : QUADRATIC EQUATION SOLVER
*H 36626A (A801) SLOPE : FINDS DERIVATIVES
*H 36627A (A801) SQRT : FINDS SQUARE ROOT
*H 36628A (A801) STATAL : ARITHMETIC MEAN
*H 36629A (A833) EFIELD : ELECTRIC FIELD STRENGTH
*H 36630A (A833) LENSES : SOLVES LENS PROBLEMS
*H 36631A (A833) PRJTL : PROJECTILE MOTION
*H 36632A (A830) CIRFLW : CIRCULAR FLOW MODEL
*T 36635A (F514) METRIC : CONVERTS ENGLISH TO METRIC
*T 36638A (F709) CTC6 : CTC ACCOUNTS PAYABLE
*H 36639A (F833) POLUT : WATER POLLUTION SIMULATION
*H 36640A (F860) POLSYS : SIMULATION OF CITY COUNCIL
*H 36641A (F833) STERL : FLY POPULATION CONTROL
*H 36642A (F833) GENE1 : GENETICS SIMULATION
T 72050A (B904) HPLOTR : AUTOMATIC PLOTTING PROGRAM FOR THE HP 2000R
T 72051A (C904) HPPLOT : AUTOMATIC PLOTTING PROGRAM

* Indicates NEW or REVISED since February 1973
Stripe indicates SUPPORTED program

FINDIT
C102-36250B

TITLE: INFORMATION RETRIEVAL SYSTEM

DESCRIPTION:

The Information Retrieval System is used with the HP 2000C and 2000F. It provides the on-line user with the ability to create, update, and interrogate data files from one or more terminals. Once a file is created, records may be added, modified, and deleted. Any record in the file is available to be printed in a variety of forms such as lists, tabulated reports, or address labels. Records may be retrieved by comparing their contents to a set of file search conditions.

Additional programs permit calculations to be performed on numerical file data, output listings to be ordered on any element, and multi-level file sorts to be performed.

INSTRUCTIONS:

A. File Structure

1. Master file (up to 10,000 records)
2. Auxiliary file (32-record scratch file)
3. Gate file (4-record scratch file)
4. DO file (2-record scratch file)

B. Program Descriptions

1.	CREATE	Used to initially define the master file structure and passwords, and to subsequently modify the file structure.
2.	UPDATE	Used to add, modify, and delete records in the master file.
3.	SEARCH	Retrival program which allows data output in a variety of formats.
4.	FINDIT	Driver program for SEARCH and UPDATE, requiring a password for access to them.
5.	CAL	Calculator program which permits computation and high-precision sums to be made on numerical file data during SEARCH operations.
6.	FIND1	File sort program which permits output listings to be ordered in any element.
7.	IRV1	File sort program which permits multi-level file sorts to be executed.
8.	FINDOR	Driver program for FIND1 and IRV1.

Complete User Instructions are contained in the FINDIT manual HP 36250, option D00.

SYSTEM SPECIFICATIONS: 2000C and Teletype

SPECIAL CONSIDERATIONS:

This system is designed for a data file of 10,000 records or less. Each record may contain approximately 425-500 characters. Element field width is variable. In a typical installation it is recommended that FINDIT be CSAVed and PROtected, except for the program CAL, which should be SAVed and not PROtected. For this reason, listings are not included in this Handbook.

PRICE

Not available on paper tape. To order a magnetic tape of all contributed BASIC programs, order HP Part No. 02000-90029 for 2000C, or HP Part No. 02000-90060 for 2000F, Price $25.00.
User's Manual, HP Part No. 36250A, Option D00, Price $5.00

ACKNOWLEDGEMENTS: Irv Brenner

RUN

```
SEARCH CONDITIONS:
? STATE=CA
? STATE>R
? STATE=OH
? CLASS<55
? CLASS>62
? cr
  PRINT OPTION? SPECIAL
    ELEMENTS, FIELD WIDTH:
?   ID,4
?   CLASS,5
?   STATE,5
?   NAME,1
?   cr


ID   CLASS STATE NAME

997  43    CA    BELAIR, R. JAMES
2274 51    OH    BURDETT, HENRY
2991 63    TE    BURNS, VIRGINIA
650  24    CA    THOMAS, JOHN F.

TOTAL RECORDS= 4




RUN
FINDIT

FILE: XXXXXX
PASSWORD: XXXXXX
**INVALID PASSWORD.
PASSWORD: XXXXXX
**INVALID PASSWORD.
PASSWORD: XXXXXX
**INVALID PASSWORD.
PASSWORD: XXXXXX
SEARCH CONDITIONS:
? MARITAL=S
?
  PRINT OPTION? SPECIAL
    ELEMENTS, FIELD WIDTHS:
?   NAME,20
?   CITY,10
?   MARITAL,1
?   DEGREE,3
?   CLASS,1
?


NAME                 CITY        M DEG CLASS

ALTMAN, LEO S.       CHICAGO     S BBA 37
BRENNAN,RICHARD R.   WASHINGTON  S MBA 53
BURDETT, HENRY       CLEVELAND   S BBA 51
  **2**
BURNS, VIRGINIA      DALLAS      S BJ  63
JACKSON, MILTON      CHICAGO     S BM  49
PRICE, HAROLD        CHICAGO     S BS  49
  **1**
SANTIN, ANDRE        PARIS       S BA  55
TAO, KENNETH         DENNIS      S BA  67
WHITE, SANDRA R.     CLEVELAND   S BFA 61
WILLIS, DONALD L.    WASHINGTON  S BA  51
YOUNG, REMINGTON     CHICAGO     S BS  49

TOTAL RECORDS= 11


AGAIN? YES
```

```
SEARCH CONDITIONS:
? CITY=CHICAGO
? MARITAL=S
?
  PRINT OPTION? SPECIAL
    ELEMENTS, FIELD WIDTHS:
?   NAME,20
?   CITY,10
?   DEGREE,3
?   CLASS,1
?


NAME                 CITY       DEG CLASS

ALTMAN, LEO S.       CHICAGO    BBA 37
JACKSON, MILTON      CHICAGO    BM  49
PRICE, HAROLD        CHICAGO    BS  49
YOUNG, REMINGTON     CHICAGO    BS  49

TOTAL RECORDS= 4


AGAIN? YES


SEARCH CONDITIONS:
? STATE=IL
? STATE=OH
? STATE=TE
? MARITAL=S
?
  PRINT OPTION? SPECIAL
    ELEMENTS, FIELD WIDTHS:
?   NAME,20
?   CITY,10
?   STATE,5
?   DEGREE,3
?   CLASS,1
?


NAME                 CITY       STATE DEG CLASS

ALTMAN, LEO S.       CHICAGO    IL    BBA 37
BURDETT, HENRY       CLEVELAND  OH    BBA 51
  **2**
BURNS, VIRGINIA      DALLAS     TE    BJ  63
JACKSON, MILTON      CHICAGO    IL    BM  49
PRICE, HAROLD        CHICAGO    IL    BS  49
WHITE, SANDRA R.     CLEVELAND  OH    BFA 61
YOUNG, REMINGTON     CHICAGO    IL    BS  49

TOTAL RECORDS= 7


AGAIN? NO

DONE
```

MESSAG

TITLE: INTERTERMINAL COMMUNICATOR F102-36284A

DESCRIPTION:

This program allows messages to be entered at one user terminal and to be received anytime afterward at another user terminal. Messages are self-dating and include the sender's and receiver's names. It is useful, for example, when a number of schools use the same computer, as it provides a fast and simple means of sending printed messages from school to school.

INSTRUCTIONS:

Open a file named MSFILE in a semi-privileged user code (i.e., one that begins with the letter A) to which all users who are to use the program have access. The program contains buffer handling routines to prevent any confusion when two or more users are writing on the file at the same time. The size of the file is variable from 2 to 128 records. Suggested starting sizes:

For 256-word records (i.e., 2000C or 2000C'/F): 20 records.
For 64-word records (i.e., 2000B): 60 records.

The program compensates for varying number of records and varying record sizes.

The program has four options selected by number:

STOP (option number 0): Stops the program.

RECEIVE MESSAGE (option number 1): The user selects the message to be printed by its number (a listing of available messages is given at the beginning of the RUN and when option number 3 is selected). After the message is printed the user may have the option of deleting the message from the file. (The program will not allow any user except the sender of the message to remove it from the file within five minutes after it is entered. This prevents one user from deleting a message being entered concurrently by another user.)

ENTER MESSAGE (option number 2): The user inputs the receiver's and his names. The program assigns the message a number and labels it with the current data and time. The user inputs his message using as many lines as needed and types the word END for the last input to stop.

RECEIVE LIST OF AVAILABLE MESSAGES (option number 3): This option gives a listing of currently available messages. It is automatically selected at the beginning of the RUN.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

RUN

```
RUN
STOP
RUN
MESSAG

INTERTERMINAL COMMUNICATOR


THERE ARE MESSAGES FOR:

   1. BOB COLLINS
   2. EVERYONE
   3. J BENTLY S.H.S
   4. ANYBODY


OPTION?  Ø = STOP, 1 = RECEIVE MESSAGE, 2 = ENTER MESSAGE,
3 = RECEIVE LIST OF AVAILABLE MESSAGES - ?1

TYPE THE NUMBER OF THE MESSAGE YOU WANT. - ?1


FOR: BOB COLLINS     FROM: D. MCCARTNEY     3/29/73     2:54 AM

BOB, I'VE COPIED YOUR PROGRAM FOR EXPERIMENT 15 FROM B102 AND WILL
HAVE MY STUDENTS BEGIN WORK ON IT TOMORROW.  I HAVE FOUND AN ERROR IN
EXPT14.  USING DATA OF 1,2,.54,.737 I OBTAIN 'SUBSCRIPT OUT OF BOUNDS
IN LINE 1750' BEFORE THE FOURTH ITERATION IS PRINTED.
END OF MESSAGE.  DO YOU WANT IT REMOVED FROM THE FILE?YES


OPTION?2
THIS MESSAGE IS FOR?DON MCCARTNEY
AND IS FROM?BOB COLLINS

ENTER MESSAGE (ANY NUMBER OF LINES, 3 MINUTE TIME LIMIT PER
LINE).  TYPE  END  FOR LAST LINE TO STOP INPUT.

? THANKS, DON.  I HAVE ALREADY FOUND THE ERROR IN EXPT14.  I HAVE SAVED
? THE CORRECTED VERSION IN THE B100 LIBRARY.
? END

OPTION?1

TYPE THE NUMBER OF THE MESSAGE YOU WANT. - ?2


FOR: EVERYONE     FROM: EASTERN H.S.     3/29/73     2:56 AM

ANNOUNCING A NEW LIBRARY PROGRAM:  TRIG

TRIG IS A DRILL PROGRAM OF TRIGONOMETRIC ANGLES.  EXAMPLE QUESTIONS
ARE 'WHAT IS THE COSINE OF 45 DEGREES?'  STUDENT RESPONSE IS
'SQR(2)/2' OR 'WHAT ANGLE HAS .5 AS ITS SINE?'  STUDENT MAY RESPOND
EITHER '30 DEGREES' OR '150 DEGREES'.  TO USE IT GET AND RUN
$TRIG.  INSTRUCTIONS ARE GIVEN.
END OF MESSAGE.  DO YOU WANT IT REMOVED FROM THE FILE?NO


OPTION?Ø

DONE
```

LISTING

```
MESSAG

1  REM  ****  HP BASIC PROGRAM LIBRARY  *********************************
2  REM
3  REM         MESSAG:   INTERTERMINAL COMMUNICATOR
4  REM
5  REM         36284 REV  A   6/73
6  REM
```

```
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  REM MESSAG - INTERTERMINAL COMMUNICATOR
20  DIM M[127],M$[72],T$[14],F$[14],X$[10],Z$[6],T[127],G$[72]
30  FILES MSFILE
40  DEF FNW(X)=INT((X+3)/2)
50  X$="0123456789"
70  D0=180
80  PRINT "INTERTERMINAL COMMUNICATOR"
90  REM DETERMINE NUMBER OF RECORDS IN FILE
100  IF  END #1 THEN 140
110  FOR R0=1 TO 128
120  READ #1,R0
130  NEXT R0
140  R0=R0-1
150  IF R0#1 THEN 180
160  PRINT "FILE TOO SMALL - MUST BE AT LEAST 2 RECORDS"
170  STOP
180  REM DETERMINE NUMBER OF WORDS PER RECORD
190  READ #1,R0
200  IF  END #1 THEN 240
210  FOR W0=0 TO 254 STEP 2
220  PRINT #1;0
230  NEXT W0
240  GOSUB 1630
250  F1=0
260  PRINT "OPTION";
270  IF F1 THEN 310
280  PRINT "?  0 = STOP, 1 = RECEIVE MESSAGE, 2 = ENTER MESSAGE,"
290  PRINT "3 = RECEIVE LIST OF AVAILABLE MESSAGES - ";
300  F1=1
310  INPUT F2
320  IF F2<0 OR F2>3 OR F2#INT(F2) THEN 250
330  GOTO F2+1 OF 1980,340,1020,1520
340  PRINT "TYPE THE NUMBER OF THE MESSAGE YOU WANT, - ";
350  INPUT M1
360  F0=0
370  FOR R1=1 TO R0-1
380  READ #1,R1
390  IF TYP(-1)#1 THEN 770
400  READ #1;M0
410  IF M1#M0 THEN 770
420  READ #1,R1
430  IF F0 THEN 730
440  F0=1
450  READ #1;M0,T$,F$,T0,T1,T2,T3,M$
460  T9=T2
470  PRINT "FOR: ";T$;"    FROM: ";F$;"    ";
480  RESTORE 1970
490  FOR T4=1 TO 12
500  READ T5
510  T5=T5+(T4=2 AND T3/4=INT(T3/4))
520  IF T5 >= T2 THEN 550
530  T2=T2-T5
540  NEXT T4
550  Z0=10000*T4+100*T2+T3
560  GOSUB 1850
570  PRINT Z$[1,Z1-3];"/";Z$[Z1-2,Z1-1];"/";Z$[Z1,Z1+1];"    ";
580  Z0=100*(T1-12*INT((T1-1)/12))+T0
590  GOSUB 1850
600  PRINT Z$[1,Z1-1];":";Z$[Z1,Z1+1];
610  IF T1=0 AND T0=0 THEN 700
620  IF T1=12 AND T0=0 THEN 680
630  IF T1<12 THEN 660
640  PRINT " PM"
650  GOTO 710
660  PRINT " AM"
670  GOTO 710
680  PRINT " NOON"
690  GOTO 710
700  PRINT " MIDNIGHT"
710  PRINT
720  GOTO 750
730  IF TYP(-1)=4 THEN 770
740  READ #1;M0,T0,T1,M$
750  PRINT M$
760  GOTO 430
770  NEXT R1
780  IF F0 THEN 840
```

```
790  PRINT "THERE IS NO MESSAGE NUMBER";M1
800  PRINT "POSSIBLY IT HAS BEEN REMOVED FROM THE FILE SINCE THE LAST"
810  PRINT "LISTING OF AVAILABLE MESSAGES WAS PRODUCED.  OPTION 3 WILL"
820  PRINT "GIVE A LISTING OF AVAILABLE MESSAGES."
830  GOTO 260
840  T=60*TIM(1)+TIM(0)-60*T1-T0
845  T[M1]=0
850  IF T >= 0 AND T <= 4 AND T9=TIM(2) AND  NOT T[M1] THEN 1000
860  PRINT "END OF MESSAGE.  DO YOU WANT IT REMOVED FROM THE FILE":
870  INPUT Y$[1,1]
880  IF Y$="N" THEN 260
890  IF Y$#"Y" THEN 860
900  FOR R1=1 TO R0-1
910  READ #1,R1
920  IF TYP(-1)#1 THEN 960
930  READ #1;M0
940  IF M0#M1 THEN 960
950  PRINT #1,R1
960  NEXT R1
970  T[M1]=0
980  READ #1,R0
990  GOTO 260
1000  PRINT "MESSAGE ENTERED RECENTLY - POSSIBLY NOT COMPLETE"
1010  GOTO 260
1020  R2=0
1030  PRINT "THIS MESSAGE IS FOR":
1040  INPUT T$
1050  PRINT "AND IS FROM":
1060  INPUT F$
1070  PRINT "ENTER MESSAGE (ANY NUMBER OF LINES, 3 MINUTE TIME LIMIT PER"
1080  PRINT "LINE).  TYPE  END  FOR LAST LINE TO STOP INPUT."
1090  F3=0
1100  PRINT "? ":
1110  ENTER D0,D1,M$
1120  IF D1#-256 THEN 1150
1130  PRINT "TIME OUT - INPUT TERMINATED"
1140  GOTO 260
1150  IF M$="END" THEN 260
1160  IF F3 THEN 1360
1170  REM FIND AN AVAILABLE MESSAGE NUMBER
1180  MAT M=ZER
1190  FOR R1=1 TO R0-1
1200  READ #1,R1
1210  IF TYP(-1)#1 THEN 1240
1220  READ #1;M0
1230  M[M0]=1
1240  NEXT R1
1250  FOR M1=1 TO 127
1260  IF  NOT M[M1] THEN 1290
1270  NEXT M1
1280  GOTO 1500
1290  GOSUB 1540
1300  IF R2=0 THEN 1500
1310  PRINT #1,R2;M1,T$,F$,TIM(0),TIM(1),TIM(2),TIM(3),M$
1320  W1=10+FNW(LEN(T$))+FNW(LEN(F$))+FNW(LEN(M$))
1330  F3=1
1340  T[M1]=1
1350  GOTO 1480
1360  IF W1+6+FNW(LEN(M$)) <= W0 THEN 1400
1370  GOSUB 1540
1380  IF R2=0 THEN 1500
1390  W1=0
1400  READ #1,R2
1410  GOTO TYP(-1) OF 1420,1440,1460,1460
1420  READ #1;G0
1430  GOTO 1410
1440  READ #1;G$
1450  GOTO 1410
1460  PRINT #1;M1,TIM(0),TIM(1),M$
1470  W1=W1+6+FNW(LEN(M$))
1480  READ #1,R0
1490  GOTO 1100
1500  PRINT "FILE FULL - LAST LINE LOST - INPUT TERMINATED"
1510  GOTO 260
1520  GOSUB 1630
1530  GOTO 260
1540  REM FIND AN AVAILABLE RECORD
1550  FOR R1=R2+1 TO R0-1
```

```
1560  READ #1,R1
1570  IF TYP(-1)#1 THEN 1610
1580  NEXT R1
1590  R2=0
1600  RETURN
1610  R2=R1
1620  RETURN
1630  REM PRINT LISTING OF AVAILABLE MESSAGES
1640  MAT M=ZER
1650  F0=0
1660  FOR R1=1 TO R0-1
1670  READ #1,R1
1680  IF TYP(-1)#1 THEN 1730
1690  READ #1;M0
1700  IF M[M0] THEN 1730
1710  F0=1
1720  M[M0]=R1
1730  NEXT R1
1740  IF F0 THEN 1770
1750  PRINT "THERE ARE NO MESSAGES."
1760  RETURN
1770  PRINT "THERE ARE MESSAGES FOR:"
1780  FOR M1=1 TO 127
1790  IF  NOT M[M1] THEN 1830
1800  READ #1,M[M1];Z0,M$
1810  GOSUB 1850
1820  PRINT TAB(3-Z1);Z$;". ";M$
1830  NEXT M1
1840  RETURN
1850  REM NUMBER FORMAT SUBROUTINE
1860  IF Z0 <= 1 THEN 1890
1870  Z1=INT(LOG(Z0)/LOG(10))
1880  GOTO 1900
1890  Z1=0
1900  Z$=""
1910  FOR Z2=Z1 TO 0 STEP -1
1920  Z3=INT(Z0/10↑Z2)
1930  Z$[LEN(Z$)+1]=X$[Z3+1,Z3+1]
1940  Z0=Z0-Z3*10↑Z2
1950  NEXT Z2
1960  RETURN
1970  DATA 31,28,31,30,31,30,31,31,30,31,30,31
1980  END
```

ASCII*
C104-36256B

TITLE:

CREATES AN ASCII FILE CONTAINING ALL 256 ASCII CHARACTERS.

DESCRIPTION:

The program "ASCII*" fills the file named "ASCII" with the 256 characters of the ASCII character set. These characters are contained in four 64 character strings, two in record one, and two in record two. The characters are in ASCII order, from lowest to highest.

INSTRUCTIONS:

Open the file "ASCII" two records. The file should have 256 word records. Run the program.

```
GET-ASCII*
OPEN-ASCII,2
RUN
ASCII*

DONE
```

If the file is in the library, it should then be sanctified if possible. The program"-ASCII", (HP36257) creates a list of the characters which are put in the file "ASCII".

SYSTEM SPECIFICATIONS:

2000C and Teletype

SPECIAL CONSIDERATIONS:

This program will work only on a 2000C system. When the file "ASCII" is opened in the library, be sure its records are 256 word records if you plan on using an Aardwolf & Company Writing Team program which uses "ASCII".

The Aardvark and Company Writing Team has designed programs to take up an absolute minimum of computer storage and perform a maximum purpose. The team encourages people to send good programs to Aardvark. As a slight encouragement, the team will give anyone who sends a program which is accepted a free "subscription" to the program handbook and include the contributor as a member of the writing team.

ACKNOWLEDGEMENTS:

Aardvark and Company
2130 Bell Court
Lakewood, Colorado 80215

RUN

```
RUN
ASCII*

DONE
```

LISTING

```
ASCII*

1   REM  ****  HP BASIC PROGRAM LIBRARY  ******************************
2   REM
3   REM         ASCII*:  CREATES AN ASCII FILE CONTAINING ALL 256 ASCII
4   REM                  CHARACTERS
5   REM
6   REM         36256  REV B   6/73
7   REM
8   REM  ****  CONTRIBUTED PROGRAM  **********************************
10   REM  *** AARDWOLF & COMPANY WRITING TEAM ***  PHILIP J. TUBB
20   REM  *** ASCII* *** 0335 WORDS *** VERSION 7 ***
30   E$="ASCII"
40   READ O1,O2,O3,J1
50   DATA 1,2,3,32
60   DIM A$[64],B$[64],C$[64],D$[64],E$[5]
70   FILES *,ASCII
80   A$=" !#$%&'()*+,-./0123456789:;<=>?"
90   B$="@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑←"
100   ASSIGN E$,O1,B,"!!"
110   PRINT #O1;A$,B$
120   ASSIGN E$,O1,B,F$
130   READ #O1,O1;C$,D$
140   B$[J1]=D$[J1]
150   A$[O3,O3]=C$[O3]
160   PRINT #O1,O1;A$,B$
170   ASSIGN E$,O1,B,"0000"
180   READ #O1,O1;C$,D$
190   C$[J1+O1]=A$
200   B$[J1+O1]=D$
210   PRINT #O2,O1;C$,B$
220   ASSIGN E$,O1,B,"@@@@@@@"
230   PRINT #O1,O2;B$,C$
240   READ #O1,O1
250   READ #O2,O2;A$,B$
260   PRINT #O2;"FOR MORE INFORMATION WRITE:""THE AARDWOLF AND
      COMPANY WRITING TEAM""2130 BELL COURT""LAKEWOOD, COLO.  80215"
270   END
```

ALFTOV
F104-36296A

TITLE: ALPHA TO VARIABLE CONVERSION

DESCRIPTION:

This subroutine will convert a numeric value contained within an alpha string into a number stored in a variable. Blanks are ignored, and any number, positive or negative, integer, decimal, or even written with an "E" exponent will be converted. Conversion stops when a non-valid character is encountered.

INSTRUCTIONS:

The main program must dimension A$, which is the input string. The index of the character in A$ at which the conversion is to start is stored in ZO. The value is returned in the variable Z.

Input Parameters:	A$,	the string containing the numeric value
	ZO,	the index of A$ to begin conversion
Output Parameters:	Z,	the value of the numeric quantity
	ZO,	the index of the first non-valid character or the end of A$

Entry Point: 9010

Variables Used: Z$(15),(internally dimensioned), Z1, Z2, Z3, Z4, Z5, Z8, and Z9

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

The length of the subroutine is 433 words. This may be shortened to 282 by removing the unnecessary REM statements.

Overflow or underflow errors will occur if the numeric quantity is outside the limits of the machine.

ACKNOWLEDGEMENTS:

Lawrence E. Turner, Jr.
Pacific Union College

RUN

```
LIS
TEST

10  DIM A$[72]
20  INPUT A$
30  LET Z0=5
40  GOSUB 9010
50  LET X=Z
60  LET Z0=Z0+1
70  GOSUB 9010
80  PRINT "VALUES ARE: ";X;Z
90  GOTO 20

APP-ALFTOV
0-9999 END

RUN
TEST

?AAA=-45.6,78.9006E-7
VALUES ARE: -45.6         7.89006E-06
?AAA=13,-56
VALUES ARE:  13   -56
?AAA=0  7  8  ;-  45E+2
VALUES ARE:  78   -4500
?
DONE
```

LISTING

```
ALFTOV

9000  REM   ***** SUBROUTINE ALFTOV *****
9001  REM
9002  REM      CONVERTS AN ALPHA NUMBER TO A VARIABLE
9003  REM
9004  REM  INPUT:  A$, INPUT STRING
9005  REM          Z0, INDEX TO BEGIN SCAN
9006  REM  OUTPUT: Z,  RETURNED VALUE
9007  REM          Z0, INDEX AFTER LAST VALID CHARACTER
9008  REM
9009  DIM Z$[15]
9010  REM
9011  LET Z=Z4=Z5=0
9012  LET Z1=Z2=Z3=1
9013  LET Z$="0123456789+-.E "
9015  FOR Z9=Z0 TO LEN(A$)
9016  FOR Z8=1 TO 15
9017  IF A$[Z9,Z9]=Z$[Z8,Z8] THEN 9020
9018  NEXT Z8
9019  GOTO 9080
9020  IF Z8=15 THEN 9070
9021  IF Z8>10 THEN 9040
9022  GOTO Z2 OF 9025,9030,9035
9025  LET Z=10*Z+Z8-1
9028  GOTO 9070
9030  LET Z5=Z5+1
9031  GOTO 9025
9035  LET Z4=10*Z4+Z8-1
9037  GOTO 9070
9039  REM
9040  GOTO Z8-10 OF 9045,9045,9055,9060
9045  IF Z2<3 AND Z>0 THEN 9080
9046  IF Z2=3 THEN 9050
9047  LET Z1=23-2*Z8
9048  GOTO 9070
9050  IF Z4>0 THEN 9080
9051  LET Z3=23-2*Z8
9052  GOTO 9070
9055  IF Z2>1 THEN 9080
9056  LET Z2=2
9057  GOTO 9070
```

```
9060  IF Z2=3 THEN 9080
9061  LET Z2=3
9070  NEXT Z9
9075  REM
9080  LET Z0=Z9
9081  IF ABS(Z3*Z4-Z5)<50 THEN 9085
9082  LET Z5=Z3*(Z4-50)
9085  LET Z=Z1*Z*10↑(Z3*Z4-Z5)
9087  RETURN
9090  REM    ***** END ALFTOV *****
```

TIMER
F104-36297A

TITLE: TIME OF THE DAY

DESCRIPTION:

This subroutine returns the time of day in a convenient alpha format.

INSTRUCTIONS:

The time is returned in a string array T$ internally dimensioned for 10 characters.

Output Parameter: T$, the time of day

Entry Point: 9910

Variables Used: T$(10), Z$(10), Z1, Z2, Z3, and Z8

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

The length of this subroutine is 390 words. This may be reduced to 285 words by deleting the REM statements.

If this subroutine is used in conjunction with DATER, the DIM statement is line 9906 should be deleted. Other than this, the two subroutines are perfectly compatible.

ACKNOWLEDGEMENTS:

Lawrence E. Turner, Jr.
Pacific Union College

RUN

```
LIS
TEST

10  GOSUB 9910
20  PRINT T$
30  STOP

APP-TIMER
9999 END

RUN
TEST

11:31 AM

DONE
```

LISTING

TIMER

```
1   REM  ****  HP BASIC PROGRAM LIBRARY  **********************************
2   REM
3   REM          TIMER:   TIME OF THE DAY
4   REM
5   REM          36297 REV  A    6/73
6   REM
7   REM  ****  CONTRIBUTED PROGRAM  ***************************************
9900  REM  ***** SUBROUTINE TIMER *****
9901  REM
9902  REM  RETURNS THE CURRENT TIME OF DAY IN ALPHA FORMAT
9903  REM  OUTPUT PARAMETER: T$
9904  REM
9905  DIM T$[10]
9906  DIM Z$[10]
9907  REM
9908  REM   *** ENTRY 9910 ***
9909  REM
9910  LET Z$="0123456789"
9911  LET Z1=TIM(0)
9912  LET Z2=TIM(1)
9915  IF Z1>0 THEN 9925
9916  IF Z2#0 THEN 9920
9917  LET T$="12:00 MIDN"
9918  RETURN
9920  IF Z2#12 THEN 9925
9921  LET T$="12:00 NOON"
9922  RETURN
9925  LET Z8=0
9926  IF Z2<12 THEN 9932
9927  IF Z2=12 THEN 9930
9928  LET Z2=Z2-12
9930  LET Z8=1
9932  IF Z2>0 THEN 9935
9933  LET Z2=12
9934  REM
9935  LET T$=" "
9936  LET Z3=INT(Z2/10)
9937  IF Z3=0 THEN 9940
9938  LET T$="1"
9940  LET Z3=Z2-Z3*10+1
9941  LET T$[2]=Z$[Z3,Z3]
9942  LET T$[3]=":"
9945  LET Z3=INT(Z1/10)
9946  LET T$[4]=Z$[Z3+1,Z3+1]
9947  LET Z3=Z1-Z3*10+1
9948  LET T$[5]=Z$[Z3,Z3]
9950  LET T$[6]=" AM  "
9951  IF Z8=0 THEN 9955
9952  LET T$[6]=" PM  "
9955  RETURN
9956  REM
9960  REM   ***** END TIMER *****
```

DATER

TITLE: DATE AND DAY OF THE WEEK F104-36298A

DESCRIPTION:

This is a subroutine to return the current date and day of the week in two alpha strings.

INSTRUCTIONS:

The current date is returned as: dd MON yy
where

dd	is an integer day of the month
MON	is a three letter abbreviation of the month
yy	is the last two digits of the year

The day of the week is returned as a three letter abbreviation.

Output Parameters: N$, the current date (internally dimensioned: N$(9))

M$, the current day of the week (internally dimensioned: M$(36), logical length is 3)

Entry Point: 9810

Variables Used: M$(36), n$(9), Z(12), Z$(10), Z1, Z2, Z3, and Z8

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

The length of the subroutine is 536 words. This may be shortened to 392 words by the deletion of the REM statements.

ACKNOWLEDGEMENTS:

Lawrence E. Turner, Jr.
Pacific Union College

```
RUN

LIS
TEST

10  GOSUB 9810
20  PRINT N$,M$
30  STOP

APP-DATER
9999 END

RUN
TEST

27 MAR 73       TUE

DONE
```

LISTING

```
DATER

9790  REM  ****  HP BASIC PROGRAM LIBRARY  ************************
9791  REM
9792  REM        DATER:  DATE AND DAY OF THE WEEK SUBROUTINE
9793  REM
9794  REM        36298  REV A  --  6/73
9795  REM
9796  REM  ****  CONTRIBUTED PROGRAM  ******************************
9800  REM  ***** SUBROUTINE DATER *****
9801  REM
9802  REM  RETURNS THE CURRENT DATE AND DAY OF THE WEEK IN ALPHA FORMAT
9803  REM  OUTPUT PARAMETERS:  N$. DATE
9804  REM                      M$. DAY OF WEEK
9805  REM
9806  DIM M$[36],N$[9],Z[12],Z$[10]
9807  REM
9808  REM   *** ENTRY 9810 ***
9809  REM
9810  MAT Z=CON
9811  MAT Z=(31)*Z
9812  LET Z[4]=Z[6]=Z[9]=Z[11]=30
9813  LET Z[2]=29
9815  LET M$="JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC"
9817  LET Z$="0123456789"
9820  REM
9821  LET Z1=TIM(3)
9822  IF Z1/4=INT(Z1/4) THEN 9825
9823  LET Z[2]=28
9825  LET Z8=TIM(2)
9830  FOR Z2=1 TO 12
9831  IF Z8 <= Z[Z2] THEN 9840
9832  LET Z8=Z8-Z[Z2]
9835  NEXT Z2
9838  REM
9840  LET Z3=INT(Z8/10)
9841  LET N$=" "
9842  IF Z3=0 THEN 9845
9843  LET N$=Z$[Z3+1,Z3+1]
9845  LET Z3=Z8-Z3*10+1
9846  LET N$[2]=Z$[Z3,Z3]
9847  LET N$[3]=" "
9848  LET N$[4]=M$[3*Z2-2,3*Z2]
9849  LET N$[7]=" "
9850  LET Z3=INT(Z1/10)
9851  LET N$[8]=Z$[Z3+1,Z3+1]
9852  LET Z3=Z1-Z3*10+1
9853  LET N$[9]=Z$[Z3,Z3]
9855  REM
9860  LET M$="SUNMONTUEWEDTHUFRISAT"
9861  LET Z3=TIM(2)+Z1+INT((Z1-1)/4)
9862  LET Z3=Z3-INT(Z3/7)*7
9863  LET M$=M$[3*Z3+1,3*Z3+3]
9865  RETURN
9868  REM
9870  REM      ***** END DATER *****
```

TITLE: ALPHANUMERIC SORT

ALPHA
A107-36292A

DESCRIPTION:

Sorts strings of any length by converting them to numbers and sorting their values.

INSTRUCTIONS:

Each alphanumeric string is entered on a separate line.

The matrix size can be adjusted if it is too large for the system or if more than 60 strings are being sorted. The first dimension is the number of strings, the second is the length in characters of the largest string.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

David Brown
Ridgewood Computer Club

```
RUN


RUN
ALPHA

HOW MANY WORDS DO YOU WANT TO ALPHABETIZE?17
PUT EACH WORD ON A LINE.
?JONES
?SMITH
?KAYES
?MOUNCE
?BROWN
?DITMARS
?FORD
?FEHRLE
?HAY
?HILTUNEN
?CHISHOLM
?DOUGHERTY
?HULLEY
?KALENA
?SAXMAN
?WHITE
?HURLEY



BROWN
CHISHOLM
DITMARS
DOUGHERTY
FEHRLE
FORD
HAY
HILTUNEN
HULLEY
HURLEY
JONES
KALENA
KAYES
MOUNCE
SAXMAN
SMITH
WHITE

DONE
```

ALPHA

```
1   REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2   REM
3   REM          ALPHA:   ALPHANUMERIC SORT
4   REM
5   REM          36292 REV  A   6/73
6   REM
7   REM  ****  CONTRIBUTED PROGRAM  ************************************
10   REM:   DAVID BROWN  PROGRAMMER
20   REM:   ALPHABETIZES STRINGS BY CONVERSION
30   REM:   TO PSEUDO EBCDIC CODE
40   DIM A[60,72],A$[72],S$[72]
50   S$="!#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]↑"
60   MAT A=ZER
70   PRINT "HOW MANY WORDS DO YOU WANT TO ALPHABETIZE";
80   INPUT L
90   PRINT "PUT EACH WORD ON A LINE."
100   FOR N=1 TO L
110   INPUT A$
120   FOR F=1 TO LEN(A$)
130   FOR X=1 TO LEN(S$)
140   IF A$[F,F]#S$[X,X] THEN 170
150   A[N,F]=X
160   GOTO 180
```

```
170  NEXT X
180  NEXT F
190  NEXT N
200  PRINT "";
210  Z=1
220  FOR X=1 TO L
230  IF A[X,1]=0 OR X=Z THEN 300
240  FOR M=1 TO LEN(S$)
250  IF A[Z,M]=0 OR A[Z,M]<A[X,M] THEN 300
260  IF A[Z,M]=A[X,M] THEN 290
270  Z=X
280  GOTO 220
290  NEXT M
300  NEXT X
310  FOR P=1 TO LEN(S$)
320  IF A[Z,P]=0 THEN 350
330  PRINT S$[A[Z,P],A[Z,P]];
340  NEXT P
350  A[Z,1]=0
360  PRINT
370  FOR Z=1 TO L
380  IF A[Z,1]=0 THEN 400
390  GOTO 220
400  NEXT Z
410  END
```

NEWIDS

TITLE: CREATES NEW USER NUMBERS F108-36248A

DESCRIPTION:

These programs use 1 file, IDFILE, and generate new user numbers over a stipulated range with 6 character passwords, all printing or all non-printing. User numbers (numeric portion) can range from 100 to 999. User numbers ending in 00 (group masters) are not produced. This program package contains 4 separate programs:

IDNOS - Generates new user numbers with passwords and places them in a file, IDFILE.

NEWIDS - Reads IDFILE and punches a tape to be used on system console for authorizing the new user numbers.

KILIDS - Punches a tape to kill the user numbers.

IDFORM - Prepares a form for assigning the user numbers to the users. Non-printing passwords are indicated by an ↑ at the end of the password.

INSTRUCTIONS:

Follow the instructions of each program.

1. OPEN - IDFILE,32 (max file size ever needed)
2. GET-IDNOS and RUN
3. GET the appropriate program and run - NEWIDS, KILIDS, or IDFORM.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

CTRL E and CTRL T may need to be removed from the non-printing character set if:

1. Teletype answer back responds to CTRL E.
2. Teletype is equipped to respond to CTRL T (tape punch off)

New user numbers should be authorized when system is idle or 5-10 spaces should be punched on the tape preceeding the NEW command. Don't forget about logon/logoff interruptions.

ACKNOWLEDGEMENTS:

Fred Stone
Tuskegee Institute

RUN

```
RUN
IDNOS

1 FOR PRINTING AND  2 FOR NON PRINTING PASSWORDS?2
ENTER FIRST AND LAST ID NUMBERS?100,110
ENTER ID LETTER?R

DONE


GET-NEWIDS
RUN
NEWIDS

ENTER MAX TIME AND SPACE?50000,100
START PUNCH UNIT

NEW-R101,,50000,100
NEW-R102,,50000,100
NEW-R103,,50000,100
NEW-R104,,50000,100
NEW-R105,,50000,100
NEW-R106,,50000,100
NEW-R107,,50000,100
NEW-R108,,50000,100
NEW-R109,,50000,100
NEW-R110,,50000,100


DONE



GET-IDFORM
RUN
IDFORM
```

ID PSWD	NAME	LOCAL ADDRESS
R101,KQDWHR †		
R102,RLIURP †		
R103,WLUVBH †		
R104,PEELIB †		
R105,WAFABA †		
R106,EBDVFH †		
R107,FIKDFZ †		
R108,RDVVHT †		
R109,QPRDIR †		
R110,HVHBFT †		

```
DONE
```

```
GET-KILIDS
RUN
KILIDS

START PUNCH UNIT

KIL-R101
KIL-R102
KIL-R103
KIL-R104
KIL-R105
KIL-R106
KIL-R107
KIL-R108
KIL-R109
KIL-R110


DONE
```

LISTING

```
IDNOS

1  REM  ****  HP BASIC PROGRAM LIBRARY  ****************************
2  REM
3  REM         NEWIDS:  CREATES NEW USER NUMBERS
4  REM         IDNOS, PART 1 OF 4
5  REM
6  REM         36248  REV A  6/73
7  REM
8  REM  ****  CONTRIBUTED PROGRAM  ***********************************
10  FILES IDFILE
20  DIM A$[6],B$[72],C$[26]
30  C$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
40  PRINT "1 FOR PRINTING AND  2 FOR NON PRINTING PASSWORDS";
50  INPUT C1
60  GOTO C1 OF 130,90
70  PRINT "YOU MUST ENTER 1 OR 2"
80  GOTO 40
90  RESTORE 470
100  READ B$
110  D2=18
120  GOTO 180
130  PRINT "HOW MANY CHARACTERS ARE TO BE USED";
140  RESTORE 450
150  READ B$
160  INPUT D2
170  IF D2<1 OR D2>58 THEN 410
180  D=1000/D2-1
190  PRINT "ENTER FIRST AND LAST ID NUMBERS";
200  INPUT F1,F2
210  IF F1<100 OR F2<100 OR F2<F1 THEN 430
220  PRINT "ENTER ID LETTER";
230  INPUT L$
240  FOR I=1 TO 26
250  IF L$=C$[I,I] THEN 290
260  NEXT I
270  PRINT L$;" IS NOT VALID"
280  GOTO 220
290  FOR I=F1 TO F2
300  I1=INT(I/100)
310  I1=I1*100
320  IF I=I1 THEN 390
330  FOR N1=1 TO 6
340  L1=INT(RND(I)*1000/D)
350  IF L1<1 OR L1>D2 THEN 340
360  A$[N1,N1]=B$[L1,L1]
370  NEXT N1
380  PRINT #1:L$,I,A$, END
390  NEXT I
400  STOP
410  PRINT "LIMIT CHARACTERS TO 58"
420  GOTO 130
430  PRINT "F1 MUST BE > F2 AND NEITHER < 100"
```

```
440  GOTO 190
450  DATA "ABCDEFGHIJKLMNOPQRSTUVWXYZ123456789!#$%&'()*:=-@[\+;↑]<>?/"
460  REM NEXT STAT ABDEFHIKLPQRTUVWYZ  FOR CONTROL CHARACTERS
470  DATA ""
480  END
```

NEWIDS

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ******************************
2  REM
3  REM         NEWIDS:  CREATES NEW USER NUMBERS
4  REM         NEWIDS, PART 2 OF 4
5  REM
6  REM         36248  REV A  6/73
7  REM
8  REM  ****  CONTRIBUTED PROGRAM  **********************************
10  FILES IDFILE
20  DIM A$[11]
30  PRINT "ENTER MAX TIME AND SPACE";
40  INPUT T,S
50  PRINT "START PUNCH UNIT"
60  PRINT "                                                    "
70  IF  END #1 THEN 120
80  READ #1;L$,I,A$
90  PRINT  USING 100;L$,I,A$,T,S
100  IMAGE "NEW-",A,3D,",",6A,",",5D,",",3D
110  GOTO 80
120  PRINT "                                                    "
130  END
```

IDFORM

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ******************************
2  REM
3  REM         NEWIDS:  CREATES NEW USER NUMBERS
4  REM         IDFORM, PART 3 OF 4
5  REM
6  REM         36248  REV A  6/73
7  REM
8  REM  ****  CONTRIBUTED PROGRAM  **********************************
10  FILES IDFILE
20  DIM C$[72],P$[72]
30  REM NEXT LINE CONTAINS 12 CTRL N
40  PRINT ""
50  READ P$
60  READ C$
70  DIM A$[6]
80  IF  END #1 THEN 320
90  L=0
100  PRINT "ID    PSWD        NAME";TAB(50);"LOCAL ADDRESS"
110  GOSUB 340
120  READ #1;L$,I,A$
130  IF A$[1,1]>" " THEN 230
140  FOR K2=1 TO 6
150  FOR K1=1 TO LEN(C$)
160  IF A$[K2,K2]#C$[K1,K1] THEN 180
170  A$[K2,K2]=P$[K1,K1]
180  NEXT K1
190  NEXT K2
200  PRINT  USING 210;L$,I,A$
210  IMAGE A,3D,",",6A," ↑"
220  GOTO 250
230  PRINT  USING 240;L$,I,A$
240  IMAGE A,3D,",",6A
250  GOSUB 340
260  L=L+1
270  IF L#25 THEN 120
280  REM NEXT LINE CONTAINS 12 CTRL N
290  PRINT ""
300  GOTO 90
310  REM NEXT LINE CONTAINS 12 CTRL N
320  PRINT ""
330  STOP
340  FOR I1=1 TO 69
```

```
350  PRINT "-";
360  IF I1=13 OR I1=48 THEN 380
370  GOTO 390
380  PRINT "I";
390  NEXT I1
400  PRINT "-"
410  RETURN
420  REM DATA FOR CONTROL CONVERSION IS NEXT
430  DATA "ABDEFHIKLPQRTUVWYZ"
440  REM CTRL CHARACTERS OF ABOVE LINE ARE NEXT
450  DATA ""
460  END
```

KILIDS

```
1   REM  ****  HP BASIC PROGRAM LIBRARY  **********************************
2   REM
3   REM         NEWIDS:  CREATES NEW USER NUMBERS
4   REM         KILIDS, PART 4 OF 4
5   REM
6   REM         36248  REV A  6/73
7   REM
8   REM  ****  CONTRIBUTED PROGRAM  ***************************************
10   FILES IDFILE
20   DIM A$[6]
30   PRINT "START PUNCH UNIT"
40   PRINT "                                                              "
50   IF  END #1 THEN 100
60   READ #1;L$,I,A$
70   PRINT  USING 80;L$,I
80   IMAGE "KIL-",A,3D
90   GOTO 60
100   PRINT "                                                            "
110   END
```

JULI
F108-36279A

TITLE: ONE PAGE JULIAN CALENDAR

DESCRIPTION:

This program generates a Julian Calendar for any year. The calendar is useful to keep near a Time Share Computer to obtain the current Julian day.

INSTRUCTIONS:

Run the program and enter complete year (such as 1973) where requested.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Paul Wittman
HP, Midwest Sales Region

RUN

```
RUN
JULI

THIS PROGRAM WILL GENERATE A CALENDAR OF DATES VERSUS
THE DAY OF THE YEAR
WHAT YEAR DO YOU WANT?1973
```

CALENDAR FOR YEAR 1973

DAY	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
1	1	32	60	91*	121	152	182*	213	244	274	305	335
2	2	33	61	92	122	153	183	214	245*	275	306	336*
3	3	34	62	93	123	154*	184	215	246	276	307	337
4	4	35*	63*	94	124	155	185	216	247	277	308*	338
5	5	36	64	95	125	156	186	217*	248	278	309	339
6	6	37	65	96	126*	157	187	218	249	279	310	340
7	7*	38	66	97	127	158	188	219	250	280*	311	341
8	8	39	67	98*	128	159	189*	220	251	281	312	342
9	9	40	68	99	129	160	190	221	252*	282	313	343*
10	10	41	69	100	130	161*	191	222	253	283	314	344
11	11	42*	70*	101	131	162	192	223	254	284	315*	345
12	12	43	71	102	132	163	193	224*	255	285	316	346
13	13	44	72	103	133*	164	194	225	256	286	317	347
14	14*	45	73	104	134	165	195	226	257	287*	318	348
15	15	46	74	105*	135	166	196*	227	258	288	319	349
16	16	47	75	106	136	167	197	228	259*	289	320	350*
17	17	48	76	107	137	168*	198	229	260	290	321	351
18	18	49*	77*	108	138	169	199	230	261	291	322*	352
19	19	50	78	109	139	170	200	231*	262	292	323	353
20	20	51	79	110	140*	171	201	232	263	293	324	354
21	21*	52	80	111	141	172	202	233	264	294*	325	355
22	22	53	81	112*	142	173	203*	234	265	295	326	356
23	23	54	82	113	143	174	204	235	266*	296	327	357*
24	24	55	83	114	144	175*	205	236	267	297	328	358
25	25	56*	84*	115	145	176	206	237	268	298	329*	359
26	26	57	85	116	146	177	207	238*	269	299	330	360
27	27	58	86	117	147*	178	208	239	270	300	331	361
28	28*	59	87	118	148	179	209	240	271	301*	332	362
29	29		88	119*	149	180	210*	241	272	302	333	363
30	30		89	120	150	181	211	242	273*	303	334	364*
31	31		90		151		212	243		304		365

SUNDAYS ARE SHOWN WITH AN *

```
DONE
```

LISTING

```
JULI

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         JULI:  ONE PAGE JULIAN CALENDAR
4  REM
5  REM         36279 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***********************************
8  REM  THIS PROGRAM WILL GENERATE A HANDY 1 PAGE JULIAN CALENDAR
9  REM  FOR USE WITH TIME SHARE SYSTEMS
10  REM WRITTEN BY PAUL WITTMAN ON 12/29/72   (364/72)
11  REM DIM Z(3)
12  PRINT "THIS PROGRAM WILL GENERATE A CALENDAR OF DATES VERSUS"
20  PRINT "THE DAY OF THE YEAR"
30  PRINT "WHAT YEAR DO YOU WANT";
40  INPUT Y
45  GOSUB 3000
50  DIM M[13],C$[10],D[13,31]
```

```
60  DATA 31,31,28,31,30,31,30,31,31,30,31,30,31
70  FOR C=1 TO 13
80  READ M[C]
90  NEXT C
91  FOR C=1 TO 31
92  D[1,C]=C
93  NEXT C
95  T=Y/100
97  FOR T1=1 TO 2
100  IF INT(100*T+.01)=INT(100*(INT(T)+.001)) THEN 110
102  T=Y/4
104  NEXT T1
106  GOTO 115
110  IF T1=1 THEN 115
111  M[3]=29
115  D1=0
120  FOR C=2 TO 13
130  FOR G=1 TO M[C]
140  D[C,G]=D1+G
150  NEXT G
160  FOR G=M[C]+1 TO 31
170  D[C,G]=0
190  NEXT G
200  D1=D1+M[C]
210  NEXT C
220  C$="1234567890"
230  DIM P$[64]
240  GOSUB 1000
250  PRINT
260  PRINT
270  PRINT
280  PRINT
290  PRINT TAB(18)"CALENDAR FOR YEAR"Y
300  PRINT
310  PRINT
311  DIM H$[39]
312  DATA "DAYJANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC"
313  READ H$
320  FOR H=1 TO 13
330  S=H*5-4
332  T=H*3-2
335  P$[S,S+2]=H$[T,T+2]
340  NEXT H
350  PRINT P$
360  PRINT
370  FOR C=1 TO 31
400  N=D[1,C]
410  GOSUB 2000
420  GOSUB 1000
440  IF Z[2]#0 THEN 455
450  P$[1,1]=" "
451  GOTO 470
455  P$[1,1]=C$[Z[2],Z[2]]
470  IF Z[3]#0 THEN 490
480  P$[2,2]="0"
485  GOTO 495
490  P$[2,2]=C$[Z[3],Z[3]]
495  FOR W=1 TO 12
500  N=D[W+1,C]
520  W1=W*5+1
522  IF N=0 THEN 670
530  GOSUB 2000
540  IF Z[1]#0 THEN 560
550  P$[W1,W1]=" "
555  GOTO 570
560  P$[W1,W1]=C$[Z[1],Z[1]]
570  IF Z[2]#0 OR Z[1]#0 THEN 590
580  P$[W1+1,W1+1]=" "
585  GOTO 630
590  IF Z[2]#0 THEN 620
600  P$[W1+1,W1+1]="0"
610  GOTO 630
620  P$[W1+1,W1+1]=C$[Z[2],Z[2]]
630  IF Z[3]#0 THEN 660
640  P$[W1+2,W1+2]="0"
650  GOTO 662
660  P$[W1+2,W1+2]=C$[Z[3],Z[3]]
662  N=D[W+1,C]
```

```
664   IF INT((N-D9)/7)#(N-D9)/7 THEN 670
666   P$[W1+3,W1+3]="*"
670   NEXT W
675   PRINT P$
680   NEXT C
900   PRINT
910   PRINT
920   PRINT TAB(20)"SUNDAYS ARE SHOWN WITH AN *"
999   STOP
1000   FOR B=1 TO 64
1010   P$[B,B]=" "
1020   NEXT B
1030   RETURN
2000   REM  THIS SUB BRINGS BACK 3 NOS. = TO THE 3 DIGITS OF NUMBER
2010   REM  ENTER WITH NUMBER N. LEAVE WITH NUMBERS Z(I)
2020   FOR Z1=3 TO 1 STEP -1
2030   N=N/10
2040   Z[Z1]=INT(10*(N-INT(N))+.001)
2050   NEXT Z1
2060   RETURN
3000   Y1=Y-1
3005   D9=37+INT(Y1/4)+Y1+INT(Y1/400)-INT(Y1/100)
3010   Q=7000
3020   IF D9-Q <= 0 THEN 3050
3030   D9=D9-Q
3040   GOTO 3020
3050   IF Q=7 THEN 3080
3060   Q=Q/10
3070   GOTO 3020
3080   D9=D9+1
3090   IF D9<8 THEN 3200
3100   D9=D9-7
3110   GOTO 3090
3200   D9=10-D9
3202   IF D9<8 THEN 3210
3204   D9=D9-7
3210   RETURN
9999   END
```

CALNDR
F108-36288A

TITLE: PRINTS A CALENDAR

DESCRIPTION:

This program enables a user to print a calendar for any month or all months for any year.

INSTRUCTIONS:

No special instructions are needed.

The user may answer "Y" or "N" for "Yes/No" questions. A response of "END" to a "Yes" or "No" takes the user to the previous question. A response of "AID" explains any input that is required. A response of "STOP" automatically ends execution of the program.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Steve Mudrick
Leasco Response Inc.

RUN

```
RUN
CALNDR


CALENDAR FOR WHAT YEAR?AID
     ENTER A YEAR AFTER 1581 AND BEFORE 8388608.

CALENDAR FOR WHAT YEAR?1973

ANY PARTICULAR MONTH (Y OR N)?N

                    CALENDAR FOR THE YEAR 1973

      JANUARY                  FEBRUARY                 MARCH
 S  M  T  W  T  F  S      S  M  T  W  T  F  S      S  M  T  W  T  F  S
    1  2  3  4  5  6                  1  2  3                  1  2  3
 7  8  9 10 11 12 13      4  5  6  7  8  9 10      4  5  6  7  8  9 10
14 15 16 17 18 19 20     11 12 13 14 15 16 17     11 12 13 14 15 16 17
21 22 23 24 25 26 27     18 19 20 21 22 23 24     18 19 20 21 22 23 24
28 29 30 31              25 26 27 28              25 26 27 28 29 30 31

      APRIL                    MAY                      JUNE
 S  M  T  W  T  F  S      S  M  T  W  T  F  S      S  M  T  W  T  F  S
 1  2  3  4  5  6  7               1  2  3  4  5                  1  2
 8  9 10 11 12 13 14      6  7  8  9 10 11 12      3  4  5  6  7  8  9
15 16 17 18 19 20 21     13 14 15 16 17 18 19     10 11 12 13 14 15 16
22 23 24 25 26 27 28     20 21 22 23 24 25 26     17 18 19 20 21 22 23
29 30                    27 28 29 30 31           24 25 26 27 28 29 30

      JULY                     AUGUST                   SEPTEMBER
 S  M  T  W  T  F  S      S  M  T  W  T  F  S      S  M  T  W  T  F  S
 1  2  3  4  5  6  7                  1  2  3  4                     1
 8  9 10 11 12 13 14      5  6  7  8  9 10 11      2  3  4  5  6  7  8
15 16 17 18 19 20 21     12 13 14 15 16 17 18      9 10 11 12 13 14 15
22 23 24 25 26 27 28     19 20 21 22 23 24 25     16 17 18 19 20 21 22
29 30 31                 26 27 28 29 30 31        23 24 25 26 27 28 29
                                                  30

      OCTOBER                  NOVEMBER                 DECEMBER
 S  M  T  W  T  F  S      S  M  T  W  T  F  S      S  M  T  W  T  F  S
    1  2  3  4  5  6                  1  2  3                     1
 7  8  9 10 11 12 13      4  5  6  7  8  9 10      2  3  4  5  6  7  8
14 15 16 17 18 19 20     11 12 13 14 15 16 17      9 10 11 12 13 14 15
21 22 23 24 25 26 27     18 19 20 21 22 23 24     16 17 18 19 20 21 22
28 29 30 31              25 26 27 28 29 30        23 24 25 26 27 28 29
                                                  30 31
```

```
MORE (Y OR N)?Y

CALENDAR FOR WHAT YEAR?2000

ANY PARTICULAR MONTH (Y OR N)?Y

WHAT MONTH?2



                FEBRUARY        2000

 S  M  T  W  T  F  S
          1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29



MORE (Y OR N)?N

DONE
```

LISTING

CALNDR

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         CALNDR:   PRINTS A CALENDAR
4  REM
5  REM         36288 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
1000  REM ** CALENDAR DEMONSTRATION **
1010  REM ** WRITTEN BY STEVE HTOPHCK **
1020  DIM Y$[72],Z$[72],A[504],B$[72],B[12],E[42,12],J$[72],M$[72],C$[72]
1030  DIM L[80],F$[72],P$[72],U$[72],L$[72],T$[72],S$[72],O$[72],N$[72]
1040  DIM A$[72],G$[72],H$[72],I$[72],E$[72],D$[72]
1050  REM * SETS OUTPUT STRING TO SPACES
1060  LET Y$[1,72]=" "
1070  LET Z$[1,72]=" "
1080  LET K0=S1=0
1090  LET H$="0123456789"
1100  REM * READS LAST DAY OF MONTH
1110  FOR I=1 TO 12
1120  READ B[I]
1130  NEXT I
1140  DATA 31,28,31,30,31,30,31,31,30,31,30,31
1150  PRINT
1160  LET G$=" S  M  T  W  T  F  S"
1170  REM * INPUT DATA SECTION
1180  PRINT "CALENDAR FOR WHAT YEAR";
1190  INPUT J$
1200  LET I$=J$
1210  IF J$="END" THEN 4030
1220  IF J$="STOP" THEN 4030
1230  IF J$="AID" THEN 1830
1240  LET Y=0
1250  FOR I=1 TO LEN(J$)
1260  FOR J=1 TO 10
1270  IF J$[I,I]=H$[J,J] THEN 1300
1280  NEXT J
1290  GOTO 1350
1300  LET Y=Y*10+J-1
1310  NEXT I
1320  GOSUB 3900
1330  IF Y#INT(Y) OR Y<0 THEN 1350
```

```
1340  GOTO 1370
1350  PRINT "     ENTRY IGNORED. INVALID YEAR."
1360  GOTO 1180
1370  IF Y<1582 THEN 1830
1380  IF Y>2↑23-1 THEN 1830
1390  PRINT "ANY PARTICULAR MONTH (Y OR N)":
1400  INPUT B$
1410  IF B$="END" THEN 1180
1420  IF B$="STOP" THEN 4030
1430  IF B$[1,1]="Y" THEN 1470
1440  IF B$[1,1]="N" THEN 1660
1450  PRINT "     ANSWER 'Y' OR 'YES'.....'N' OR 'NO'"
1460  GOTO 1390
1470  PRINT "WHAT MONTH":
1480  INPUT C$
1490  IF C$="STOP" THEN 4030
1500  IF C$="END" THEN 1390
1510  IF C$#"AID" THEN 1540
1520  PRINT "     ENTER '1' FOR JANUARY, '2' FOR FEBRUARY, ETC."
1530  GOTO 1470
1540  LET B1=0
1550  FOR I=1 TO LEN(C$)
1560  FOR J=1 TO 10
1570  IF C$[I,I]=H$[J,J] THEN 1600
1580  NEXT J
1590  GOTO 1640
1600  LET B1=B1*10+J-1
1610  NEXT I
1620  IF B1 <= 0 OR B1>12 OR B1#INT(B1) THEN 1640
1630  GOTO 1660
1640  PRINT "     ENTRY IGNORED, INVALID MONTH."
1650  GOTO 1470
1660  IF B$[1,1]="N" THEN 1810
1670  PRINT ""
1680  GOSUB 3510
1690  REM * SPACE COUNTER
1700  LET X3=0
1710  FOR S3=1 TO 5
1720  IF C$[S3,S3]=" " THEN 1740
1730  GOTO 1760
1740  LET X3=X3+1
1750  NEXT S3
1760  PRINT TAB(19-X3);C$;TAB(38-LEN(I$));I$
1770  PRINT
1780  PRINT TAB(18);G$
1790  GOTO 2120
1800  REM * PAGE SEPERATOR
1810  GOSUB 4040
1820  IF Y >= 1582 THEN 1860
1830  PRINT "     ENTER A YEAR AFTER 1581 AND BEFORE 8388608."
1840  GOTO 1180
1850  REM * HEADINGS
1860  PRINT
1870  PRINT TAB(23);"CALENDAR FOR THE YEAR ";I$
1880  PRINT
1890  LET I=1
1900  PRINT
1910  PRINT
1920  PRINT
1930  REM * TAB VALUES
1940  LET B=5
1950  LET C=31
1960  LET D=55
1970  REM * K0= NO OF GROUPS OF 3 MONTHS PRINTED
1980  LET K0=K0+1
1990  GOTO K0 OF 2010,2030,2050,2070
2000  REM * PRINTS MONTH HEADINGS
2010  PRINT TAB(B);J$;TAB(C);F$;TAB(D);M$
2020  GOTO 2080
2030  PRINT TAB(B);P$;TAB(C);E$;TAB(D);U$
2040  GOTO 2080
2050  PRINT TAB(B);L$;TAB(C);T$;TAB(D);S$
2060  GOTO 2080
2070  PRINT TAB(B);O$;TAB(C);N$;TAB(D);D$
2080  PRINT
2090  PRINT G$;TAB(25);G$;TAB(50);G$
2100  GOTO I OF 2120,2870,2880,2880
2110  REM * CHECKS FOR LEAP YEARS
```

```
2120  LET H=7
2130  LET L=0
2140  LET L=INT((Y-1201)/400)-INT((Y-1501)/100)+INT((Y-1581)/4)
2150  LET I1=Y-INT(Y/100)*100
2160  IF I1 <> 0 THEN 2190
2170  IF I1-INT(I1/4)*4=0 THEN 2210
2180  GOTO 2290
2190  IF Y-INT(Y/4)*4=0 THEN 2210
2200  GOTO 2290
2210  FOR X=1600 TO Y STEP 400
2220  IF Y=X THEN 2260
2230  NEXT X
2240  GOSUB 3780
2250  IF S1=1 THEN 2290
2260  LET R=366
2270  LET B[2]=29
2280  GOTO 2300
2290  LET R=365
2300  LET M=365*(Y-1583)+L
2310  LET N=M-INT(M/7)*7
2320  REM * ROUTINE TO DETERMINE PROPER DATES
2330  LET H=H+N
2340  IF H <= 7 THEN 2360
2350  LET H=H-7
2360  GOTO 2370
2370  FOR S=1 TO 12
2380  FOR T=1 TO H-1
2390  LET E[T,S]=0
2400  NEXT T
2410  FOR T=H TO B[S]+H-1
2420  LET E[T,S]=T-H+1
2430  NEXT T
2440  FOR T=B[S]+H TO 42
2450  LET E[T,S]=0
2460  NEXT T
2470  LET H=H+B[S]-INT(B[S]/7)*7
2480  IF H <= 7 THEN 2500
2490  LET H=H-7
2500  NEXT S
2510  LET Q=0
2520  LET W=0
2530  FOR V=1 TO 12
2540  FOR U=1 TO 42
2550  LET W=W+1
2560  LET A[W]=E[U,V]
2570  NEXT U
2580  NEXT V
2590  IF B$[1,1]="N" THEN 2880
2600  FOR X=0 TO 5
2610  LET I=-2
2620  LET J=0
2630  FOR V=1 TO 7
2640  LET I=I+3
2650  LET J=J+3
2660  LET A=A[V+7*X+42*B1-42]
2670  LET A1=INT(A/10)
2680  LET B=A-A1*10
2690  IF A=0 THEN 2740
2700  LET C$[1,1]=H$[A1+1,A1+1]
2710  LET C$[2,2]=H$[B+1,B+1]
2720  IF A<10 THEN 2770
2730  GOTO 2750
2740  LET C$="  "
2750  LET A$[I,J]=C$
2760  GOTO 2800
2770  LET A$[I,1]=" "
2780  LET C$[1,1]=" "
2790  LET A$[I,J]=C$
2800  NEXT V
2810  PRINT TAB(18);A$
2820  NEXT X
2830  PRINT
2840  PRINT
2850  GOTO 3390
2860  REM
2870  REM
2880  FOR X=0 TO 5
2890  LET R=-2
```

```
2900  LET S=0
2910  FOR U=0 TO 2
2920  FOR V=1+7*X TO 7+7*X
2930  LET R=R+3
2940  LET S=S+3
2950  REM * TO RIGHT JUSTIFY OUTPUT DATE
2960  LET A=A[V+42*U]
2970  IF A=0 THEN 3050
2980  LET A1=INT(A/10)
2990  LET B=A-A1*10
3000  LET C$[1,1]=H$[A1+1,A1+1]
3010  LET C$[2,2]=H$[B+1,B+1]
3020  IF C$="0" THEN 3050
3030  IF A<10 THEN 3080
3040  GOTO 3060
3050  LET C$="   "
3060  LET Z$[R,S]=C$
3070  GOTO 3110
3080  LET Z$[R,R]=" "
3090  LET C$[1,1]=" "
3100  LET Z$[R,S]=C$
3110  NEXT V
3120  NEXT U
3130  REM * TO PLACE INTO STRING FOR OUTPUT
3140  FOR G=1 TO 21
3150  LET Y$[G,G]=Z$[G,G]
3160  NEXT G
3170  FOR G=1 TO 4
3180  LET Y$[21+G,21+G]=" "
3190  NEXT G
3200  FOR G=22 TO 42
3210  LET Y$[G+4,G+4]=Z$[G,G]
3220  NEXT G
3230  FOR G=1 TO 4
3240  LET Y$[46+G,46+G]=" "
3250  NEXT G
3260  FOR G=43 TO 63
3270  LET Y$[G+8,G+8]=Z$[G,G]
3280  NEXT G
3290  PRINT Y$
3300  NEXT X
3310  PRINT
3320  FOR X=1 TO 378
3330  LET A[X]=A[X+126]
3340  NEXT X
3350  LET Q=Q+1
3360  IF Q=4 THEN 3390
3370  LET I=I+1
3380  GOTO 1910
3390  PRINT ""
3400  PRINT "MORE (Y OR N)":
3410  INPUT C$
3420  RESTORE
3430  IF C$[1,1]="Y" THEN 1060
3440  IF C$[1,1]="N" THEN 4030
3450  IF C$="END" THEN 4030
3460  IF C$="STOP" THEN 4030
3470  PRINT "     ANSWER 'Y' OR 'YES'.....'N' OR 'NO'"
3480  GOTO 3400
3490  GOTO 3400
3500  REM * TO DETERMINE WHAT MONTH
3510  GOTO R1 OF 3530,3550,3570,3590,3610,3630,3650,3670,3690,3710,3730,3750
3520  RETURN
3530  LET C$=J$
3540  RETURN
3550  LET C$=F$
3560  RETURN
3570  LET C$=M$
3580  RETURN
3590  LET C$=P$
3600  RETURN
3610  LET C$=E$
3620  RETURN
3630  LET C$=U$
3640  RETURN
3650  LET C$=L$
3660  RETURN
3670  LET C$=T$
```

```
3680  RETURN
3690  LET C$=S$
3700  RETURN
3710  LET C$=O$
3720  RETURN
3730  LET C$=N$
3740  RETURN
3750  LET C$=D$
3760  RETURN
3770  REM * DETERMINES IF CENTURY YEAR
3780  LET Z$=I$
3790  LET K=LEN(Z$)
3800  IF Z$[K-1,K-1]="0" THEN 3830
3810  LET S1=0
3820  RETURN
3830  IF Z$[K,K]="0" THEN 3860
3840  LET S1=0
3850  RETURN
3860  LET S1=1
3870  RETURN
3880  END
3890  PRINT
3900  LET J$=" JANUARY "
3910  LET F$="FEBRUARY "
3920  LET M$="  MARCH  "
3930  LET P$="  APRIL  "
3940  LET E$="   MAY   "
3950  LET U$="  JUNE   "
3960  LET L$="  JULY   "
3970  LET T$=" AUGUST "
3980  LET S$="SEPTEMBER"
3990  LET O$=" OCTOBER "
4000  LET N$="NOVEMBER "
4010  LET D$="DECEMBER "
4020  RETURN
4030  STOP
4040  PRINT ""
4050  RETURN
4060  END
```

FILRPT
F110-36247A

TITLE:

A REPORT ON FILE CONTENTS AND STRUCTURE

DESCRIPTION:

This program prints a description of each record in a file which includes the amount of numbers and the amount of strings in the record, presence of an end-of-file mark, and the number of words of data in each record and in the entire file.

The program provides a useful check on the contents of a file without the necessity of listing the entire file.

INSTRUCTIONS:

User must input the name of his file as the program requests it.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

This program may be adapted to a 2000A, 2000B, or 2000E system by deleting lines 1050-1110 and declaring the file in line 1010.

ACKNOWLEDGEMENTS:

RUN

```
RUN
FILRPT

FILENAME? GENPUR



RECORD 1:  3 STRINGS, 17 WORDS.
RECORD 2:  2 NUMBERS, 4 STRINGS, 39 WORDS.
RECORD 3:  1 STRING, 8 WORDS.
RECORD 4:  0 WORDS.
RECORD 5:  0 WORDS.
RECORD 6:  0 WORDS.
RECORD 7:  0 WORDS.
RECORD 8:  0 WORDS.
RECORD 9:  0 WORDS.
RECORD 10:  0 WORDS.

TOTAL FILE LENGTH = 10 RECORDS, 64 WORDS.
AVERAGE OF 6 WORDS PER RECORD.

DONE


RUN
FILRPT

FILENAME? NAMES



RECORD 1:  3 STRINGS, 59 WORDS.
RECORD 2:  0 WORDS.

TOTAL FILE LENGTH = 2 RECORDS, 59 WORDS.
AVERAGE OF 30 WORDS PER RECORD.

DONE


RUN
FILRPT

FILENAME? IREP



RECORD 1:  16 NUMBERS, 32 WORDS.
RECORD 2:  0 WORDS.
RECORD 3:  3 STRINGS, 10 WORDS.
RECORD 4:  0 WORDS.
RECORD 5:  0 WORDS.
RECORD 6:  0 WORDS.
RECORD 7:  1 NUMBER, 2 WORDS.
RECORD 8:  1 NUMBER, 2 WORDS.
RECORD 9:  1 NUMBER, 2 WORDS.
RECORD 10:  0 WORDS.
RECORD 11:  0 WORDS.
RECORD 12:  0 WORDS.
RECORD 13:  0 WORDS.
RECORD 14:  0 WORDS.
RECORD 15:  0 WORDS.
RECORD 16:  0 WORDS.
RECORD 17:  0 WORDS.
RECORD 18:  0 WORDS.
RECORD 19:  0 WORDS.
RECORD 20:  0 WORDS.

TOTAL FILE LENGTH = 20 RECORDS, 48 WORDS.
AVERAGE OF 2 WORDS PER RECORD.

DONE
```

```
RUN
FILRPT

FILENAME? $CHARS



RECORD 1:  2 STRINGS, EOF, 66 WORDS.

TOTAL FILE LENGTH = 1 RECORD, 66 WORDS.

DONE
```

LISTING

```
FILRPT

1   REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2   REM
3   REM         FILRPT:   REPORTS FILE CONTENTS AND STRUCTURE
4   REM
5   REM         36247 REV  A   6/73
6   REM
7   REM  ****  CONTRIBUTED PROGRAM  ************************************
1010  FILES *
1020  DIM A$[72],D$[10],F$[7],T$[72],Z$[6],M[7]
1030  D$="0123456789"
1040  L9=0
1050  PRINT "FILENAME? ";
1060  ENTER 255,T,F$
1070  IF T=-256 THEN 1110
1080  PRINT "";
1090  ASSIGN F$,1,T1
1100  IF T1<3 THEN 1120
1110  STOP
1120  IF  END #1 THEN 1600
1130  PRINT ""
1140  FOR I=1 TO 32767
1150  F1=F2=F3=F9=0
1160  READ #1,I
1170  GOTO TYP(-1) OF 1180,1220,1260,1270
1180  READ #1;A
1190  F1=F1+1
1200  F9=F9+2
1210  GOTO 1170
1220  READ #1;A$
1230  F2=F2+1
1240  F9=F9+INT((LEN(A$)+1)/2)+1
1250  GOTO 1170
1260  F3=1
1270  L9=L9+F9
1280  T$="RECORD "
1290  Z=I
1300  GOSUB 1810
1310  T$[8]=Z$
1320  T$[LEN(T$)+1]=":  "
1330  IF F1=0 THEN 1410
1340  Z=F1
1350  GOSUB 1810
1360  T$[LEN(T$)+1]=Z$
1370  T$[LEN(T$)+1]=" NUMBER"
1380  IF F1=1 THEN 1400
1390  T$[LEN(T$)+1]="S"
1400  T$[LEN(T$)+1]=", "
1410  IF F2=0 THEN 1490
1420  Z=F2
1430  GOSUB 1810
1440  T$[LEN(T$)+1]=Z$
1450  T$[LEN(T$)+1]=" STRING"
1460  IF F2=1 THEN 1480
1470  T$[LEN(T$)+1]="S"
1480  T$[LEN(T$)+1]=", "
1490  IF F1+F2=0 OR F3=0 THEN 1510
```

```
1500  T$[LEN(T$)+1]="EOF. "
1510  Z=F9
1520  GOSUB 1810
1530  T$[LEN(T$)+1]=Z$
1540  T$[LEN(T$)+1]=" WORD"
1550  IF F9=1 THEN 1570
1560  T$[LEN(T$)+1]="S"
1570  T$[LEN(T$)+1]="."
1580  PRINT T$
1590  NEXT I
1600  Z=I-1
1610  GOSUB 1810
1620  T$="TOTAL FILE LENGTH = "
1630  T$[21]=Z$
1640  T$[LEN(T$)+1]=" RECORD"
1650  IF I-1=1 THEN 1670
1660  T$[LEN(T$)+1]="S"
1670  T$[LEN(T$)+1]=". "
1680  Z=L9
1690  GOSUB 1810
1700  T$[LEN(T$)+1]=Z$
1710  T$[LEN(T$)+1]=" WORDS."
1720  PRINT LIN(1);T$
1730  IF I-1=1 THEN 1800
1740  T$="AVERAGE OF "
1750  Z=INT(L9/(I-1)+.5)
1760  GOSUB 1810
1770  T$[12]=Z$
1780  T$[LEN(T$)+1]=" WORDS PER RECORD."
1790  PRINT T$
1800  STOP
1810  FOR K1=1 TO 7
1820  IF Z<10↑K1 THEN 1840
1830  NEXT K1
1840  FOR K2=K1 TO 1 STEP -1
1850  M[K2]=INT(Z/10↑(K2-1))
1860  FOR K3=K2 TO K1-1
1870  M[K2]=M[K2]-M[K3+1]*10↑(K3+1-K2)
1880  NEXT K3
1890  Z$[K1+1-K2]=D$[M[K2]+1,M[K2]+1]
1900  NEXT K2
1910  RETURN
1920  END
```

DATA
F212-36287A

TITLE: DUMP FILE TO DATA STATEMENTS

DESCRIPTION:

This program dumps the contents of a file into BASIC "DATA" statements. It inputs a starting and step value for the statement numbers. The program "READ" is included as a sample program to illustrate filling a file ("TEST") with data.

INSTRUCTIONS:

The file "CHARS" (HP 36220) must be in the system library, containing at least the first 58 ASCII characters in order in one string. The user's file may be filled using "READ", "FILOAD" (HP 36010) or a similar format. The DATA program prompts the user to enter the file name, and the starting statement number (SS) and interval (I). The paper tape punch is then turned on, and the paper tape is generated. The tape dump should include leader and no "X OFF" characters on the tape.

SYSTEM SPECIFICATIONS: 2000F and Teletype

SPECIAL CONSIDERATIONS:

The file CHARS can be filled by ASSIGN statement masks.

ACKNOWLEDGEMENTS: Bruce A. Robinson
The Evergreen State College

READ

```
10  FILES TEST
20  DIM A$[72]
30  GOTO TYP(0) OF 40,70,150
40  READ A
50  PRINT #1;A
60  GOTO 30
70  READ A$
80  PRINT #1;A$
90  GOTO 30
100  DATA 123.124,383,0.59,-293.45,-3.8E-12,"NOW IS THE "
110  DATA "FOR ALL GOOD MEN TO COME TO THE AID OF THEIR PARTY"
120  DATA 3.73774E+21,3.5E+16,-3.7E-15,1.E-13
130  DATA "THIS STRING IS 72 CHARACTERS LONG 567890123456789012345678901234567890123456789012"
140  DATA 0,0,0,1,2,3,4,-1,-2,-3,-4,"OVER AND OUT"
150  END
```

RUN

```
RUN
FILE?TEST
SS,I?500,1
TEST

500  DATA  123.124,383,0,59,-293.45,-3.80000E-12,"NOW IS THE "
501  DATA  "FOR ALL GOOD MEN TO COME TO THE AID OF THEIR PARTY"
502  DATA  +3.73774E+21,+3.50000E+16,-3.70000E-15,+1.00000E-13
503  DATA  "THIS STRING IS 72 CHARACTERS LONG 567890123456789012345678901234567890123456789012"
504  DATA  0,0,0,1,2,3,4,-1,-2,-3,-4,"OVER AND OUT"
```

LISTING

DATA

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         DATA:   DUMPS FILE TO DATA STATEMENTS
4  REM
5  REM         36287 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
1000  FILES *,$CHARS
1050  DIM A$[4],B$[9],C$[10],D$[8],E$[7],F$[72],G$[72]
1100  DEF FNA(X)=2+(X >= 10)+(X >= 100)+(X >= 1000)+(X >= 10000)+(X >= 100000.)
1150  READ #2;G$
1200  A$="#, D"
1250  B$="#,A,  A,A"
1300  E$="#, D. D"
1350  D$="  DATA  "
1400  C$=G$[49,58]
1450  H$=G$[14,14]
1500  I$=G$[35,35]
1550  FOR A=2 TO 72
1600  G$[A]=G$[1,1]
1650  NEXT A
1700  PRINT "FILE";
1750  INPUT F$
1800  ASSIGN F$,1,H
1850  IF H<3 THEN 2000
1900  PRINT "FILE NOT ACCESSIBLE"
1950  GOTO 1700
2000  PRINT "SS,I";
2050  INPUT B,C
2100  PRINT G$;
2150  PRINT F$
2200  PRINT G$;
2250  D=E=73
2300  GOTO TYP(1) OF 2350,3700,5550
2350  READ #1;F
2400  IF ABS(F)>999998. OR (F AND ABS(F)<.000001) THEN 2500
2450  GOTO (F=INT(F))+1 OF 3000,2700
2500  GOSUB (E+13>72)+1 OF 5450,4950
```

```
2550  E=E+13
2600  PRINT  USING "#.5D.5DE";F
2650  GOTO 2300
2700  A=FNA(ABS(F))+(F<0)
2750  GOSUB (E+A>72)+1 OF 5450,4950
2800  E=E+A
2850  A$[3,3]=C$[A,A]
2900  PRINT  USING A$;F
2950  GOTO 2300
3000  A=FNA(ABS(F))+(F<0)
3050  E$[3,3]=C$[A,A]
3100  H=ABS(F)
3150  FOR G=2 TO 7
3200  H=H*10
3250  IF H=INT(H) THEN 3350
3300  NEXT G
3350  E$[6,6]=C$[G,G]
3400  GOSUB (E+A+G>72)+1 OF 5450,4950
3450  E=E+G+A
3500  PRINT  USING E$;F
3550  E$[3,3]=" "
3600  E$[6,6]=" "
3650  GOTO 2300
3700  READ #1;F$
3750  A=LEN(F$)+3
3800  GOSUB (E+A>72)+1 OF 5450,4950
3850  IF A<60 THEN 4300
3900  D=0
3950  PRINT I$;F$[1,29];
4000  F$=F$[30]
4050  F$[LEN(F$)+1]=I$
4100  F$[LEN(F$)+1]=H$
4150  F$[LEN(F$)+1]=""
4200  A=LEN(F$)+3
4250  E=E+29
4300  E=E+A
4350  G=INT((A-3)/10)+1
4400  B$[5,5]=C$[G,G]
4450  G=A-2-(G-1)*10
4500  B$[6,6]=C$[G,G]
4550  IF D THEN 4800
4600  B$[3,4]="  "
4650  PRINT  USING B$;F$
4700  B$[3,4]="A."
4750  GOTO 4850
4800  PRINT  USING B$;1$,F$.I$
4850  B$[5,6]="  "
4900  GOTO 2300
4950  H=FNA(B)
5000  A$[3,3]=C$[H,H]
5050  IF  NOT D THEN 5150
5100  PRINT "";
5150  PRINT  USING A$;B
5200  D=1
5250  PRINT  USING "#,8A";D$
5300  E=H+8
5350  B=B+C
5400  RETURN
5450  PRINT ",";
5500  RETURN
5550  IF  NOT D THEN 5650
5600  PRINT "";
5650  PRINT G$
5700  END
```

TITLE:	INTEGRATES A FUNCTION (ROMBERG METHOD)	ROMINT A310-36022B
DESCRIPTION:	This program will integrate a given function by the Romberg Method.	
INSTRUCTIONS:	Define the integrand in line 100 by a "DEF FNF(X)=..." statement i.e., 100 DEF FNF(X)=X↑2" The lower and upper limits of integration will be requested during execution. The output is the sequence of the first five approximations which should converge to the value of the integral. The number of approximations may be increased by changing the value of N in line 107.	
SYSTEM SPECIFICATIONS:	2000A and Teletype	
SPECIAL CONSIDERATIONS:	Specifying an order of integration greater than 5 can result in excessive running time and usually will not improve accuracy.	
ACKNOWLEDGEMENTS:	B. Gateley Colorado College	

RUN

```
RUN
ROMINT

'RUN-100' TO AVOID INSTRUCTIONS.

THIS PROGRAM INTEGRATES A FUNCTION USING THE ROMBERG METHOD.  SEE THE
ACM ALGORITHM #60 FOR TECHNICAL DETAILS.  TO USE, DEFINE THE INTEGRAND
IN LINE 100 BY A 'DEF FNF(X)=...' STATEMENT - E.G.,
    100 DEF FNF(X)=X↑2
THE LOWER AND UPPER LIMITS OF INTEGRATION WILL BE REQUESTED DURING
EXECUTION.  THE OUTPUT IS THE SEQUENCE OF THE FIRST FIVE APPROXIMATIONS
(WHICH SHOULD CONVERGE TO THE VALUE OF THE INTEGRAL).  THE NUMBER OF
APPROXIMATIONS MAY BE INCREASED BY CHANGING THE VALUE OF N IN LINE 107.


LOWER LIMIT?0
UPPER LIMIT?1

APPROXIMATION VALUES:
 .5
 .5
 .5
 .5
 .5

DONE


RUN-100
ROMINT

LOWER LIMIT?-1
UPPER LIMIT?1

APPROXIMATION VALUES:
 0
 0
 0
 0
 0

DONE
```

LISTING

```
ROMINT

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         ROMINT:   INTEGRATES A FUNCTION (ROMBERG METHOD)
4  REM
5  REM         36022 REV  B   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  REM  ***  ROMBERG INTEGRATION.  HP VERSION MODIFIED BY B. GATELEY:10/24/72 ***
15  PRINT "'RUN-100' TO AVOID INSTRUCTIONS."LIN(1)
20  PRINT "THIS PROGRAM INTEGRATES A FUNCTION USING THE ROMBERG METHOD.  SEE THE"
25  PRINT "ACM ALGORITHM #60 FOR TECHNICAL DETAILS.  TO USE, DEFINE THE INTEGRAND"
30  PRINT "IN LINE 100 BY A 'DEF FNF(X)=...' STATEMENT - E.G.,"
35  PRINT "    100 DEF FNF(X)=X↑2"
40  PRINT "THE LOWER AND UPPER LIMITS OF INTEGRATION WILL BE REQUESTED DURING"
45  PRINT "EXECUTION.  THE OUTPUT IS THE SEQUENCE OF THE FIRST FIVE APPROXIMATIONS"
50  PRINT "(WHICH SHOULD CONVERGE TO THE VALUE OF THE INTEGRAL).  THE NUMBER OF"
55  PRINT "APPROXIMATIONS MAY BE INCREASED BY CHANGING THE VALUE OF N IN LINE 107."
60  PRINT LIN(1)
100  DEF FNF(X)=X
101  DIM Z[20]
102  PRINT "LOWER LIMIT";
103  INPUT A
104  PRINT "UPPER LIMIT";
105  INPUT B
106  PRINT LIN(1)"APPROXIMATION VALUES:"
```

```
107  N=5
108  Z1=B-A
109  LET Z8=1
110  LET Z[1]=(FNF(A)+FNF(B))/2
111  FOR Z5=1 TO N
112  LET Z6=0
113  Z7=Z1/(2*Z8)
114  FOR Z9=1 TO 2*Z8-1 STEP 2
115  LET Z6=Z6+FNF(A+Z7*Z9)
116  NEXT Z9
117  LET Z[Z5+1]=(Z6/Z8+Z[Z5])/2
118  LET Z6=1
119  FOR Z9=Z5 TO 1 STEP -1
120  LET Z6=4*Z6
121  LET Z[Z9]=Z[Z9+1]+(Z[Z9+1]-Z[Z9])/(Z6-1)
122  NEXT Z9
123  LET Z8=Z8*2
124  LET Z7=Z7/2
125  PRINT Z[1]*Z1
126  NEXT Z5
127  END
```

FISHER
F407-36606A

TITLE: FISHER'S EXACT PROBABILITY TEST

DESCRIPTION:

This program analyzes discrete data from two independent small random samples which fall into one or another of two mutually exclusive classes. The printout includes a summary table with marginal frequencies and the probability of occurence by chance of the distribution under examination.

INSTRUCTIONS:

Instructions for the use of this program are given at run-time for the entry of data into a 2 x 2 table of the following format:

```
+----------+---------+
     A          B
+----------+---------+
     C          D
+----------+---------+
```

Reference: Siegel, Sidney NON-PARAMETRIC STATISTICS, McGraw-Hill; New York 1956, Page 96

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Robert M. Smith
University of Alabama School of Medicine

```
RUN


RUN
FISHER

FISHER'S EXACT PROBABILITY TEST
===============================
ENTER THE FREQUENCY IN CELL 'A'
?10
ENTER THE FREQUENCY IN CELL 'B'
?0
ENTER THE FREQUENCY IN CELL 'C'
?4
ENTER THE FREQUENCY IN CELL 'D'
?5

     SUMMARY TABLE
     =============

+----------+----------+
   10            0          10
+----------+----------+
    4            5           9
+----------+----------+
   14            5          19

P =    0.01084


DONE


RUN
FISHER

FISHER'S EXACT PROBABILITY TEST
===============================
ENTER THE FREQUENCY IN CELL 'A'
?1
ENTER THE FREQUENCY IN CELL 'B'
?6
ENTER THE FREQUENCY IN CELL 'C'
?4
ENTER THE FREQUENCY IN CELL 'D'
?1

     SUMMARY TABLE
     =============

+----------+----------+
    1            6           7
+----------+----------+
    4            1           5
+----------+----------+
    5            7          12

P =    0.04419


DONE
```

```
RUN
FISHER

FISHER'S EXACT PROBABILITY TEST
==============================
ENTER THE FREQUENCY IN CELL 'A'
?0
ENTER THE FREQUENCY IN CELL 'B'
?7
ENTER THE FREQUENCY IN CELL 'C'
?5
ENTER THE FREQUENCY IN CELL 'D'
?0

     SUMMARY TABLE
     =============

+----------+----------+
    0           7           7
+----------+----------+
    5           0           5
+----------+----------+
    5           7          12

P =    0.00126


DONE
```

LISTING

```
FISHER

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         FISHER:   FISHER'S EXACT PROBABILITY TEST
4  REM
5  REM         36606 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
9000  PRINT "FISHER'S EXACT PROBABILITY TEST"
9010  PRINT "=============================="
9020  PRINT "ENTER THE FREQUENCY IN CELL 'A'"
9030  INPUT A
9040  PRINT "ENTER THE FREQUENCY IN CELL 'B'"
9050  INPUT B
9060  PRINT "ENTER THE FREQUENCY IN CELL 'C'"
9070  INPUT C
9080  PRINT "ENTER THE FREQUENCY IN CELL 'D'"
9090  INPUT D
9100  N=A+B+C+D
9110  D1=D2=D3=D4=D5=1
9120  N1=N2=N3=N4=N5=1
9130  FOR I=1 TO A+B
9140  N1=N1*I
9150  NEXT I
9160  FOR I=1 TO C+D
9170  N2=N2*I
9180  NEXT I
9190  FOR I=1 TO A+C
9200  N3=N3*I
9210  NEXT I
9220  FOR I=1 TO B+D
9230  N4=N4*I
9240  NEXT I
9250  FOR I=1 TO A+B+C+D
9260  D1=D1*I
9270  NEXT I
9280  FOR I=1 TO A
9290  D2=D2*I
9300  NEXT I
9310  FOR I=1 TO B
9320  D3=D3*I
9330  NEXT I
```

```
9340  FOR I=1 TO C
9350  D4=D4*I
9360  NEXT I
9370  FOR I=1 TO D
9380  D5=D5*I
9390  NEXT I
9400  PRINT
9410  PRINT "      SUMMARY TABLE"
9420  PRINT "      ==============="
9430  PRINT
9440  PRINT "+----------+----------+"
9450  PRINT  USING 9460:A,B,A+B
9460  IMAGE "   ",DD,"           ",DD,"           ",DD
9470  PRINT "+----------+----------+"
9480  PRINT  USING 9460:C,D,C+D
9490  PRINT "+----------+----------+"
9500  PRINT  USING 9460:A+C,B+D,A+B+C+D
9510  PRINT
9520  P=N1*N2*N3*N4
9530  P=P/(D1*D2*D3*D4*D5)
9540  PRINT  USING 9550:P
9550  IMAGE "P =",XXXDD.DDDDD
9560  PRINT LIN(5)
9570  END
```

ANOVA3

TITLE: THREE FACTORIAL ANALYSIS OF VARIANCE F410-36271A

DESCRIPTION:

This program computes an analysis of variance for an experiment with three factors. Each factor may have up to 8 levels. The number of observations for each cell must be the same.

The printout consists of a table listing sum of squares, mean squares, and F-ratios, for Rows, Columns, Layers, and the various interactions.

INSTRUCTIONS:

Enter data beginning in line 9000. The first four items must be the number of rows, then the number of columns, then the number of layers, and finally the number of observations in each cell (n).

Then enter the observations by cell, starting with Layer 1, Row 1, Column 1; the Layer 1, Row 1, Column 2; etc.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

This program will handle up to an 8x8x8 analysis. To increase the number of levels allowed for any factor, change line 70 to read:

```
70  DIM  X(R+1, (C+1)*(L+1) )
```

, where R, C, L are the numbers of Rows, Columns, and Layers.

ACKNOWLEDGEMENTS: A. B. Jensen
MacMurray College

RUN

```
ANOVA3

9000  DATA 2,3,2,6
9001  DATA 27,22,45,18,76,33
9002  DATA 31,37,52,45,86,66
9003  DATA 55,62,76,85,104,126
9004  DATA 55,40,81,50,36,70
9005  DATA 77,76,98,68,42,104
9006  DATA 132,104,96,70,89,142
9007  DATA 61,39,76,60,46,59
9008  DATA 61,71,82,92,103,105
9009  DATA 140,122,99,92,68,101
9010  DATA 88,92,95,103,51,73
9011  DATA 100,120,120,131,89,76
9012  DATA 142,150,96,105,80,125


RUN
ANOVA3


SOURCE TABLE

          SUM OF SQUARES        DF      MEAN SQUARE             F

ROW       7667.31               1       7667.31                 15.9905
COLUMN    23630.1               2       11815.                  24.6408
LAYER     9730.19               1       9730.19                 20.2928
R*C       136.25                2       68.125                  .142078
R*L       8.6875                1       8.6875                  1.81182E-02
C*L       751.625               2       375.812                 .783774
R*C*L     223.75                2               111.875         .233321
W/GROUP   28769.4               60      479.491
-------------------------------------------------------------------
TOTAL     70917.3               71

DONE
```

LISTING

```
ANOVA3

1   REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2   REM
3   REM          ANOVA3:   THREE FACTORIAL ANALYSIS OF VARIANCE
4   REM
5   REM          36271 REV  A   6/73
6   REM
7   REM  ****  CONTRIBUTED PROGRAM  ************************************
10  REM  THREE-WAY ANALYSIS OF VARIANCE  MAY/1972
20  REM  GENERAL STATISTICS PROGRAM
30  REM  DATA STARTS AT 9000. #ROWS,COLUMNS,LAYERS,N/CELL
40  REM       WILL TAKE UP TO 8X8X8
50  REM       START AT L1,R1,C1: THEN L1,R1,C2, ETC.
60  REM
70  DIM X[9,81]
80  READ R,C,L,N
90  N1=N*R*C*L
100   C1=K1=M1=O1=P1=Q1=R1=S1=T1=0
110   MAT X=ZER[R+1,(C+1)*(L+1)]
120   F=(C+1)*L
130   FOR K=0 TO L-1
140   FOR I=1 TO R
150   FOR J=1 TO C
160   F2=K*(C+1)
170   FOR M=1 TO N
180   READ X
190   C1=C1+X↑2
200   X[I,F2+J]=X[I,F2+J]+X
210   NEXT M
220   X[I,F+J]=X[I,F+J]+X[I,F2+J]
230   K1=K1+X[I,F2+J]↑2/N
```

```
240  NEXT J
250  NEXT I
260  NEXT K
270  FOR K=0 TO L
280  FOR I=1 TO R
290  FOR J=1 TO C
300  F2=K*(C+1)
310  X[R+1,F2+J]=X[R+1,F2+J]+X[I,F2+J]
320  X[I,F2+C+1]=X[I,F2+C+1]+X[I,F2+J]
330  X[R+1,F2+C+1]=X[R+1,F2+C+1]+X[I,F2+J]
340  NEXT J
350  NEXT I
360  NEXT K
370  FOR K=0 TO L
380  F2=K*(C+1)
390  FOR I=1 TO R+1
400  FOR J=1 TO C+1
420  NEXT J
440  NEXT I
460  NEXT K
470  C2=X[R+1,F+C+1]↑2/N1
480  T=C1-C2
490  B=K1-C2
510  W=T-B
520  FOR I=1 TO R
530  M1=M1+X[I,F+C+1]↑2/(N1/R)
540  NEXT I
550  M1=M1-C2
560  FOR J=1 TO C
570  O1=O1+X[R+1,F+J]↑2/(N1/C)
580  NEXT J
590  O1=O1-C2
600  FOR K=0 TO L-1
610  F2=K*(C+1)
620  P1=P1+X[R+1,F2+C+1]↑2/(N1/L)
630  NEXT K
640  P1=P1-C2
650  FOR I=1 TO R
660  FOR J=1 TO C
670  Q1=Q1+X[I,J+F]↑2/(N1/(R*C))
680  NEXT J
690  NEXT I
700  Q1=Q1-M1-O1-C2
710  FOR K=0 TO L-1
720  F2=K*(C+1)
730  FOR I=1 TO R
740  R1=R1+X[I,F2+C+1]↑2/(N1/(L*R))
750  NEXT I
760  FOR J=1 TO C
770  S1=S1+X[R+1,F2+J]↑2/(N1/(C*L))
780  NEXT J
790  NEXT K
800  R1=R1-P1-M1-C2
810  S1=S1-O1-P1-C2
820  T1=B-M1-O1-P1-Q1-R1-S1
830  D1=R-1
840  D2=C-1
850  D3=L-1
860  D4=D1*D2
870  D5=D1*D3
880  D6=D2*D3
890  D7=D1*D2*D3
900  D8=R*C*L*(N-1)
910  D9=N1-1
920  Z1=M1
930  M1=M1/D1
940  M2=O1/D2
950  M3=P1/D3
960  M4=Q1/D4
970  M5=R1/D5
980  M6=S1/D6
990  M7=T1/D7
1000  M8=W/D8
1010  F1=M1/M8
1020  F2=M2/M8
1030  F3=M3/M8
1040  F4=M4/M8
1050  F5=M5/M8
```

```
1060  F6=M6/M8
1070  F7=M7/M8
1080  PRINT "SOURCE TABLE"
1085  PRINT "           SUM OF SQUARES        DF      MEAN SQUARE             F"
1086  PRINT
1090  PRINT "ROW      ";Z1,D1;M1,F1
1100  PRINT "COLUMN   "Q1,D2;M2,F2
1110  PRINT "LAYER    "P1,D3;M3,F3
1120  PRINT "R*C      "Q1,D4;M4,F4
1130  PRINT "R*L      "R1,D5;M5,F5
1140  PRINT "C*L      "S1,D6;M6,F6
1150  PRINT "R*C*L    "T1,D7,M7,F7
1160  PRINT "W/GROUP "W,D8;M8
1170  PRINT "-------------------------------------------------------------------"
1180  PRINT "TOTAL    "T,D9
1190  STOP
9000  DATA 2,3,2,6
9001  DATA 27,22,45,18,76,33
9002  DATA 31,37,52,45,86,66
9003  DATA 55,62,76,85,104,126
9004  DATA 55,40,81,50,36,70
9005  DATA 77,76,98,68,42,104
9006  DATA 132,104,96,70,89,142
9007  DATA 61,39,76,60,46,59
9008  DATA 61,71,82,92,103,105
9009  DATA 140,122,99,92,68,101
9010  DATA 88,92,95,103,51,73
9011  DATA 100,120,120,131,89,76
9012  DATA 142,150,96,105,80,125
9999  END
```

TITLE: ANALYSIS OF COVARIANCE

ANCOV
A410-36294A

DESCRIPTION:

This program computes an analysis of covariance table, F-ratio and adjusted means for groups of unequal size.

INSTRUCTIONS:

Enter data in line 400 in the following manner:

- first enter observation one for the first subject of group one, followed by observation two of the same subject. Observations for the second through nth subjects of group one follows the first subject. Each additional group follows the first group, one at a time. For example:

```
400  DATA X(1), Y(1), X(2), Y(2), ... X(n1), Y(n1)
401  DATA X(1), Y(1), X(2), Y(2), ... X(n2), Y(n2)
```

where:

$X(n_1)$ - the first observation of the last subject in group one.
$Y(n_1)$ - the second observation of the last subject in group one.
$X(n_2)$ - the first observation of the last subject in group two.
$Y(n_2)$ - the second observation of the last subject in group two.

SYSTEM SPECIFICATIONS:

2000A, Mark Sense Card Reader

SPECIAL CONSIDERATIONS:

For further reference, check STATISTICAL METHODS, by George W. Snedecor, pp. 318-320.

FOR INSTRUCTIONAL PURPOSES
Suitable Courses: Tests and Measurements, Statistics and Student Seminars.

Student Background Required: An understanding of the meaning of an F-ratio.

The analysis of covariance program computes the difference between two or more groups of any size that were not matched groups before the beginning of the experimental period.

ACKNOWLEDGEMENTS:

Dr. John Ingold
Goshen College

RUN

```
RUN
ANCOV

ANALYSIS OF COVARIANCE                NO. GROUPS?4

GROUP   1     NO. OBSERV.?3

GROUP   2     NO. OBSERV.?4

GROUP   3     NO. OBSERV.?5

GROUP   4     NO. OBSERV.?6


                 BETWEEN        THIN          TOTAL

  DF             3              14             17
SUM SQRS X       8.86108        124.75         133.611
SUM  XY          4.0835         106.083        110.167
SUM SQRS Y       39.4502        125.05         164.5
ADJ SS Y         38.8237        34.8401        73.6639
ADJ  DF          3              13             16
MEAN SQR         12.9412        2.68001        4.60399

F                4.8288


MEAN ADJ Y( 1    ) 8.71391
MEAN ADJ Y( 2    ) 9.65156
MEAN ADJ Y( 3    ) 12.8142
MEAN ADJ Y( 4    ) 11.0302

DONE
```

LISTING

```
ANCOV

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         ANCOV:   ANALYSIS OF COVARIANCE
4  REM
5  REM         36294 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  PRINT "ANALYSIS OF COVARIANCE";"                NO. GROUPS";
12  REM INSERT DATA AT LINES 400; FIRST X, THEN Y FOR EACH GROUP
15  INPUT G
16  PRINT
19  LET X2=Y2=C=N=M1=M2=B1=B2=B3=D1=E1=0
20  FOR I=1 TO G
25  PRINT "GROUP  ";I;" NO. OBSERV.";
26  INPUT N[I]
30  LET P2=X[I]=Y[I]=C[I]=0
35  FOR J=1 TO N[I]
40  READ X,Y
45  LET X[I]=X[I]+X
50  LET Y[I]=Y[I]+Y
55  LET C[I]=C[I]+X*Y
60  LET X2=X2+X↑2
65  LET Y2=Y2+Y↑2
70  LET C=C+X*Y
75  LET P2=P2+X↑2
80  NEXT J
85  LET N=N+N[I]
90  LET M1=M1+(X[I])
95  LET M2=M2+Y[I]
100  LET B1=B1+X[I]↑2/N[I]
105  LET B2=B2+X[I]*Y[I]/N[I]
110  LET B3=B3+Y[I]↑2/N[I]
115  LET E=C[I]-X[I]*Y[I]/N[I]
```

```
120  LET D=P2-X[I]↑2/N[I]
125  LET D1=D1+D
130  LET E1=E1+E
135  PRINT
140  NEXT I
145  LET T1=X2-M1↑2/N
150  LET B1=B1-M1↑2/N
155  LET I1=T1-B1
165  LET T3=Y2-M2↑2/N
170  LET B3=B3-M2↑2/N
175  LET I3=T3-B3
185  LET T2=C-M1*M2/N
190  LET B2=B2-M1*M2/N
195  LET I2=T2-B2
200  LET A2=T3-T2↑2/T1
205  LET A1=I3-I2↑2/I1
210  LET A3=A2-A1
218  PRINT
219  PRINT
220  PRINT "                    BETWEEN          THIN           TOTAL"
221  PRINT
225  PRINT "  DF          ",G-1,N-G,N-1
230  PRINT "SUM SQRS X",B1,I1,T1
235  PRINT "SUM  XY    ",B2,I2,T2
240  PRINT "SUM SQRS Y",B3,I3,T3
245  PRINT "ADJ SS Y   ",A3,A1,A2
250  PRINT "ADJ  DF    ",G-1,N-G-1,N-2
255  PRINT "MEAN SQR   ",A3/(G-1),A1/(N-G-1),A2/(N-2)
256  PRINT
260  PRINT "F          ",(A3/(G-1))/(A1/(N-G-1))
265  LET A=E1/D1
270  PRINT
271  PRINT
280  FOR I=1 TO G
285  PRINT "MEAN ADJ Y(";I;")";Y[I]/N[I]-A*(X[I]/N[I]-M1/N)
290  NEXT I
400  DATA 4,6,8,9,14,11
401  DATA 5,5,7,7,12,13,9,12
402  DATA 10,13,8,12,11,16,4,9,7,11
403  DATA 11,11,12,17,8,10,10,13,7,9,10,11
1000  END
```

STAT19

TITLE: KRUSKAL - WALLIS ONE WAY ANALYSIS OF VARIANCE F410-36607A

DESCRIPTION:

The Kruskal-Wallis one way analysis of variance by ranks is an extremely useful non-parametric test for deciding whether K independent samples are from different populations. The Kruskal-Wallis technique tests the null hypothesis that the K samples came from the same population or from identical populations with respect to averages.

The data is present in a table having K columns (maximum of 10) each column representing one set, or sample, from a total of N observations.

INSTRUCTIONS:

Enter the data in lines 2000-9998. Data should be entered by sample (or column) and each sample should be preceded by the number of observations in that sample. Type 'RUN' and answer the questions as they appear. The computer will print out the value of H to be compared to Chi-Square. If H is less than or equal to the value of Chi-Square at the given degrees of freedom then the null hypothesis should be rejected.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

There may only be up to 500 observations in 10 samples. (Maximum - 50 per sample).

ACKNOWLEDGEMENTS:

Larry Robbins
Babson College

RUN

```
2000  DATA 10,2,2.8,3.3,3.2,4.4,3.6,1.9,3.3,2.8,1.1
2010  DATA 8,3.5,2.8,3.2,3.5,2.3,2.4,2,1.6
2020  DATA 10,3.3,3.6,2.6,3.1,3.2,3.3,2.9,3.4,3.2,3.2

2030  DATA 8,3.2,3.3,3.3,2.2,2.9,3.3,2.5,2.6,2.8
2040  DATA 6,2.6,2.6,2.9,2,2,2.1
2050  DATA 4,3.1,2.9,3.1,2.5
2060  DATA 6,2.6,2.2,2.2,2.2,2.5,1.2,1.2
2070  DATA 4,2.5,2.4,3,1.5
9999  END
```

```
RUN
STAT19

TOTAL NUMBER OF OBSERVATIONS    ?56
NUMBER OF SAMPLES   ?8



  YOUR ANSWER WILL TAKE A FEW MINUTES...
         PLEASE WAIT..........



DO YOU WANT TO SEE THE RANKED SCORES????YES


                              RANKED SCORES


  1.0     2.0     3.0     4.0     5.0     6.0     7.0     8.0
-------------------------------------------------------------------

  8.5    52.5    47.5    41.0    23.0    36.0    23.0    18.5
 27.5    27.5    54.5    47.5    23.0    31.5    12.5    15.5
 47.5    41.0    23.0    41.0    31.5    36.0    12.5    34.0
 41.0    52.5    36.0    31.5     8.5    18.5    18.5     4.0
 56.0    14.0    41.0    47.5     8.5     0.0     2.5     0.0
 54.5    15.5    47.5    18.5    11.0     0.0     2.5     0.0
  6.0     8.5    31.5    23.0     0.0     0.0     0.0     0.0
 47.5     5.0    51.0    27.5     0.0     0.0     0.0     0.0
 27.5     0.0    41.0     0.0     0.0     0.0     0.0     0.0
  1.0     0.0    41.0     0.0     0.0     0.0     0.0     0.0

NO. OF NO'S IN COLUMN
 10.0     8.0    10.0     8.0     6.0     4.0     6.0     4.0

SUM OF NO'S IN COLUMN
 317.0   216.5   414.0   277.5   105.5   122.0    71.5    72.0


THE VALUE OF H TO BE COMPARED TO CHI SQUARE IS 18.4639
DEGREES OF FREEDOM ARE  7

DONE
```

LISTING

```
STAT19

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         STAT19:   KRUSKAL-WALLIS ONE WAY ANALYSIS OF VARIANCE
4  REM
5  REM         36607 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
```

```
10  REM PROGRAM FOR KRUSKAL-WALLIS ONE-WAY ANALYSIS OF VARIANCE
20  REM BY LARRY ROBBINS --BABSON COLLEGE
30  T=T1=T2=T3=0
40  PRINT "TOTAL NUMBER OF OBSERVATIONS    ";
50  INPUT N
60  PRINT "NUMBER OF SAMPLES   ";
70  INPUT K
80  DIM A[50,10],B[50,10],D[75],E[75]
90  PRINT ""
100  PRINT "  YOUR ANSWER WILL TAKE A FEW MINUTES..."
110  PRINT "           PLEASE WAIT..........."
120  PRINT ""
130  MAT A=ZER[50,K]
140  MAT B=ZER[52,K]
150  MAT D=ZER[N+1]
160  MAT E=ZER[N+1]
170  FOR X1=1 TO K
180  READ C
190  FOR X2=1 TO C
200  READ A[X2,X1]
210  NEXT X2
220  NEXT X1
230  X3=0
240  FOR X2=1 TO K
250  FOR X1=1 TO 50
260  IF A[X1,X2]=0 THEN 300
270  X3=X3+1
280  D[X3]=A[X1,X2]
290  NEXT X1
300  NEXT X2
310  FOR X4=1 TO N
320  IF D[X4]>D[X4+1] THEN 350
330  NEXT X4
340  GOTO 390
350  X5=D[X4]
360  D[X4]=D[X4+1]
370  D[X4+1]=X5
380  GOTO 310
390  FOR X1=1 TO N
400  D[X1]=D[X1+1]
410  E[X1]=X1
420  NEXT X1
430  D[N+1]=E[N+1]=0
440  X7=0
450  X7=X7+1
460  IF X7 >= N THEN 580
470  IF D[X7]=D[X7+1] THEN 490
480  GOTO 450
490  X8=X7
500  X8=X8+1
510  IF D[X8]=D[X8+1] THEN 500
520  X9=(X7-1)+((((X8-X7)+1)/2)+.5)
530  FOR J=X7 TO X8
540  E[J]=X9
550  NEXT J
560  X7=X8
570  GOTO 450
580  FOR X1=1 TO K
590  FOR X2=1 TO 50
600  IF A[X2,X1]=0 THEN 660
610  FOR X3=1 TO N+1
620  IF A[X2,X1]=D[X3] THEN 640
630  NEXT X3
640  B[X2,X1]=E[X3]
650  NEXT X2
660  NEXT X1
670  FOR X1=1 TO K
680  FOR X2=1 TO 50
690  IF B[X2,X1]=0 THEN 720
700  B[51,X1]=B[51,X1]+B[X2,X1]
710  NEXT X2
720  NEXT X1
730  X=0
740  T=0
750  X=X+1
760  IF X >= N THEN 910
770  IF D[X]=D[X+1] THEN 750
780  FOR X1=1 TO K
```

```
790  FOR X2=1 TO 50
800  IF A[X2,X1]=0 THEN 830
810  IF A[X2,X1]=D[X] THEN 850
820  NEXT X2
830  NEXT X1
840  GOTO 870
850  GOTO 880
860  GOTO 820
870  IF T=1 THEN 740
880  T1=((T↑3)-T)
890  T3=T3+T1
900  GOTO 740
910  IF T3=0 THEN 940
920  T2=1-((T3)/((N↑3)-N))
930  GOTO 950
940  T2=1-(0/((N↑3)-N))
950  B=0
960  FOR J=1 TO K
970  FOR G=1 TO 50
980  IF B[G,J]=0 THEN 1000
990  NEXT G
1000  B[50,J]=G-1
1010  NEXT J
1020  FOR J=1 TO K
1030  B=B+((B[51,J]↑2)/B[50,J])
1040  NEXT J
1050  H=(12/(N*(N+1)))*(B)-(3*(N+1))
1060  H=H/T2
1070  PRINT "DO YOU WANT TO SEE THE RANKED SCORES???":
1080  DIM A$[5]
1090  INPUT A$
1100  IF A$="NO" THEN 1430
1110  PRINT "                                    RANKED SCORES"
1120  Z=0
1130  FOR J=1 TO K
1140  Z=B[50,J] MAX Z
1150  NEXT J
1160  DIM C[1,20]
1170  MAT C=ZER[1,K]
1180  FOR X1=1 TO K
1190  C[1,X1]=X1
1200  NEXT X1
1210  MAT  PRINT  USING "3D.1D3X";C
1220  PRINT "-------------------------------------------------------------------------"
1230  FOR X1=1 TO Z
1240  FOR X2=1 TO K
1250  C[1,X2]=B[X1,X2]
1260  NEXT X2
1270  MAT  PRINT  USING "3D.1D3X";C
1280  NEXT X1
1290  PRINT
1300  PRINT "NO. OF NO'S IN COLUMN"
1310  FOR X2=1 TO K
1320  C[1,X2]=B[50,X2]
1330  NEXT X2
1340  MAT  PRINT  USING "3D.1D3X";C
1350  PRINT
1360  PRINT "SUM OF NO'S IN COLUMN"
1370  FOR X2=1 TO K
1380  C[1,X2]=B[51,X2]
1390  NEXT X2
1400  MAT  PRINT  USING "4D.1D2X";C
1410  PRINT
1420  PRINT
1430  PRINT "THE VALUE OF H TO BE COMPARED TO CHI SQUARE IS";H
1440  PRINT "DEGREES OF FREEDOM ARE ";K-1
1450  END
```

ACTFIL

TITLE: ACTIVE FILTER DESIGN F513-36293A

DESCRIPTION:

Designs Butterworth or Tchebyscheff active filters with roll-offs of 12, 24, or 36 db per octave. (48 for Butterworth)

INSTRUCTIONS:

The user is asked to enter:

1. Type Butterworth or Tchebyscheff
2. High or low pass
3. Cut-off frequency in hertz
4. Db of attenuation per octave, and
5. The value of C for high pass or R for low pass

If the user wishes a schematic, it is printed out on the graphic display terminal or teletype.

SYSTEM SPECIFICATIONS:

Single Terminal BASIC, 2000F and Graphic Display Terminal or Teletype

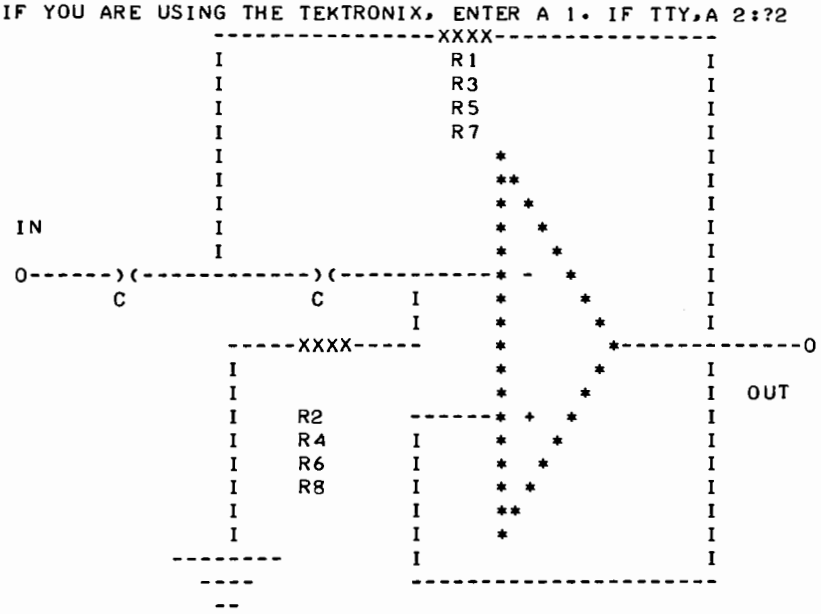
SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Brian L. Bardsley
Woods Hole Oceanographic Institution

```
RUN
ACTFIL

THIS PROGRAM WILL DESIGN BUTTERWORTH OR TCHEBYSCHEFF ACTIVE
FILTERS WITH A ROLL OFF OF 12,24,36,OR 48 DB PER OCTAVE FOR
BUTTERWORTH OR 12,24,OR 36 DB FOR TCHEBYSCHEFF. IT DOES NOT
ALLOW FOR THE ADDITION OF ANY GAIN IN THE FILTERS
YOU WILL BE REQUIRED TO ENTER THE FOLLOWING INFORMATION:

TCHEBYSCHEFF OR BUTTERWORTH
HIGH OR LOW PASS
CUT-OFF FREQUENCY
DB OF ATTENUATION PER OCTAVE
VALUE OF  C  FOR HIGH PASS OR  R  FOR LOW PASS

BE SURE TO PUSH RETURN AFTER EVERY ENTRY

IF YOURE READY,LETS BEGIN


ENTER A  1  FOR TCHEBYSCHEFF,  2  FOR BUTTERWORTH:?2
ENTER  1  FOR LOW PASS,  2  FOR HIGH PASS:?2
ENTER CUT OFF FREQUENCY IN HERTZ:?1000
ENTER  C  IN MICROFARADS:?.001
ENTER DB OF ATTENUATION PER OCTAVE:?48
R1= 31066.9
R2= 816175.
R3= 88471.4
R4= 286601.
R5= 132404.
R6= 191504.
R7= 156178.
R8= 162353.
DO YOU WANT A SCHEMATIC? 1 IF YES, 2 IF NO:?1
IF YOU ARE USING THE TEKTRONIX, ENTER A 1. IF TTY,A 2:?2
```




```
 THIS REPRESENTS ONE 12 DB SECTION.
FOR ONE SECTION, USE R1,R2.
FOR 2 SECTIONS, USE R1,R2 FOR THE FIRST AND R3,R4 FOR
THE SECOND.
FOR 3,USE R1,R2 FOR THE FIRST-R3,R4 FOR THE SECOND-ETC.
 THE VALUE OF C YOU SELECTED AT THE START OF THE
 DESIGN, IS USED FOR BOTH VALUES OF  C
IF YOU HAVE MORE TO DESIGN,ENTER 1. IF NOT, 2:?2

DONE
```

LISTING

```
ACTFIL

1  REM  ****  HP BASIC PROGRAM LIBRARY  **********************************
2  REM
3  REM         ACTFIL:   ACTIVE FILTER DESIGN
4  REM
5  REM         36293 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
8  PRINT "THIS PROGRAM WILL DESIGN BUTTERWORTH OR TCHEBYSCHEFF ACTIVE"
9  PRINT "FILTERS WITH A ROLL OFF OF 12,24,36,OR 48 DB PER OCTAVE FOR"
10  PRINT "BUTTERWORTH OR 12,24,OR 36 DB FOR TCHEBYSCHEFF. IT DOES NOT"
11  PRINT "ALLOW FOR THE ADDITION OF ANY GAIN IN THE FILTERS"
12  PRINT "YOU WILL BE REQUIRED TO ENTER THE FOLLOWING INFORMATION:"
13  PRINT
14  PRINT "TCHEBYSCHEFF OR BUTTERWORTH"
15  PRINT "HIGH OR LOW PASS"
16  PRINT "CUT-OFF FREQUENCY"
17  PRINT "DB OF ATTENUATION PER OCTAVE"
18  PRINT "VALUE OF  C  FOR HIGH PASS OR  R  FOR LOW PASS"
19  PRINT
20  PRINT "BE SURE TO PUSH RETURN AFTER EVERY ENTRY"
21  PRINT
22  PRINT "IF YOURE READY,LETS BEGIN"
23  PRINT
24  PRINT
25  GOTO 35
30  PRINT "NOT A VALID ENTRY. TRY AGAIN"
35  PRINT "ENTER A  1  FOR TCHEBYSCHEFF,  2  FOR BUTTERWORTH:";
40  INPUT T
45  IF T=1 THEN 695
50  IF T=2 THEN 60
55  GOTO 30
60  GOTO 70
65  PRINT "NOT A VALID ENTRY. TRY AGAIN"
70  PRINT "ENTER  1  FOR LOW PASS,  2  FOR HIGH PASS:";
75  INPUT P
80  IF P=1 THEN 95
85  IF P=2 THEN 395
90  GOTO 65
95  GOTO 105
100  PRINT "NOT A VALID ENTRY. TRY AGAIN"
105  PRINT "ENTER CUT-OFF FREQUENCY IN HERTZ:";
110  INPUT F
115  PRINT "ENTER  R  IN OHMS:";
120  INPUT R
125  GOTO 135
130  PRINT "NOT A VALID ENTRY. TRY AGAIN"
135  PRINT "ENTER DB OF ATTENUATION PER OCTAVE:";
140  INPUT A
145  IF A=12 THEN 170
150  IF A=24 THEN 195
155  IF A=36 THEN 240
160  IF A=48 THEN 305
165  GOTO 130
170  LET C2=.707/((6.28*F)*R)
175  LET C1=1/(R*(6.28*F)*.707)
180  PRINT "C1=";C1
185  PRINT "C2=";C2
190  GOTO 1225
195  LET C2=.3827/((6.28*F)*R)
200  LET C1=1/(R*(6.28*F)*.3827)
205  LET C4=.9239/((6.28*F)*R)
210  LET C3=1/(R*(6.28*F)*.9239)
215  PRINT "C1=";C1
220  PRINT "C2=";C2
225  PRINT "C3=";C3
230  PRINT "C4=";C4
235  GOTO 1225
240  LET C2=.2588/((6.28*F)*R)
245  LET C1=1/(R*(6.28*F)*.2588)
250  LET C4=.7071/((6.28*F)*R)
255  LET C3=1/(R*(6.28*F)*.7071)
260  LET C6=.9659/((6.28*F)*R)
265  LET C5=1/(R*(6.28*F)*.9659)
```

```
270  PRINT "C1=";C1
275  PRINT "C2=";C2
280  PRINT "C3=";C3
285  PRINT "C4=";C4
290  PRINT "C5=";C5
295  PRINT "C6=";C6
300  GOTO 1225
305  LET C2=.1951/((6.28*F)*R)
310  LET C1=1/(R*(6.28*F)*.1951)
315  LET C4=.5556/((6.28*F)*R)
320  LET C3=1/(R*(6.28*F)*.5556)
325  LET C6=.8315/((6.28*F)*R)
330  LET C5=1/(R*(6.28*F)*.8315)
340  LET C8=.9808/((6.28*F)*R)
345  LET C7=1/(R*(6.28*F)*.9808)
350  PRINT "C1=";C1
355  PRINT "C2=";C2
360  PRINT "C3=";C3
365  PRINT "C4=";C4
370  PRINT "C5=";C5
375  PRINT "C6=";C6
380  PRINT "C7=";C7
385  PRINT "C8=";C8
390  GOTO 1225
395  PRINT "ENTER CUT OFF FREQUENCY IN HERTZ:";
400  INPUT F
405  PRINT "ENTER  C  IN MICROFARADS:";
410  INPUT C
415  LET C=C*.000001
420  GOTO 430
425  PRINT "NOT A VALID ENTRY. TRY AGAIN"
430  PRINT "ENTER DB OF ATTENUATION PER OCTAVE:";
435  INPUT A
440  IF A=12 THEN 465
445  IF A=24 THEN 490
450  IF A=36 THEN 540
455  IF A=48 THEN 605
460  GOTO 425
465  LET R1=(.707/(12.56*F*C))*2
470  LET R2=1/(R1*(C↑2)*((6.28*F)↑2))
475  PRINT "R1=";R1
480  PRINT "R2=";R2
485  GOTO 1225
490  LET R1=(.3827/(12.56*F*C))*2
495  LET R2=1/(R1*(C↑2)*((6.28*F)↑2))
500  LET R3=(.9239/(12.56*F*C))*2
505  LET R4=1/(R3*(C↑2)*((6.28*F)↑2))
515  PRINT "R1=";R1
520  PRINT "R2=";R2
525  PRINT "R3=";R3
530  PRINT "R4=";R4
535  GOTO 1225
540  LET R1=(.2588/(12.56*F*C))*2
545  LET R2=1/(R1*(C↑2)*((6.28*F)↑2))
550  LET R3=(.7071/(12.56*F*C))*2
555  LET R4=1/(R3*(C↑2)*((6.28*F)↑2))
560  LET R5=(.9656/(12.56*F*C))*2
565  LET R6=1/(R5*(C↑2)*((6.28*F)↑2))
570  PRINT "R1=";R1
575  PRINT "R2=";R2
580  PRINT "R3=";R3
585  PRINT "R4=";R4
590  PRINT "R5=";R5
595  PRINT "R6=";R6
600  GOTO 1225
605  LET R1=(.1951/(12.56*F*C))*2
610  LET R2=1/(R1*(C↑2)*((6.28*F)↑2))
615  LET R3=(.5556/(12.56*F*C))*2
620  LET R4=1/(R3*(C↑2)*((6.28*F)↑2))
625  LET R5=(.8315/(12.56*F*C))*2
630  LET R6=1/(R5*(C↑2)*((6.28*F)↑2))
635  LET R7=(.9808/(12.56*F*C))*2
640  LET R8=1/(R7*(C↑2)*((6.28*F)↑2))
645  PRINT "R1=";R1
650  PRINT "R2=";R2
655  PRINT "R3=";R3
660  PRINT "R4=";R4
665  PRINT "R5=";R5
```

```
670  PRINT "R6=";R6
675  PRINT "R7=";R7
680  PRINT "R8=";R8
685  GOTO 1225
690  PRINT "NOT A VALID ENTRY. TRY AGAIN"
695  PRINT "ENTER A  1  FOR LOW PASS.  2  FOR HIGH PASS:";
700  INPUT P
705  IF P=1 THEN 720
710  IF P=2 THEN 1005
715  GOTO 690
720  PRINT "ENTER CUT-OFF FREQUENCY IN HERTZ:";
725  INPUT F
730  PRINT "ENTER  R  IN OHMS:";
735  INPUT R
740  GOTO 750
745  PRINT "NOT A VALID ENTRY,TRY AGAIN"
750  PRINT "ENTER DB OF ATTENUATION PER OCTAVE:";
755  INPUT A
760  IF A=12 THEN 785
765  IF A=24 THEN 820
770  IF A=36 THEN 890
780  GOTO 745
785  LET B1=.579/(12.56*(F*1.231)*R)
790  LET B2=1/(6.28*(F*1.231)*R)
795  LET C2=B1+(B2*(SQR((.579↑2)/4)))
800  LET C1=1/((R↑2)*C2*((6.28*(F*1.231))↑2))
805  PRINT "C1=";C1
810  PRINT "C2=";C2
815  GOTO 1225
820  LET M1=.17/(12.56*(F*1.031)*R)
825  LET M2=1/(6.28*(F*1.031)*R)
830  LET C2=M1+(M2*(SQR((.17↑2)/4)))
835  LET C1=1/((R↑2)*C2*((6.28*(F*1.031))↑2))
840  LET N1=.709/(12.56*(F*.597)*R)
850  LET N2=1/(6.28*(F*.579)*R)
855  LET C4=N1+(N2*(SQR((.709↑2)/4)))
860  LET C3=1/((R↑2)*C4*((6.28*(F*1.231))↑2))
865  PRINT "C1=";C1
870  PRINT "C2=";C2
875  PRINT "C3=";C3
880  PRINT "C4=";C4
885  GOTO 1225
890  LET M1=.077/(12.56*(F*1.011)*R)
895  LET M2=1/(6.28*(F*1.011)*R)
900  LET C2=M1+(M2*(SQR((.077↑2)/4)))
905  LET C1=1/((R↑2)*C2*((6.28*(F*1.011))↑2))
910  LET N1=.276/(12.56*(F*.768)*R)
915  LET N2=1/(6.28*(F*.768)*R)
920  LET C4=N1+(N2*(SQR((.276↑2)/4)))
925  LET C3=1/((R↑2)*C4*((6.28*(F*.768))↑2))
930  LET P1=.732/(12.56*(F*.396)*R)
935  LET P2=1/(6.28*(F*.396)*R)
940  LET C6=P1+(P2*(SQR((.732↑2)/4)))
945  LET C5=1/((R↑2)*C6*((6.28*(F*.396))↑2))
950  PRINT "C1=";C1
955  PRINT "C2=";C2
960  PRINT "C3=";C3
965  PRINT "C4=";C4
970  PRINT "C5=";C5
975  PRINT "C6=";C6
980  GOTO 1225
1005  PRINT "ENTER CUT-OFF FREQUENCY IN HERTZ:";
1010  INPUT F
1015  PRINT "ENTER  C  IN MICROFARADS:";
1020  INPUT C
1025  LET C=C*.000001
1030  PRINT "ENTER DB OF ATTENUATION:";
1035  INPUT A
1040  IF A=12 THEN 1065
1045  IF A=24 THEN 1090
1050  IF A=36 THEN 1135
1060  GOTO 1030
1065  LET R1=(.579/(2*(6.28*(F*.815))*C))*2
1070  LET R2=1/(R1*(C↑2)*((6.28*(F*.815))↑2))
1075  PRINT "R1=";R1
1080  PRINT "R2=";R2
1085  GOTO 1225
1090  LET R1=(.17/(2*(6.28*(F*.97))*C))*2
```

```
1095  LET R2=1/(R1*(C↑2)*((6.28*(F*.97))↑2))
1100  LET R3=(.709/(2*(6.28*(F*1.675))*C))*2
1105  LET R4=1/(R3*(C↑2)*((6.28*(F*1.675))↑2))
1110  PRINT "R1=";R1
1115  PRINT "R2=";R2
1120  PRINT "R3=";R3
1125  PRINT "R4=";R4
1130  GOTO 1225
1135  LET R1=(.077/(2*(6.28*(F*.989))*C))*2
1140  LET R2=1/(R1*(C↑2)*((6.28*(F*.989))↑2))
1145  LET R3=(.276/(2*(6.28*(F*1.31))*C))*2
1150  LET R4=1/(R3*(C↑2)*((6.28*(F*1.31))↑2))
1155  LET R5=(.732/(2*(6.28*(F*2.52))*C))*2
1160  LET R6=1/(R5*(C↑2)*((6.28*(F*2.52))↑2))
1165  PRINT "R1=";R1
1170  PRINT "R2=";R2
1175  PRINT "R3=";R3
1180  PRINT "R4=";R4
1185  PRINT "R5=";R5
1190  PRINT "R6=";R6
1195  GOTO 1225
1220  PRINT "NOT A VALID ENTRY. TRY AGAIN"
1225  PRINT "DO YOU WANT A SCHEMATIC? 1 IF YES, 2 IF NO:";
1226  INPUT W
1227  IF W=1 THEN 1900
1228  IF W=2 THEN 1300
1229  GOTO 1225
1290  PRINT "TRY AGAIN"
1300  PRINT "IF YOU HAVE MORE TO DESIGN,ENTER 1. IF NOT, 2:";
1310  INPUT Z
1320  IF Z=1 THEN 35
1330  IF Z=2 THEN 5000
1340  GOTO 1290
1900  PRINT "IF YOU ARE USING THE TEKTRONIX, ENTER A 1. IF TTY,A 2:";
1901  INPUT K
1902  IF K=2 THEN 1975
1903  IF K=1 THEN 1910
1904  GOTO 1900
1910  PRINT "GET A COPY OF THE CIRCUIT VALUES NOW. AFTER YOU HAVE YOUR"
1915  PRINT "COPY,PRESS ERASE THEN HOME ON THE TEKTRONIX. THIS WILL"
1920  PRINT "INSURE THAT THE ENTIRE SCHEMATIC IS PRINTED ON ONE"
1925  PRINT "PAGE. AFTER YOU HAVE PRESSED ERASE AND HOME, ENTER A"
1930  PRINT " 1 AND THE SCHEMATIC WILL BE PRINTED:";
1935  INPUT H
1940  IF H=1 THEN 1975
1945  IF H#1 THEN 1950
1950  PRINT "TRY AGAIN"
1960  GOTO 1935
1975  IF P=1 THEN 2000
1976  IF P=2 THEN 3000
1999  REM *****LOW PASS PRINT ROUTINE********
2000  PRINT "                 -----------------)(-----------------"
2010  PRINT "                 I                 C1                I"
2020  PRINT "                 I                 C3                I"
2030  PRINT "                 I                 C5                I"
2040  PRINT "                 I                 C7                I"
2050  PRINT "                 I                     *             I"
2060  PRINT "                 I                     **            I"
2070  PRINT "                 I                     * *           I"
2080  PRINT ". IN             I                     *  *          I"
2090  PRINT "                 I                     *   *         I"
2100  PRINT " 0-------XXXX----------XXXX----------* -  *         I"
2110  PRINT "           R             R      I     *      *       I"
2120  PRINT "                               I     *       *      I"
2130  PRINT "                          ---)(---   *        *-----------0"
2140  PRINT "                          I          *       *       I"
2150  PRINT "                          I  C2      *      *        I"
2160  PRINT "                          I  C4    ----* +  *        I"
2170  PRINT "                          I  C6    I   *   *         I"
2180  PRINT "                          I  C8    I   *  *          I"
2190  PRINT "                          I        I   * *           I"
2200  PRINT "                          I        I   **            I"
2210  PRINT "                          I        I   *             I"
2220  PRINT "                          I        I                 I"
2230  PRINT "                        -------    -----------------"
2240  PRINT "                         ----- "
2250  PRINT "                          ---  "
2260  PRINT "                           -   "
```

```
2270  PRINT
2280  PRINT
2290  PRINT "THIS REPRESENTS ONE 12 DB SECTION. FOR ONE SECTION"
2300  PRINT "USE C1 AND C2. FOR TWO SECTIONS. USE C1 AND C2 FOR"
2310  PRINT "THE FIRST--C3 AND C4 FOR THE SECOND. FOR THREE."
2320  PRINT "USE C1 AND C2 FOR THE FIRST. ETC."
2330  PRINT
2340  PRINT
2350  PRINT "THE VALUE OF R YOU SELECTED AT THE START OF THE "
2360  PRINT "FILTER DESIGN IS USED FOR BOTH  VALUES OF  R."
2370  GOTO 1300
2999  REM  ****HIGH PASS PRINT ROUTINE*******
3000  PRINT "                ---------------XXXX---------------"
3010  PRINT "                I                 R1                I"
3020  PRINT "                I                 R3                I"
3030  PRINT "                I                 R5                I"
3040  PRINT "                I                 R7                I"
3050  PRINT "                I                    *              I"
3060  PRINT "                I                    **             I"
3070  PRINT "                I                    * *            I"
3080  PRINT " IN             I                    *  *           I"
3090  PRINT "                I                    *   *          I"
3100  PRINT " 0------)(------------)(-----------*  -  *          I"
3110  PRINT "        C              C      I     *     *         I"
3120  PRINT "                              I     *       *       I"
3130  PRINT "                 -----XXXX-----     *        *-------------0"
3140  PRINT "                 I                  *       *       I"
3150  PRINT "                 I                  *      *        I  OUT"
3160  PRINT "                 I    R2       ------* +  *         I"
3170  PRINT "                 I    R4       I     *   *          I"
3180  PRINT "                 I    R6       I     *  *           I"
3190  PRINT "                 I    R8       I     * *            I"
3200  PRINT "                 I             I     **             I"
3210  PRINT "                 I             I     *              I"
3220  PRINT "             --------          I                    I"
3230  PRINT "               ----            ---------------------"
3240  PRINT "                -- "
3250  PRINT
3260  PRINT
3270  PRINT
3340  PRINT " THIS REPRESENTS ONE 12 DB SECTION."
3350  PRINT "FOR ONE SECTION. USE R1.R2."
3360  PRINT "FOR 2 SECTIONS. USE R1.R2 FOR THE FIRST AND R3.R4 FOR"
3370  PRINT "THE SECOND."
3380  PRINT "FOR 3.USE R1.R2 FOR THE FIRST-R3.R4 FOR THE SECOND-ETC."
3400  PRINT " THE VALUE OF C YOU SELECTED AT THE START OF THE"
3410  PRINT " DESIGN. IS USED FOR BOTH VALUES OF  C"
3420  GOTO 1300
5000  END
```

TITLE: CONVERTS ENGLISH TO METRIC, METRIC TO ENGLISH

METRIC
F514-36635A

DESCRIPTION:

This program converts 19 metric measurements into their equivalent English measurements and vice versa.

INSTRUCTIONS:

If the user responds "Y" or "YES" to the prompt, INSTRUCTIONS?, the program prints out a table of the 19 metric measurements, and assigns each conversion a number. The user then enters his choice. An entry of "20" to the "choice" prompt terminates execution of the program.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Terry Von Gease
HP, Data Systems

RUN

```
RUN
METRIC


INSTRUCTIONS ?Y

+ TO CONVERT FROM                  TO
- TO CONVERT TO                    FROM

 1            INCHES                MILLIMETERS
 2            FEET                  METERS
 3            YARDS                 METERS
 4            MILES                 KILOMETERS
 5            SQUARE INCHES         SQUARE CENTIMETERS
 6            SQUARE FEET           SQUARE METERS
 7            SQUARE YARDS          SQUARE METERS
 8            ACRES                 HECTARES
 9            CUBIC INCHES          MILLILITERS
10            CUBIC FEET            CUBIC METERS
11            CUBIC YARDS           CUBIC METERS
12            QUARTS                LITERS
13            GALLONS               LITERS
14            OUNCES                GRAMS
15            POUNDS (MASS)         KILOGRAMS
16            POUNDS (FORCE)        NEWTONS
17            P.S.I.                KILOPASCALS
18            HORSEPOWER            KILOWATTS
19            BTU                   KILOJOULE
20            END THE PROGRAM

YOUR CHOICE ?1

ENTER THE VALUE IN INCHES ?12

    12.0000 INCHES =    304.8000 MILLIMETERS

YOUR CHOICE ?-1

ENTER THE VALUE IN MILLIMETERS ?304.8000

   304.8000 MILLIMETERS =     12.0000 INCHES

YOUR CHOICE ?16

ENTER THE VALUE IN POUNDS (FORCE) ?56

    56.0000 POUNDS (FORCE) =    249.0880 NEWTONS

YOUR CHOICE ?-9

ENTER THE VALUE IN MILLILITERS ?10

    10.0000 MILLILITERS =      0.6102 CUBIC INCHES

YOUR CHOICE ?20

DONE
```

LISTING

```
METRIC

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         METRIC:   CONVERTS ENGLISH TO METRIC, METRIC TO ENGLISH
4  REM
5  REM         36635 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  DATA 25.4,.3048,.9144,1.609,6.4516,.0929,.836,.405,16.387
20  DATA .0283,.765,.946,3.784,28.35,.454,4.448,6.895,.746,1.055
30  DIM A$[72],B$[72],C$[72]
```

```
40  PRINT LIN(1)"INSTRUCTIONS ":
50  INPUT A$
60  IF A$[1,1]="N" THEN 260
70  DATA "INCHES","MILLIMETERS","FEET","METERS","YARDS","METERS","MILES"
80  DATA "KILOMETERS","SQUARE INCHES","SQUARE CENTIMETERS","SQUARE FEET"
90  DATA "SQUARE METERS","SQUARE YARDS","SQUARE METERS","ACRES","HECTARES"
100  DATA "CUBIC INCHES","MILLILITERS","CUBIC FEET","CUBIC METERS"
110  DATA "CUBIC YARDS","CUBIC METERS","QUARTS","LITERS","GALLONS"
120  DATA "LITERS","OUNCES","GRAMS","POUNDS (MASS)","KILOGRAMS"
130  DATA "POUNDS (FORCE)","NEWTONS","P.S.I.","KILOPASCALS"
140  DATA "HORSEPOWER","KILOWATTS","BTU","KILOJOULE"
150  PRINT LIN(1)"+ TO CONVERT FROM";TAB(35);"TO"
160  PRINT "- TO CONVERT TO";TAB(35);"FROM"
170  PRINT
180  RESTORE 70
190  FOR A=1 TO 19
200  READ A$,B$
210  PRINT  USING 220:A
220  IMAGE#,DDXX9X
230  PRINT A$;TAB(35);B$
240  NEXT A
250  PRINT "20            END THE PROGRAM"
260  PRINT LIN(1)"YOUR CHOICE ":
270  INPUT A
280  IF A=20 OR A=-20 THEN 560
290  IF A=0 OR A>19 OR A<-19 THEN 260
300  RESTORE 10
310  FOR B=1 TO ABS(A)
320  READ C
330  NEXT B
340  RESTORE 70
350  FOR B=1 TO ABS(A)
360  READ A$,B$
370  NEXT B
380  IF A<0 THEN 490
390  PRINT LIN(1)"ENTER THE VALUE IN ";A$;" ";
400  INPUT D
410  E=D*C
420  PRINT
430  PRINT  USING 440:D
440  IMAGE#,6D.4DX
450  PRINT A$;" = ";
460  PRINT  USING 440:E
470  PRINT B$
480  GOTO 260
490  C$=A$
500  A$=B$
510  B$=C$
520  PRINT LIN(1)"ENTER THE VALUE IN ";A$;" ";
530  INPUT D
540  E=D/C
550  GOTO 420
560  END
```

CTC4
A706-36213A

TITLE: CTC PAYROLL PROGRAMS

DESCRIPTION:

These CTC Payroll Programs are part of a total accounting system written by Computer Terminal Corporation for the HP 2000A. (See A717-36212 CTC Projection Programs, A708-36210 CTC Manufacturing Parts Control, A711-36214 CTC Accounts Receivable, A708-36211 CTC Inventory Control for Finished Products.) Abstracts of the 33 payroll programs are on the following page along with an index to the documentation.

INSTRUCTIONS:

Order option DØØ

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

This package was written for a 2000A with a non-standard disc. The file structure has 200 physical records per file.

Teleprinter output is directed to a 132 column AB Dick printer.

Non-printing control characters direct the cursor on the HP 2600A CRT. (The characters are ignored by a standard teletype.)

These differences mean that the user will need to modify the software to run on a standard HP 2000A system. Although it will RUN without modification on an HP 2000C, its 64 word/record limitation makes inefficient use of the system.

PRICE

36213A, Option K01 (Paper Tape) $110.00
36213A, Option D00 (Documentation) 5.00

ACKNOWLEDGEMENTS:

Jackie Shelton
Computer Terminal Corporation

CTC6

TITLE: CTC ACCOUNTS PAYABLE F709-36638A

DESCRIPTION:

These CTC Accounts Payable programs are part of a total accounting system written by Computer Terminal Corporation, now named Datapoint Corporation. (See also A706-36213 CTC Payroll Program, A717-36212 CTC Projection Programs, A711-36214 CTC Accounts Receivable, A708-36211 CTC Inventory Control for Finished Products, and A708-36210 CTC Manufacturing Parts Control.) Abstracts of the 24 accounts payable programs are on the following page along with an index to the documentation.

INSTRUCTIONS:

Order option DØØ

SYSTEM SPECIFICATIONS:

2000F and Teletype; also Datapoint 2200 Version II (optional)

SPECIAL CONSIDERATIONS:

This package was written for a 2000A with a non-standard disc, but has been modified to run on a 2000F.

Teleprinter output may be directed to a 132 column AB Dick, or equivalent printer.

Non-printing control characters direct the cursor on the HP 2600A CRT. (The characters are ignored by a standard teletype.)

Programs APDTAP and APDSTP make use of a Datapoint 2200 Computer with cassette tape facilities. They are included in this package as an optional feature for any users who have this capability.

These differences mean that the user will need to modify the software to run on a standard HP 2000A system. Although it will RUN without modification on an HP 2000C, or 2000C'/F, its 64 word/record limitation makes inefficient use of the system.

PRICE

36638A, Option K01 (Paper Tape)	$80.00
36638A, Option D00 (Documentation)	5.00

ACKNOWLEDGEMENTS: Jackie Shelton
Datapoint Corporation

INDEX TO ACCOUNTS PAYABLE PROGRAMS

S. APAGPG — Ages the checks held file by vendor placing each vendor on a separate page. The user may specify an interval or group of vendor numbers he wishes to be aged.

T. CTC6 — This program may be used to initialize the files. Just GET and RUN CTC6 to perform the initialization.

TITLE: CTC PROJECTION PROGRAMS

CTC3
A717-36212B

DESCRIPTION:

These CTC Projection Programs are part of a total accounting system written by Computer Terminal Corporation for the HP 2000A. (See A706-36213 CTC Payroll Program, A708-36210 CTC Manufacturing Parts Control, A711-36214 CTC Accounts Receivable, and A708-36211 CTC Inventory Control for Finished Projects.) Abstracts of the 10 projection programs are on the following page along with an index to the documentation.

INSTRUCTIONS:

Order option DØØ.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

This package was written for a 2000A with a non-standard disc. The file structure has 200 physical records per file.

Teleprinter output is directed to a 132 column AB Dick printer.

Non-printing control characters direct the cursor on the HP 2600A CRT. (The characters are ignored by a standard teletype.)

These differences mean that the user will need to modify the software to run on a standard HP 2000A system. Although it will RUN without modification on an HP 2000C, its 64 word/record limitation makes inefficient use of the system.

PRICE

36212B, Option K01 (Paper Tape)	$30.00
36212B, Option D00 (Documentation)	5.00

ACKNOWLEDGEMENTS:

Jackie Shelton
Computer Terminal Corporation

INDEX TO PROJECTION PROGRAMS

TITLE: COLLEGE REGISTRATION DEMO

COLREG
F720-36282A

DESCRIPTION:

COLREG is a comprehensive demo package that illustrates the feasibility of an on-line, multi-terminal college registration. The package consists of 10 programs: CLEAR, FILE, PREREG, REPS, AVAIL, REG, SOFAR, GRAD, CLASS, and COPY.

INSTRUCTIONS:

It is estimated that 1 million characters are needed for each 1,000 students. This estimate is for student and course information only and does not account for program storage.

The operation here assumes that the college registrar will manually plan a list of available courses and associated instructors. This list is entered into the system as well as sent to all students. The students then plan their individual programs at home and then at registration day they enter their programs into the system on one of 32 terminals. This system then checks for validity, class conflict, etc. as students enter their courses. Upon successful completion, the student is given an on-the-spot report of his program. In addition, class reports for the instructors can be obtained at the end of registration.

During the year the individual instructors may enter grades for his students and provide reports for all student grades.

The time per student at registration seems to be about 5 minutes, so a 32 terminal system could handle 384 students/hour.

Continued on following page.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

1. Maximum number of students - 32
2. Maximum number of courses one student can take - 9
3. Maximum number of credits a student may take - 17
4. Maximum number of courses available - 9
5. Maximum number of sections within a course - 6
6. Maximum number of hours per week a section may have - 5
7. Sections must be scheduled at same start time and length on each available day.
8. Two programs that write on files may not run simultaneously.
9. No deletes are provided for students, courses, or sections.
10. May not change the original quota for any section.

PRICE

Not available on paper tape. To order a magnetic tape of all contributed BASIC programs, order HP Part No. 02000-90029 for 2000C, or HP Part No. 02000-90060 for 2000F, Price $25.00.

ACKNOWLEDGEMENTS: George Tibaldi/Dave Denman
HP, Eastern Sales Region

INSTRUCTIONS: continued

NAME OF PROGRAM	FUNCTION
CLEAR	Clears student and course files
FILE	To enter available courses and sections
PREREG	To enter student names and addresses. Prior to registration (if required)
REPS	Lists students' name and addresses
AVAIL	Lists courses and sections available
REG	To register students (can be used to list any students programs)
SOFAR	To list courses, sections and students registered SOFAR
GRAD	To enter midterm or final grades
CLASS	To provide a class report for each instructor
COPY	Restores (copies) two back-up files to student and course

1. OPEN-STUDNT, 128
   OPEN-COURSE, 128
   OPEN-WORK, 3
   OPEN-WORK1, 10
   OPEN-WORK2, 10
2. RUN "CLEAR" to clean student and course file
3. RUN "FILE" to enter an available curriculum
4. RUN "AVAIL" to get a report of all courses and sections entered with "FILE"
5. May RUN "PREREG" to enter names and addresses of students. This saves time at actual registration ("REG") in that names, etc. are already entered.
6. RUN "REG" to enter courses for students. This student should use this program with a copy of the report from "AVAIL" in his hand.
7. May RUN "SOFAR" at any time to see the number of students enrolled SOFAR
8. May RUN "CLASS" at any time to see the actual students enrolled in each class. This program thus provides class reports for all courses to be given to each instructor
9. At midterm and final time, each instructor would run "GRAD" to enter the grades for his students
10. After all midterm or final grades are entered, run "CLASS" again to see entered grades
11. RUN "REPS" at any time to see students in file

The following list of programs may be run at any time with out altering any files since they read only:

REPS, AVAIL, SOFAR, CLASS or REG (with just typing "END" in response to "COURSE-SEC")

For demo purposes, the following procedures are useful:

A. RUN "CLEAR"

B. RUN "FILE" for 7 courses and 3 to 4 sections per course

C. RUN "PREREG" for about 10 students

D. Using a file copy program, ("COPY") save both student and course files. In this way this state can easily be recreated from back-up files (CSAVE and SSAVE).

The file copy program, COPY, included in this program package has been set up to perform these steps. The user may just RUN COPY to restore the state. The sample RUNs which follow illustrate the building of the files rather than using COPY which has already set up the files.

```
RUN

GET-CLEAR
RUN
CLEAR                          CLEARS STUDENT AND COURSE FILE

DONE


GET-REPS
RUN
REPS                           FILES ARE EMPTY

ALL OR STUDENT # ?
?ALL

NOT IN FILES

DONE


GET-PREREG
RUN
PREREG

RESPOND WITH ANSWER OR END

STUDENT #??229-56-5036
LAST NAME??BANISCH
FIRST NAME??JIM
MIDDLE INITIAL??Z
PHONE??265-7000
SEX??M
CITY??C-KING OF PRUSSIA
STATE??PENNSYLVANIA
BAD INPUT, RETYPE FROM ITEM 1          PREREGISTER STUDENTS
??PENN.
STREET ADDRESS??1021 EIGHTH ST.
STUDENT #??149-24:686
LAST NAME??FRANK
FIRST NAME??LEE
MIDDLE INITIAL??Z
PHONE??667-4000
SEX??M
CITY??CHERRY HILL
STATE??NEW JERSEY
BAD INPUT, RETYPE FROM ITEM 1
??N.J.
STREET ADDRESS??1060 N. KINGS HIGHWAY
STUDENT #??136-26-1841
LAST NAME??BOLCIK
FIRST NAME??BOB
MIDDLE INITIAL??Z
PHONE??948-6370
SEX??M
CITY??ROCKVILLE
STATE??MD.
STREET ADDRESS??2 CHOKE CHERRY LANE
STUDENT #??065-32:2709
LAST NAME??TIBALDI
FIRST NAME??GEORGE
MIDDLE INITIAL??A
PHONE??265-5000
SEX??M
CITY??PARAMUS
STATE??N.J.
STREET ADDRESS??120 W. CENTURY RD.
STUDENT #??012-28-2705
LAST NAME??KELLEY
FIRST NAME??KEN
```

```
MIDDLE INITIAL??Z
PHONE??948-6370
SEX??M
CITY??ROCKVILLE
STATE??MD.
STREET ADDRESS??2 CHOKE CHERRY LANE
STUDENT #??END

DONE



GET-REPS
RUN
REPS

ALL OR STUDENT # ?
?ALL
229-56-5036 BANISCH JIM Z
1021 EIGHTH ST.  KING OF PRUSSIA,PENN.
265-7000      SEX M

                                                    LISTS ALL STUDENTS PREREGISTERED

149-24:686 FRANK LEE Z
1060 N. KINGS HIGHWAY  CHERRY HILL,N.J.
667-4000      SEX M



136-26-1841 BOLCIK BOB Z
2 CHOKE CHERRY LANE  ROCKVILLE,MD.
948-6370      SEX M



065-32:2709 TIBALDI GEORGE A
120 W. CENTURY RD.  PARAMUS,N.J.
265-5000      SEX M



012-28-2705 KELLEY KEN Z
2 CHOKE CHERRY LANE  ROCKVILLE,MD.
948-6370      SEX M



END OF FILE

DONE



GET-FILE
RUN
FILE

TITLE ?ENG1                                         ENTER COURSES
CREDITS ?3
SECTIONS ?3



SECTION # ?2
START TIME ?10
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?3
DAY  3     ?5
INSTRUCTOR ?SMITH
QUOTA ?5
```

```
SECTION # ?1
START TIME ?9
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?4
DAY  3     ?5
INSTRUCTOR ?DRAPER
QUOTA ?4



SECTION # ?3
START TIME ?3
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?3
DAY  3     ?5
INSTRUCTOR ?BROWN
QUOTA ?4
DONE??NO
TITLE ?HIST1
CREDITS ?3
SECTIONS ?4



SECTION # ?1
START TIME ?9
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?3
DAY  3     ?5
INSTRUCTOR ?WATTS
QUOTA ?4



SECTION # ?2
START TIME ?11
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?3
DAY  3     ?5
INSTRUCTOR ?WATTS
QUOTA ?4



SECTION # ?3
START TIME ?1
LENGTH ?1
TIMES/WEEK ?3
DAY  1     ?1
DAY  2     ?4
DAY  3     ?5
INSTRUCTOR ?DRAPER
QUOTA ?3
```

```
SECTION # ?4
START TIME ?4
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?3
DAY  3    ?5
INSTRUCTOR ?MARK
QUOTA ?3
DONE??NO
TITLE ?EC01
CREDITS ?3
SECTIONS ?4



SECTION # ?1
START TIME ?10
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?3
DAY  3    ?5
INSTRUCTOR ?I-HARRIS
QUOTA ?3



SECTION # ?2
START TIME ?11
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?2
DAY  3    ?5
INSTRUCTOR ?MARTIN
QUOTA ?4



SECTION # ?3
START TIME ?3
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?3
DAY  3    ?5
INSTRUCTOR ?HARRIS
QUOTA ?3



SECTION # ?4
START TIME ?4
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?2
DAY  2    ?3
DAY  3    ?5
INSTRUCTOR ?BONNER
QUOTA ?4
DONE??NO
TITLE ?BI01
CREDITS ?4
SECTIONS ?4
```

```
SECTION # ?1
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?2
INSTRUCTOR ?MC CLEAN
QUOTA ?4



SECTION # ?2
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?4
INSTRUCTOR ?MARIN
QUOTA ?3



SECTION # ?3
START TIME ?1
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?4
INSTRUCTOR ?KNOPP
QUOTA ?4



SECTION # ?4
START TIME ?1
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?5
INSTRUCTOR ?MC CLEAN
QUOTA ?4
DONE??NO
TITLE ?CHEM1
CREDITS ?4
SECTIONS ?3



SECTION # ?1
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?2
INSTRUCTOR ?TOTIE
QUOTA ?4



SECTION # ?2
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1    ?4
INSTRUCTOR ?MATHEWS
QUOTA ?3
```

```
SECTION # ?3
START TIME ?1
LENGTH ?4
TIMES/WEEK ?1
DAY  1     ?3
INSTRUCTOR ?TOTIE
QUOTA ?4
DONE??NO
TITLE ?CHEM1
DUPLICATE COURSE NAME
SECTION ADDITION (Y OR N) ?N
TITLE ?PHY1
CREDITS ?4
SECTIONS ?4



SECTION # ?1
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1     ?1
INSTRUCTOR ?KING
QUOTA ?4



SECTION # ?2
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1     ?2
INSTRUCTOR ?FULLER
QUOTA ?3



SECTION # ?3
START TIME ?9
LENGTH ?4
TIMES/WEEK ?1
DAY  1     ?4
INSTRUCTOR ?KING
QUOTA ?4



SECTION # ?4
START TIME ?1
LENGTH ?4
TIMES/WEEK ?1
DAY  1     ?4
INSTRUCTOR ?FOSTER
QUOTA ?4
DONE??NO
TITLE ?PHYED1
CREDITS ?3
SECTIONS ?4



SECTION # ?1
START TIME ?9
LENGTH ?2
TIMES/WEEK ?2
DAY  1     ?2
DAY  2     ?4
INSTRUCTOR ?PARRISH
QUOTA ?5
```

```
SECTION # ?2
START TIME ?9
LENGTH ?2
TIMES/WEEK ?2
DAY  1    ?1
DAY  2    ?3
INSTRUCTOR ?PEREIRA
QUOTA ?5



SECTION # ?3
START TIME ?2
LENGTH ?2
TIMES/WEEK ?2
DAY  1    ?4
DAY  2    ?5
INSTRUCTOR ?MEYER
QUOTA ?5
```


```
SECTION # ?4
START TIME ?1
LENGTH ?2
TIMES/WEEK ?2
DAY  1    ?1
DAY  2    ?2
INSTRUCTOR ?MACKENZIE
QUOTA ?5
DONE??YES

DONE



GET-AVAIL
RUN
AVAIL                                PRINTS OUT LIST OF ENTERED COURSES

COURSE       SEC  CRDS  INSTRUCTOR          TIME

ENG1         2     3    SMITH              10-11MOWDFR
ENG1         1     3    DRAPER             9-10MOTHFR
ENG1         3     3    BROWN              3-4MOWDFR

HIST1        1     3    WATTS              9-10MOWDFR
HIST1        2     3    WATTS              11-12MOWDFR
HIST1        3     3    DRAPER             1-2MOTHFR
HIST1        4     3    MARK               4-5MOWDFR

EC01         1     3    HARRIS             10-11MOWDFR
EC01         2     3    MARTIN             11-12MOTUFR
EC01         3     3    HARRIS             3-4MOWDFR
EC01         4     3    BONNER             4-5TUWDFR

BI01         1     4    MC CLEAN           9-1TU
BI01         2     4    MARIN              9-1TH
BI01         3     4    KNOPP              1-5TH
BI01         4     4    MC CLEAN           1-5FR

CHEM1        1     4    TOTIE              9-1TU
CHEM1        2     4    MATHEWS            9-1TH
CHEM1        3     4    TOTIE              1-5WD

PHY1         1     4    KING               9-1MO
PHY1         2     4    FULLER             9-1TU
PHY1         3     4    KING               9-1TH
PHY1         4     4    FOSTER             1-5TH
```

```
PHYED1        1      3     PARRISH                  9-11TUTH
PHYED1        2      3     PEREIRA                  9-11MOWD
PHYED1        3      3     MEYER                    2-4THFR
PHYED1        4      3     MACKENZIE                1-3MOTU


DONE




GET-REG
RUN
REG

RESPOND WITH ANSWER OR END


STUDENT # ??229-56-5036

                                    REGISTRATION OF A STUDENT WHO HAD PREREGISTERED
COURSE-SECTION
?HIST1-3
CREDITS ENTERED SOFAR=  3

COURSE-SECTION
?ENG1-1
CREDITS ENTERED SOFAR=  6

COURSE-SECTION
?HIST-1
COURSE OR SECTION NON EXISTENT
                                          STUDENT PICKS COURSES FROM "AVAIL" PRINTOUT
COURSE-SECTION
?HIST1-1
ALREADY HAVE HIST1

COURSE-SECTION
?ECO1-1
COURSE OR SECTION NON EXISTENT

COURSE-SECTION
?EC01-1
CREDITS ENTERED SOFAR=  9

COURSE-SECTION
?CHEM1-3
CREDITS ENTERED SOFAR=  13

COURSE-SECTION
?PHYED1-2
TIME CONFLICT WITH ENG1

COURSE-SECTION
?PHYED1-1
TIME CONFLICT WITH ENG1

COURSE-SECTION
?PHYED1-4
TIME CONFLICT WITH HIST1

COURSE-SECTION
?PHYED1-3
CREDITS ENTERED SOFAR=  16

COURSE-SECTION
?END
```

```
229-56-5036  BANISCH  JIM Z                    STUDENT'S PROGRAM OF CLASSES

COURSE        SEC   CRDS   INSTRUCTOR              TIME

HIST1         3      3     DRAPER                  1-2MOTHFR
ENG1          1      3     DRAPER                  9-10MOTHFR
EC01          1      3     HARRIS                  10-11MOWDFR
CHEM1         3      4     TOTIE                   1-5WD
PHYED1        3      3     MEYER                   2-4THFR

TOTAL CREDITS=  16


STUDENT # ??END

DONE



GET-FILE
RUN
FILE

TITLE ?PHYED1
DUPLICATE COURSE NAME
SECTION ADDITION (Y OR N) ?Y
SECTION # ?8
START TIME ?1                          ADDS A NEW SECTION OF PHYS.ED. BECAUSE OF NEED
LENGTH ?2                              EVIDENT ON REGISTRATION DAY
TIMES/WEEK ?2
DAY  1    ?3
DAY  2    ?4
INSTRUCTOR ?PIXLER
QUOTA ?2
DONE??YES

DONE



GET-FILE
RUN
FILE

TITLE ?MATH1
CREDITS ?3
SECTIONS ?2

                                       ENTERS A NEW COURSE WITH 2 SECTIONS


SECTION # ?1
START TIME ?9
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?3
DAY  3    ?4
INSTRUCTOR ?GOODMAN
QUOTA ?5



SECTION # ?2
START TIME ?11
LENGTH ?1
TIMES/WEEK ?3
DAY  1    ?1
DAY  2    ?3
DAY  3    ?5
INSTRUCTOR ?BLAIR
QUOTA ?5
DONE??YES

DONE
```

```
GET-SOFAR
RUN
SOFAR
```

DURING REGISTRATION PROCESS WE CAN SEE THE ACTIVITY SO FAR

```
COURSE        SEC   CRDS   INSTRUCTOR              QUOTA SOFAR

ENG1          2       3    SMITH                     5     Ø
ENG1          1       3    DRAPER                    4     1
ENG1          3       3    BROWN                     4     Ø

HIST1         1       3    WATTS                     4     Ø
HIST1         2       3    WATTS                     4     Ø
HIST1         3       3    DRAPER                    3     1
HIST1         4       3    MARK                      3     Ø

EC Ø1         1       3    HARRIS                    3     1
EC Ø1         2       3    MARTIN                    4     Ø
EC Ø1         3       3    HARRIS                    3     Ø
EC Ø1         4       3    BONNER                    4     Ø

BI Ø1         1       4    MC CLEAN                  4     Ø
BI Ø1         2       4    MARIN                     3     Ø
BI Ø1         3       4    KNOPP                     4     Ø
BI Ø1         4       4    MC CLEAN                  4     Ø

CHEM1         1       4    TOTIE                     4     Ø
CHEM1         2       4    MATHEWS                   3     Ø
CHEM1         3       4    TOTIE                     4     1

PHY1          1       4    KING                      4     Ø
PHY1          2       4    FULLER                    3     Ø
PHY1          3       4    KING                      4     Ø
PHY1          4       4    FOSTER                    4     Ø

PHYED1        1       3    PARRISH                   5     Ø
PHYED1        2       3    PEREIRA                   5     Ø
PHYED1        3       3    MEYER                     5     1
PHYED1        4       3    MACKENZIE                 5     Ø
PHYED1        8       3    PIXLER                    2     Ø

MATH1         1       3    GOODMAN                   5     Ø
MATH1         2       3    BLAIR                     5     Ø

TOTAL REGISTERED SOFAR = 5
```

NOTE NEW COURSES WE ADDED

NUMBER OF SEATS, NOT STUDENTS

```
DONE
```

```
GET-REG
RUN
REG

RESPOND WITH ANSWER OR END


STUDENT # ??111-22-3333
LAST NAME??STAUBER
FIRST NAME??LEE
MIDDLE INITIAL??Z
PHONE??345-6980
SEX??M
CITY??CA.
STATE??CA.
STREET ADDRESS??123 FORBES ST.

COURSE-SECTION
?MATH1-2
CREDITS ENTERED SOFAR=  3

COURSE-SECTION
?PHYED1-8
CREDITS ENTERED SOFAR=  6
```

REGISTERS A STUDENT WHOSE NAME WAS NOT PREREGISTERED

```
COURSE-SECTION
?HIST1-3
TIME CONFLICT WITH PHYED1

COURSE-SECTION
?ENG1-1
CREDITS ENTERED SOFAR=  9

COURSE-SECTION
?END


111-22-3333  STAUBER  LEE Z

COURSE        SEC  CRDS  INSTRUCTOR              TIME

MATH1         2     3    BLAIR                  11-12MOWDFR
PHYED1        8     3    PIXLER                 1-3WDTH
ENG1          1     3    DRAPER                 9-10MOTHFR

TOTAL CREDITS=  9


STUDENT # ??END

DONE
```

```
GET-CLASS                        AFTER REGISTRATION, 'CLASS' PROVIDES TEACHER WITH
RUN                              ENROLLMENT IN HIS OWN CLASS
CLASS

ALL,COURSE, OR COURSE-SECTION ?
?ENG1-1


ENG1-1  CR= 3      9-10MOTHFR  DRAPER  COUNT= 2

STUDENT #       NAME                                          MT    F

229-56-5036     BANISCH  JIM  Z
111-22-3333     STAUBER  LEE  Z


DONE
```

```
GET-GRADE
RUN
GRADE

ENTER COURSE TITLE?ENG1
ENTER SECTION#?1
MID-TERM OR FINAL (M OR F)?M        THE TEACHER MAY ENTER GRADES FOR HIS CLASS
ENTER MARK FOR JIM BANISCH?F
ENTER MARK FOR LEE STAUBER?A

DONE
```

```
GET-CA-LASS
RUN
CLASS

ALL,COURSE, OR COURSE-SECTION ?
?ENG1-1


                              RUN 'CLASS' AGAIN TO SHOW ENROLLMENT AND GRADES



ENG1-1  CR= 3      9-10MOTHFR  DRAPER  COUNT= 2

STUDENT #       NAME                                          MT   F

229-56-5036     BANISCH  JIM  Z                               F
111-22-3333     STAUBER  LEE  Z                               A



DONE


LISTING


CLEAR

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         CLEAR:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 1 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  REM  CLEAR STUDENT AND COURSE FILES
20  FILES STUDNT,COURSE
30  K=1
40  FOR I=1 TO 32
50  PRINT #1,K;"  ","  ", END
60  PRINT #1,K+2;0, END
70  K=K+4
80  NEXT I
100  K=1
110  FOR I=1 TO 9
120  PRINT #2,K;"  ", END
130  J=K+1
140  FOR M=1 TO 6
150  PRINT #2,J;"  ", END
160  J=J+2
170  NEXT M
175  K=K+13
180  NEXT I
190  END


FILE

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         FILE:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 2 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  FILES STUDNT,COURSE
20  DIM A$[10],B$[20],D$[20],F$[10],G$[20],H$[11],Q$[5],S$[15]
30  DIM J$[2]
40  DIM K$[20]
50  DIM E$[2]
```

```
60  DIM D[5],E[5]
70  C2=0
80  S$="123456789101112"
90  F$="MOTUWDTHFR"
100  L=0
110  FOR N=1 TO 5
120  E[N]=N+L
130  L=L+1
140  NEXT N
150  H$="ABCDEFGHIJK"
160  PRINT "TITLE ";
170  INPUT A$
180  Z$=A$
190  GOSUB 9200
200  IF Z=0 THEN 260
210  PRINT "DUPLICATE COURSE NAME"
220  PRINT "SECTION ADDITION (Y OR N) ";
230  INPUT I$
240  IF I$="Y" THEN 1090
250  GOTO 160
260  Z$="  "
270  GOSUB 9200
280  PRINT "CREDITS ";
290  INPUT C1
300  PRINT "SECTIONS ";
310  INPUT S1
320  IF S1 <= 6 THEN 350
330  PRINT "ONLY 6 SECTIONS ALLOWED"
340  GOTO 300
350  READ #2,Z
360  PRINT #2;A$,C1,S1, END
370  Z=Z+1
380  FOR S=1 TO S1
390  FOR I=1 TO 5
400  PRINT
410  NEXT I
420  PRINT "SECTION # ";
430  INPUT E$
440  PRINT "START TIME ";
450  INPUT S2
460  PRINT "LENGTH ";
470  INPUT L
480  PRINT "TIMES/WEEK ";
490  INPUT L1
500  FOR I=1 TO L1
510  PRINT "DAY ";I;
520  INPUT D[I]
530  NEXT I
540  PRINT "INSTRUCTOR ";
550  INPUT D$
560  PRINT "QUOTA ";
570  INPUT Q1
580  IF Q1 <= 8 THEN 610
590  PRINT "ONLY 8 STUDENTS ALLOWED"
600  GOTO 560
610  S9=S2
620  IF S2<10 THEN 700
630  IF S2=10 THEN 680
640  IF S2=12 THEN 670
650  S2=12
660  GOTO 680
670  S2=14
680  S3=S2+1
690  GOTO 710
700  S3=S2
710  G$=S$[S2,S3]
720  G$[LEN(G$)+1]="-"
730  S4=S9+L
740  IF S4 <= 12 THEN 760
750  S4=S4-12
760  IF S4<10 THEN 840
770  IF S4=10 THEN 820
780  IF S4=12 THEN 810
790  S4=12
800  GOTO 820
810  S4=14
820  S3=S4+1
830  GOTO 850
```

```
840  S3=S4
850  G$[LEN(G$)+1]=S$[S4,S3]
860  FOR I=1 TO L1
870  G$[LEN(G$)+1]=F$[E[D[I]],E[D[I]]+1]
880  NEXT I
890  FOR I=1 TO L1
900  S5=S9
910  FOR J=1 TO L
920  K$[LEN(K$)+1]=H$[S5,S5]
930  K$[LEN(K$)+1]=H$[D[I],D[I]]
940  S5=S9+J
950  NEXT J
960  NEXT I
970  PRINT #2,Z
980  PRINT #2;E$,G$,K$,D$,Q1,C2, END
990  Z=Z+2
1000  K$=M$
1010  IF I$="Y" THEN 1030
1020  NEXT S
1030  PRINT "DONE?";
1040  INPUT Q$
1050  IF Q$#"NO" THEN 1070
1060  GOTO 160
1070  STOP
1080  END
1090  READ #2;A$,C1,S1
1100  IF S1<6 THEN 1130
1110  PRINT "NO MORE SECTIONS CAN BE ADDED"
1120  STOP
1130  S2=S1+1
1140  PRINT #2,Z;A$,C1,S2, END
1150  PRINT "SECTION # ";
1160  INPUT E$
1170  Z1=Z
1180  Z=Z+1
1190  FOR I=1 TO S1
1200  READ #2,Z;J$
1210  IF J$#E$ THEN 1250
1220  PRINT "DUPLICATE SECTION # "
1230  Z=Z1
1240  GOTO 1150
1250  Z=Z+2
1260  NEXT I
1270  GOTO 440
9000  REM GET STUDENT OR COURSE FILE.
9030  DIM W$[20],X$[12],Y$[20],Z$[10]
9031  IF  END #1 THEN 9038
9032  X1=1
9033  FOR X2=1 TO 32
9034  READ #1,X1;W$
9035  IF W$=X$ THEN 9040
9036  X1=X1+4
9037  NEXT X2
9038  X=0
9039  RETURN
9040  READ #1,X1
9045  X=X1
9046  RETURN
9200  IF  END #2 THEN 9207
9201  X1=1
9202  FOR X2=1 TO 9
9203  READ #2,X1;W$
9204  IF W$=Z$ THEN 9209
9205  X1=X1+13
9206  NEXT X2
9207  Z=0
9208  RETURN
9209  READ #2,X1
9210  Z=X1
9211  RETURN
9212  END
```

PREREG

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         PREREG:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 3 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
8  REM  FILES STUDNT
10  REM  PRE REGESTRATION - ENTER STUDENT NAMES
20  DIM A$[12],B$[20],C$[20],D$[1],E$[12],F$[1]
30  DIM G$[30],H$[8],I$[30]
35  DIM Q$[3]
40  PRINT "RESPOND WITH ANSWER OR END"
50  PRINT
60  PRINT "STUDENT #?";
70  INPUT A$
75  IF A$="END" THEN 700
80  X$=A$
90  GOSUB 9000
100  IF X#0 THEN 500
110  X$="  "
120  GOSUB 9000
130  IF X=0 THEN 600
140  PRINT "LAST NAME?";
150  INPUT B$
160  PRINT "FIRST NAME?";
170  INPUT C$
180  PRINT "MIDDLE INITIAL?";
190  INPUT D$
200  PRINT "PHONE?";
210  INPUT E$
220  PRINT "SEX?";
230  INPUT F$
240  PRINT "CITY?";
250  INPUT G$
260  PRINT "STATE?";
265  INPUT H$
270  PRINT "STREET ADDRESS?";
280  INPUT I$
300  PRINT #1;A$,B$,C$,D$,E$,F$,G$,H$,I$, END
310  GOTO 60
500  PRINT "STUDENT IN FILES  DO YOU WISH TO CHANGE DATA?"
510  INPUT Q$
520  IF Q$="YES" THEN 140
530  IF Q$="NO" THEN 60
540  GOTO 500
600  PRINT "NO MORE STUDENTS ALLOWED"
700  STOP
9000  REM GET STUDENT OR COURSE FILE.
9030  DIM W$[20],X$[12],Y$[20],Z$[10]
9031  IF  END #1 THEN 9038
9032  X1=1
9033  FOR X2=1 TO 32
9034  READ #1,X1;W$
9035  IF W$=X$ THEN 9040
9036  X1=X1+4
9037  NEXT X2
9038  X=0
9039  RETURN
9040  READ #1,X1
9045  X=X1
9046  RETURN
9200  IF  END #2 THEN 9207
9201  X1=1
9202  FOR X2=1 TO 9
9203  READ #2,X1;W$
9204  IF W$=Z$ THEN 9209
9205  X1=X1+13
9206  NEXT X2
9207  Z=0
9208  RETURN
9209  READ #2,X1
9210  Z=X1
9211  RETURN
9212  END
```

REPS

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         REPS:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 4 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  FILES STUDNT
15  DIM V$[12]
20  DIM A$[12],B$[20],C$[20],E$[12],G$[30],H$[8],I$[30]
22  F=0
24  PRINT "ALL OR STUDENT # ?"
26  INPUT V$
30  IF  END #1 THEN 140
35  Z=1
40  READ #1,Z;A$
41  IF A$="  " THEN 110
42  IF V$="ALL" THEN 44
43  IF V$#A$ THEN 110
44  F=1
45  READ #1;B$,C$,D$,E$,F$,G$,H$,I$
50  PRINT A$;" ";B$;" ";C$;" ";D$
60  PRINT I$;"  ";G$;",";H$
70  PRINT E$;"      SEX ";F$
80  FOR I=1 TO 3
90  PRINT
100  NEXT I
110  Z=Z+4
130  GOTO 40
140  FOR I=1 TO 5
150  PRINT
160  NEXT I
162  IF F=1 THEN 170
164  PRINT "NOT IN FILES"
166  STOP
170  PRINT "END OF FILE"
180  END
```

AVAIL

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         AVAIL:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 5 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  REM COURSES AND SECTIONS AVAILABLE REPORT
20  FILES COURSE
30  DIM N$[10],O$[20],P$[20],Q$[20],R$[20]
40  IF  END #1 THEN 280
60  PRINT
70  PRINT "COURSE";TAB(13);"SEC";TAB(18);"CRDS";TAB(24);
75  PRINT "INSTRUCTOR";TAB(47);"TIME"
80  PRINT
90  K1=1
100  FOR I=1 TO 9
110  READ #1,K1;N$
120  IF N$="  " THEN 240
130  READ #1;N3
140  M1=K1+1
150  FOR J=1 TO 6
160  READ #1,M1;O$
170  IF O$="  " THEN 210
180  READ #1;P$,Q$,R$,N5,N6
190  PRINT N$;TAB(13);O$;TAB(18);N3;TAB(24);R$;TAB(47);P$
210  M1=M1+2
220  NEXT J
230  PRINT
240  K1=K1+13
250  NEXT I
260  PRINT
270  PRINT
280  END
```

```
REG

1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         REG:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 6 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
8  M9=17
9  FILES STUDNT,COURSE,WORK,STUDNT,COURSE
10  REM ACTUAL REGESTRATION
20  DIM A$[12],B$[20],C$[20],D$[1],E$[12],F$[1]
30  DIM G$[30],H$[8],I$[30]
35  DIM J$[10],K$[2],L$[2],M$[2],N$[10],O$[2]
36  DIM P$[20],Q$[20],R$[20],S$[12],T$[2],U$[2]
40  PRINT "RESPOND WITH ANSWER OR END"
50  PRINT
60  PRINT
61  PRINT
62  PRINT "STUDENT # ?";
70  INPUT A$
75  IF A$="END" THEN 8510
80  X$=A$
90  GOSUB 9000
100  IF X#0 THEN 500
110  X$="  "
120  GOSUB 9000
130  IF X=0 THEN 8600
140  PRINT "LAST NAME?";
150  INPUT B$
160  PRINT "FIRST NAME?";
170  INPUT C$
180  PRINT "MIDDLE INITIAL?";
190  INPUT D$
200  PRINT "PHONE?";
210  INPUT E$
220  PRINT "SEX?";
230  INPUT F$
240  PRINT "CITY?";
250  INPUT G$
260  PRINT "STATE?";
265  INPUT H$
270  PRINT "STREET ADDRESS?";
280  INPUT I$
300  PRINT #1;A$,B$,C$,D$,E$,F$,G$,H$,I$, END
500  REM NAME,ADD ,ETC ARE NOW STORED AWAY
502  REM A$=STUD #  &  X=SECTOR
510  READ #1,X+2
520  READ #1;N1
530  PRINT #3,1; END
540  C1=0
550  IF N1=0 THEN 800
560  FOR I=1 TO N1
570  READ #1;Z$,K$,L$,M$,N2
580  GOSUB 9200
590  IF Z=0 THEN 8500
600  READ #2;Z$,N3,N4
610  IF N2#N3 THEN 8500
620  IF N4=0 THEN 8500
630  J=Z+1
640  FOR K=1 TO N4
650  READ #2,J;O$,P$,Q$
660  IF K$=O$ THEN 700
670  J=J+2
680  NEXT K
690  GOTO 8500
700  C1=C1+N2
710  PRINT #3;Z$,Q$, END
720  NEXT I
800  PRINT
801  PRINT "COURSE-SECTION"
802  INPUT G$
805  IF G$="END" THEN 7000
820  FOR I=1 TO 13
830  IF G$[I,I]="-" THEN 900
840  NEXT I
850  PRINT "ERROR"
```

```
860  GOTO 800
900  J$=G$[1,I-1]
910  K$=G$[I+1]
920  Z$=J$
930  GOSUB 9200
940  REM Z$=COURSE  K$=SECTION  Z=SECTOR
950  IF Z=0 THEN 8000
960  IF  END #3 THEN 1010
970  READ #3,1
980  READ #3;N$,Q$
990  IF N$=Z$ THEN 8100
1000  GOTO 980
1010  READ #2;N$,N3,N4
1020  IF C1+N3>M9 THEN 8200
1030  K1=Z+1
1040  IF N4=0 THEN 8200
1050  FOR I=1 TO N4
1060  READ #2,K1
1070  READ #2;O$,P$,Q$,R$,N5,N6
1080  IF O$=K$ THEN 1130
1090  K1=K1+2
1100  NEXT I
1110  GOTO 8000
1120  REM
1130  IF N5=N6 THEN 8300
1140  IF  END #3 THEN 1250
1150  READ #3,1
1160  READ #3;N$,B$
1170  IF LEN(B$)=0 THEN 8500
1180  FOR I=1 TO LEN(B$) STEP 2
1190  FOR J=1 TO LEN(Q$) STEP 2
1200  IF Q$[J,J+1]=B$[I,I+1] THEN 8400
1210  NEXT J
1220  NEXT I
1230  GOTO 1160
1240  REM ALL OK NOW TO REG IN COURSE
1250  PRINT #3;Z$,Q$, END
1260  N6=N6+1
1270  C1=C1+N3
1280  READ #5,K1
1290  PRINT #5;O$,P$,Q$,R$,N5,N6
1300  IF N6-1=0 THEN 1350
1310  FOR I=1 TO N6-1
1320  READ #2;S$,T$,U$
1330  PRINT #5;S$,T$,U$
1340  NEXT I
1350  PRINT #5;A$,"  ","  ", END
1351  READ #5,1
1352  READ #2,1
1400  IF  END #1 THEN 1500
1410  READ #1,X+2;N1
1420  N1=N1+1
1430  PRINT #4,X+2;N1
1440  IF N1-1=0 THEN 1500
1450  FOR I=1 TO N1-1
1460  READ #1;J$,K$,L$,M$,N2
1470  PRINT #4;J$,K$,L$,M$,N2
1480  NEXT I
1500  PRINT #4;Z$,O$,"  ","  ",N3, END
1501  READ #4,1
1502  READ #1,1
1600  PRINT "CREDITS ENTERED SOFAR= ";C1
1610  GOTO 800
7000  READ #1,X;A$,B$,C$,D$
7010  PRINT
7011  PRINT
7012  PRINT
7020  PRINT A$;"  ";B$;"  ";C$;" ";D$
7030  PRINT
7040  READ #1,X+2;N1
7050  IF N1=0 THEN 7170
7052  PRINT "COURSE";TAB(13);"SEC";TAB(18);"CRDS";TAB(24);
7053  PRINT "INSTRUCTOR";TAB(47);"TIME"
7054  PRINT
7060  C1=0
7070  FOR I=1 TO N1
7080  READ #1;Z$,K$,L$,M$,N2
7090  GOSUB 9200
```

```
7100  IF Z=0 THEN 8500
7110  READ #2;N$,N3,N4
7112  K4=Z+1
7114  FOR J=1 TO N4
7115  READ #2,K4;O$,P$,Q$,R$
7116  IF O$=K$ THEN 7130
7117  K4=K4+2
7118  NEXT J
7130  PRINT N$;TAB(13);O$;TAB(18);N3;TAB(24);R$;TAB(47);P$
7140  C1=C1+N3
7150  NEXT I
7160  PRINT
7170  PRINT "TOTAL CREDITS= ";C1
7171  PRINT
7172  PRINT
7180  GOTO 60
8000  PRINT "COURSE OR SECTION NON EXISTENT"
8010  GOTO 800
8100  PRINT "ALREADY HAVE ";Z$
8110  GOTO 800
8200  IF C1=M9 THEN 8230
8210  PRINT "TOO MANY CREDITS"
8220  GOTO 800
8230  PRINT "AT MAX CREDITS NOW"
8240  GOTO 7000
8300  PRINT "SECTION FULL"
8310  REM COULD PRINT AVAILABLE
8320  GOTO 800
8400  PRINT "TIME CONFLICT WITH ";N$
8410  GOTO 800
8500  PRINT "TROUBLE"
8510  STOP
8600  PRINT "NO ROOM FOR STUDENT"
8610  STOP
9000  REM GET STUDENT OR COURSE FILE.
9030  DIM W$[20],X$[12],Y$[20],Z$[10]
9031  IF  END #1 THEN 9038
9032  X1=1
9033  FOR X2=1 TO 32
9034  READ #1,X1;W$
9035  IF W$=X$ THEN 9040
9036  X1=X1+4
9037  NEXT X2
9038  X=0
9039  RETURN
9040  READ #1,X1
9045  X=X1
9046  RETURN
9200  IF  END #2 THEN 9207
9201  X1=1
9202  FOR X2=1 TO 9
9203  READ #2,X1;W$
9204  IF W$=Z$ THEN 9209
9205  X1=X1+13
9206  NEXT X2
9207  Z=0
9208  RETURN
9209  READ #2,X1
9210  Z=X1
9211  RETURN
9212  END
```

SOFAR

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  **********************************
2  REM
3  REM         SOFAR:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 7 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  REM STUDENTS REGISTERED SO FAR REPORT
20  FILES COURSE
30  DIM N$[10],O$[20],P$[20],Q$[20],R$[20]
40  IF  END #1 THEN 280
50  E1=0
```

```
60  PRINT
70  PRINT "COURSE";TAB(13);"SEC";TAB(18);"CRDS";TAB(24);
75  PRINT "INSTRUCTOR";TAB(47);"QUOTA";TAB(53);"SOFAR"
80  PRINT
90  K1=1
100  FOR I=1 TO 9
110  READ #1,K1;N$
120  IF N$="  " THEN 240
130  READ #1;N3
140  M1=K1+1
150  FOR J=1 TO 6
160  READ #1,M1;O$
170  IF O$="  " THEN 210
180  READ #1;P$,Q$,R$,N5,N6
190  PRINT N$;TAB(13);O$;TAB(18);N3;TAB(24);R$;TAB(47);N5;TAB(53);N6
200  E1=E1+N6
210  M1=M1+2
220  NEXT J
230  PRINT
240  K1=K1+13
250  NEXT I
260  PRINT "TOTAL REGISTERED SOFAR =";E1
270  PRINT
280  END
```

GRAD

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         GRAD:  COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 8 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  FILES STUDNT,COURSE,WORK1,WORK2
20  DIM O$[2],P$[20],Q$[20],R$[20]
30  DIM A$[12],B$[20],C$[20],D$[2],J$[10],K$[2],M$[2]
35  DIM E$[2],T$[2],U$[2]
36  DIM L$[2]
40  DIM N$[10]
45  DIM F$[12],H$[20]
50  PRINT "ENTER COURSE TITLE";
60  INPUT Z$
70  PRINT "ENTER SECTION#";
80  INPUT E$
90  PRINT "MID-TERM OR FINAL (M OR F)";
100  INPUT V$
110  GOSUB 9200
120  IF Z#0 THEN 150
130  PRINT "COURSE NON EXISTANT"
140  STOP
150  READ #2;N$,N3,N4
160  Z=Z+1
165  IF N4=0 THEN 220
170  FOR I=1 TO N4
180  READ #2,Z;O$
190  IF O$=E$ THEN 240
200  Z=Z+2
210  NEXT I
220  PRINT "SECTION NON EXISTANT"
230  STOP
240  READ #2;P$,Q$,R$,N5,N6
245  IF N6=0 THEN 280
246  FOR L=1 TO N6
250  READ #2;X$,T$,U$
260  GOSUB 9000
270  IF X#0 THEN 300
280  PRINT "STUDENT NOT IN FILE"
290  STOP
300  READ #1;A$,B$,C$
310  PRINT "ENTER MARK FOR ";C$;" ";B$;
320  INPUT D$
330  X=X+2
340  READ #1,X;N1
345  IF N1=0 THEN 400
350  FOR I=1 TO N1
```

```
360  READ #1;J$,K$,L$,M$,N2
370  IF J$=Z$ THEN 420
380  PRINT #4;J$,K$,L$,M$,N2, END
390  NEXT I
400  PRINT "COURSE NOT IN STUDENT FILE"
410  STOP
420  I9=I
425  IF V$="F" THEN 460
430  L$=D$
440  T$=D$
450  GOTO 480
460  M$=D$
470  U$=D$
480  PRINT #4;J$,K$,L$,M$,N2, END
490  PRINT #3;X$,T$,U$, END
500  FOR I=1 TO N1-I9
510  READ #1;J$,K$,L$,M$,N2
520  PRINT #4;J$,K$,L$,M$,N2, END
530  NEXT I
540  READ #4,1
550  READ #1,X
560  PRINT #1;N1, END
570  FOR I=1 TO N1
580  READ #4;J$,K$,L$,M$,N2
590  PRINT #1;J$,K$,L$,M$,N2, END
600  NEXT I
610  NEXT L
620  READ #3,1
630  READ #2,Z
635  PRINT #2;O$,P$,Q$,R$,N5,N6, END
640  FOR I=1 TO N6
650  READ #3;X$,T$,U$
660  PRINT #2;X$,T$,U$, END
670  NEXT I
680  STOP
690  END
9000  REM GET STUDENT OR COURSE FILE.
9030  DIM W$[20],X$[12],Y$[20],Z$[10]
9031  IF  END #1 THEN 9038
9032  X1=1
9033  FOR X2=1 TO 32
9034  READ #1,X1;W$
9035  IF W$=X$ THEN 9040
9036  X1=X1+4
9037  NEXT X2
9038  X=0
9039  RETURN
9040  READ #1,X1
9045  X=X1
9046  RETURN
9200  IF  END #2 THEN 9207
9201  X1=1
9202  FOR X2=1 TO 9
9203  READ #2,X1;W$
9204  IF W$=Z$ THEN 9209
9205  X1=X1+13
9206  NEXT X2
9207  Z=0
9208  RETURN
9209  READ #2,X1
9210  Z=X1
9211  RETURN
9212  END
```

CLASS

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         CLASS:   COLLEGE REGISTRATION DEMO
4  REM
5  REM         36282 REV  A  PART 9 OF 10   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ***************************************
10  REM CLASS REPORTS
20  FILES STUDNT,COURSE
30  DIM N$[10],O$[20],P$[20],Q$[20],R$[20]
```

```
40  DIM A$[12],B$[20],C$[20],D$[1]
50  DIM S$[12],T$[2],U$[2],G$[30],J$[10],K$[2]
60  F=0
70  PRINT "ALL,COURSE, OR COURSE-SECTION ?"
80  INPUT G$
90  IF G$="ALL" THEN 180
100  FOR I=1 TO 13
110  IF G$[I,I]="-" THEN 160
120  NEXT I
130  J$=G$
140  K$="-"
150  GOTO 180
160  J$=G$[1,I-1]
170  K$=G$[I+1]
180  IF  END #1 THEN 670
190  PRINT
200  PRINT
210  K1=1
220  FOR I=1 TO 9
230  READ #2,K1;N$
240  IF N$="  " THEN 590
250  IF G$="ALL" THEN 270
260  IF J$#N$ THEN 590
270  READ #2;N3
280  M1=K1+1
290  FOR J=1 TO 6
300  READ #2,M1;O$
310  IF O$="  " THEN 570
320  IF G$="ALL" THEN 350
330  IF K$="-" THEN 350
340  IF K$#O$ THEN 570
350  F=1
360  READ #2;P$,Q$,R$,N5,N6
370  PRINT
380  PRINT
390  PRINT
400  PRINT
410  PRINT
420  PRINT
430  PRINT
440  PRINT N$;"-";O$;"  CR=";N3;"  ";P$;"  ";R$;"  COUNT=";N6
450  IF N6=0 THEN 570
460  PRINT
470  PRINT "STUDENT #";TAB(15);"NAME";TAB(60);"MT";TAB(65);"F"
480  PRINT
490  FOR L=1 TO N6
500  READ #2;S$,T$,U$
510  X$=S$
520  GOSUB 9000
530  IF X=0 THEN 680
540  READ #1;A$,B$,C$,D$
550  PRINT A$;TAB(15);B$;"  ";C$;"  ";D$;TAB(60);T$;TAB(65);U$
560  NEXT L
570  M1=M1+2
580  NEXT J
590  K1=K1+13
600  NEXT I
610  IF F=1 THEN 630
620  PRINT "NOT IN FILES"
630  PRINT
640  PRINT
650  PRINT
660  PRINT
670  STOP
680  PRINT "TROUBLE"
690  END
9000  REM GET STUDENT OR COURSE FILE.
9030  DIM W$[20],X$[12],Y$[20],Z$[10]
9031  IF  END #1 THEN 9038
9032  X1=1
9033  FOR X2=1 TO 32
9034  READ #1,X1;W$
9035  IF W$=X$ THEN 9040
9036  X1=X1+4
9037  NEXT X2
9038  X=0
9039  RETURN
9040  READ #1,X1
```

```
9045  X=X1
9046  RETURN
9200  IF  END #2 THEN 9207
9201  X1=1
9202  FOR X2=1 TO 9
9203  READ #2,X1;w$
9204  IF w$=Z$ THEN 9209
9205  X1=X1+13
9206  NEXT X2
9207  Z=0
9208  RETURN
9209  READ #2,X1
9210  Z=X1
9211  RETURN
9212  END
```

COPY

```
1   REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2   REM
3   REM         COPY:   COLLEGE REGISTRATION DEMO
4   REM
5   REM         36282 REV  A  PART 10 OF 10   6/73
6   REM
7   REM  ****  CONTRIBUTED PROGRAM  ************************************
8   REM THIS PROGRAM RESTORES THE STUDENT AND COURSE FILE
9   REM       FROM THE BACKUP FILES OF SSAVE AND CSAVE
10   FILES SSAVE,STUDNT,CSAVE,COURSE
20   DIM A$[72]
25   FOR A=1 TO 3 STEP 2
27   B=A+1
30   IF  END #A THEN 170
40   IF  END #B THEN 180
50   FOR I=1 TO 32000
60   READ #A,I
70   PRINT #B,I
80   GOTO TYP(-A) OF 90,120,150,160
90   READ #A;X
100   PRINT #B;X
110   GOTO 80
120   READ #A;A$
130   PRINT #B;A$
140   GOTO 80
150   PRINT #B; END
160   NEXT I
170   NEXT A
175   STOP
180   PRINT "SECOND FILE TOO SMALL"
190   END
```

DRAG
A903-36601A

TITLE: SIMULATES A DRAG RACE

DESCRIPTION:

This program allows the user to design a dragster to his own specifications and then race it against a dragster designed by another user. He may also race it against a "standard" dragster design stored in the computer. A one-quarter mile race is simulated with the computer printing out elapsed time, speed and distance for both vehicles.

INSTRUCTIONS:

Instructions are contained within the program itself. Users are required to answer questions with "yes" or "no" or are asked to enter a numeric value. There are no files used.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Thomas G. Cleaver
University of Louisville

```
RUN


RUN
DRAG

WELCOME TO THE DRAG STRIP.
WOULD YOU LIKE INSTRUCTIONS   ?YES
YOU MAY RACE AGAINST ONE OF YOUR FRIENDS OR YOU MAY RACE
AGAINST MY DRAGSTER.  YOU WILL BE ASKED TO DESIGN YOUR
OWN MACHINE, SPECIFYING HORSEPOWER, REAR END RATIO (X:1),
TIRE WIDTH IN INCHES AND TIRE DIAMETER IN FEET.
DO YOU WANT TO RACE AGAINST ME?YES
I WILL HAVE CAR#1.
DESIGN CAR#2:
HORSEPOWER=     ?500
REAR END RATIO=?4
TIRE WIDTH=     ?15
TIRE DIAMETER= ?3

GO!


ELAPSED         CAR#1                    CAR#2
TIME    SPEED       DISTANCE      SPEED       DISTANCE
(SEC)   (MPH)         (FT)        (MPH)         (FT)

CAR# 2     STOPS BURNING RUBBER
 1      22.0707      16.3821      20.2038      14.9338
 2      43.5753      64.7826      40.2131      59.4559
 3      64.0015      143.982      55.4805      132.817
 4      82.9393      252.079      77.6997      233.693
 5      100.107      386.668      94.6339      360.364
 6      115.356      545.025      110.124      510.816
CAR# 1     STOPS BURNING RUBBER
 7      128.639      724.299      124.088      682.863
 8      139.186      921.156      136.426      874.221
 9      146.922      1131.33      146.855      1082.27
 9.86047      151.81       1320      154.445      1272.53
           WINNER

DO YOU WANT TO TRY AGAIN       ?YES
DO YOU WANT TO RACE AGAINST ME?NO
DESIGN CAR#1:
HORSEPOWER=     ?600
REAR END RATIO=?4.5
TIRE WIDTH=     ?17
TIRE DIAMETER= ?3
DESIGN CAR#2:
HORSEPOWER=     ?500
REAR END RATIO=?4
TIRE WIDTH=     ?2.6
TIRE DIAMETER= ?2.6

GO!


ELAPSED         CAR#1                    CAR#2
TIME    SPEED       DISTANCE      SPEED       DISTANCE
(SEC)   (MPH)         (FT)        (MPH)         (FT)

 1      20.7443      15.3889      14.376       10.6551
 2      41.0949      60.9584      28.6336      42.3217
 3      60.6856      135.855      42.6575      94.7399
 4      79.2044      238.724      56.3401      167.487
 5      96.4112      367.806      69.5843      259.987
 6      112.145      521.05       82.3064      371.536
 7      126.322      696.221      94.4372      501.313
 8      138.929      891.022      105.923      648.409
CAR# 1     STOPS BURNING RUBBER
 9      149.833      1103.13      116.726      811.85
 9.96002      157.772       1320      126.531      983.187
           WINNER

DO YOU WANT TO TRY AGAIN       ?NO

DONE
```

LISTING

DRAG

```
1  REM  ****  HP BASIC PROGRAM LIBRARY  ********************************
2  REM
3  REM         DRAG:   SIMULATES A DRAG RACE
4  REM
5  REM         36601 REV  A   6/73
6  REM
7  REM  ****  CONTRIBUTED PROGRAM  ************************************
10  DIM I$[3],P[2],E[2],W[2],S[2],X[2],M[2],C[2],B[2],Y[2]
20  DIM Q[2]
30  PRINT "WELCOME TO THE DRAG STRIP."
40  PRINT "WOULD YOU LIKE INSTRUCTIONS",
50  INPUT I$
60  IF I$="NO" THEN 110
70  PRINT "YOU MAY RACE AGAINST ONE OF YOUR FRIENDS OR YOU MAY RACE"
80  PRINT "AGAINST MY DRAGSTER.  YOU WILL BE ASKED TO DESIGN YOUR"
90  PRINT "OWN MACHINE. SPECIFYING HORSEPOWER. REAR END RATIO (X:1),"
100  PRINT "TIRE WIDTH IN INCHES AND TIRE DIAMETER IN FEET."
110  PRINT "DO YOU WANT TO RACE AGAINST ME".
120  INPUT I$
130  IF I$="NO" THEN 200
140  PRINT "I WILL HAVE CAR#1."
150  P[1]=600
160  E[1]=5.9
170  W[1]=22
180  D[1]=3.9
190  GOTO 290
200  PRINT "DESIGN CAR#1:"
210  PRINT "HORSEPOWER=",
220  INPUT P[1]
230  PRINT "REAR END RATIO=",
240  INPUT E[1]
250  PRINT "TIRE WIDTH=",
260  INPUT W[1]
270  PRINT "TIRE DIAMETER=",
280  INPUT D[1]
290  PRINT "DESIGN CAR#2:"
300  PRINT "HORSEPOWER=",
310  INPUT P[2]
320  PRINT "REAR END RATIO=",
330  INPUT E[2]
340  PRINT "TIRE WIDTH=",
350  INPUT W[2]
360  PRINT "TIRE DIAMETER=",
370  INPUT D[2]
380  PRINT
390  PRINT "GO!"
400  K1=500
410  K2=1.6
420  K3=2
430  K4=.0006
440  K5=.00006
450  K6=.2
460  K7=4
470  K8=.00015
480  Q[1]=Q[2]=0
490  S[1]=S[2]=0
500  X[1]=X[2]=0
510  REM:M IS MASS
520  FOR J=1 TO 2
530  M[J]=(K1+K2*P[J]+K3*W[J]*D[J]+K7*D[J]↑2)/32.2
540  REM:C IS DRAG FROM WIND
550  C[J]=K4*M[J]↑(2/3)+K8*W[J]*D[J]
560  REM:B IS THE MAX ACCELERATION WITHOUT BURNING
570  B[J]=15+28*W[J]*D[J]/((W[J]+6)*(D[J]+1))
580  REM:Y IS THE SCALE FACTOR FOR RPM VS POWER
590  Y[J]=3.7-.0033*P[J]
600  NEXT J
610  PRINT
620  PRINT
630  PRINT "ELAPSED";TAB(15);"CAR#1";TAB(39);"CAR#2"
640  PRINT "TIME   SPEED      DISTANCE     SPEED      DISTANCE"
650  PRINT "(SEC)  (MPH)        (FT)       (MPH)        (FT)"
660  PRINT
```

```
670  FOR T=0 TO 100 STEP 1
680  FOR T1=1 TO 100 STEP 1
690  FOR J=1 TO 2
700  REM:R IS RPM
710  R=60*S[J]*E[J]/(3.14159*D[J])
720  REM:L0 IS ENGINE TORQUE
730  L0=(P[J]/42.5)*(50+.0078*(R/Y[J])-4.E-10*(R/Y[J])↑3)
740  REM:L1 IS TORQUE FROM FRICTION
750  L1=P[J]*(K5*R+K6)
760  REM:L2 IS REAR AXEL TORQUE
770  L2=E[J]*(L0-L1)
780  REM F IS FORCE ON ROAD FROM TIRES
790  F=2*L2/D[J]
800  REM:TEST FOR BURN
810  IF F>M[J]*B[J] THEN 880
820  REM:A=ACCELERATION
830  IF Q[J] <> 0 THEN 860
840  PRINT "CAR#";J;"STOPS BURNING RUBBER"
850  Q[J]=1
860  A=(F-C[J]*S[J]↑2)/M[J]
870  GOTO 900
880  A=B[J]-C[J]*S[J]↑2/M[J]
890  REM:S=SPEED IN FT/SEC
900  S[J]=S[J]+A*.01
910  REM:X=DISTANCE IN FT
920  X[J]=X[J]+S[J]*.01
930  NEXT J
940  REM:TEST FOR FINISH
950  IF X[1]<5280/4 AND X[2]<5280/4 THEN 1160
960  IF X[1]>X[2] THEN 1080
970  T3=(X[2]-5280/4)/S[2]
980  T=T+T1/100-T3
990  X[2]=5280/4
1000  X[1]=X[1]-S[1]*T3
1010  PRINT T;S[1]*3600/5280;X[1];S[2]*3600/5280;X[2]
1020  PRINT TAB(40);"WINNER"
1030  PRINT
1040  PRINT "DO YOU WANT TO TRY AGAIN".
1050  INPUT I$
1060  IF I$="YES" THEN 110
1070  STOP
1080  T3=(X[1]-5280/4)/S[1]
1090  T=T+T1/100-T3
1100  X[1]=5280/4
1110  X[2]=X[2]-S[2]*T3
1120  PRINT T;S[1]*3600/5280;X[1];S[2]*3600/5280;X[2]
1130  PRINT TAB(10);"WINNER"
1140  PRINT
1150  GOTO 1040
1160  NEXT T1
1170  PRINT T+1;S[1]*3600/5280;X[1];S[2]*3600/5280;X[2]
1180  NEXT T
1190  STOP
1200  END
```

PRINT
F904-36299A

TITLE: GENERATES LARGE LETTERS

DESCRIPTION:

This program prints large block letters. Four different sizes may be selected (2, 3.5, 5, or 7-inch characters).

INSTRUCTIONS:

Type a '1' to the question 'INFORMATION' to receive instructions. When the machine types 'SIZE' type a number from 1 to 4 for characters of the above sizes. When the machine types 'WHAT IS YOUR MESSAGE' input a message from 1 to 72 characters. For quotes use a control K. When the machine types 'METHOD' type 1 for characters printed in the character they represent, and 2 for characters printed in a character of your choice.

SYSTEM SPECIFICATIONS:

2000F and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

Randy Gilbertson
Stillwater Senior High School

```
RUN
PRINT

INFORMATION?1
THIS PROGRAM PRINTS CHARACTERS OF HEIGHT 2, 3.5, 5, OR 7
INCHES. WHEN THE MACHINE TYPES 'SIZE?' TYPE A NUMBER
FROM 1 TO 4 FOR CHARACTERS OF THE ABOVE SIZES. WHEN
THE MACHINE TYPES 'WHAT IS YOUR MESSAGE?' INPUT A
MESSAGE FROM 1 TO 72 CHARACTERS. FOR QUOTES USE A CONTROL
K. WHEN THE MACHINE TYPES 'METHOD?' TYPE (1) CHARACTERS
PRINTED IN THE CHARACTER THEY REPRESENT (2) PRINTED
IN A CHARACTER OF YOUR CHOICE.
SIZE?1
WHAT IS YOUR MESSAGE?
H-P
METHOD?1



HHHHHHHHHHHHHHHHHH
        HH
        HH
        HH
HHHHHHHHHHHHHHHHHH


        --
        --
        --
        --
        --


PPPPPPPPPPPPPPPPP
        PP      PP
        PP      PP
        PP      PP
        PPPPPPPPPP


DONE
RUN
PRINT

INFORMATION?0
SIZE?2
WHAT IS YOUR MESSAGE?
TSB
METHOD?2
INPUT 2 OF YOUR CHARACTERS?HP
```

QUBIC
A903-36106A

TITLE: PLAYS 3-DIMENSIONAL TIC-TAC-TOE

DESCRIPTION:

The game is Tic-Tac-Toe in a 4 by 4 cube.

INSTRUCTIONS:

Each move is indicated by a three digit number, with each digit between 1 and 4 inclusive. The digits indicate the level, row, and column, respectively, of the indicated place. After each pair of moves the computer will print the game board with X's indicating the computer's positions and O's indicating the player's positions. To use, simply type run.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

RUN

```
GET-QUBIC
RUN
QUBIC
THIS IS THE GAME OF QUBIC

DO YOU WANT INSTRUCTIONS?YES

THE GAME IS TIC-TAC-TOE IN A 4 X 4 X 4 CUBE.
EACH MOVE IS INDICATED BY A 3 DIGIT NUMBER, WITH EACH
DIGIT BETWEEN 1 AND 4 INCLUSIVE.  THE DIGITS INDICATE THE
LEVEL, ROW, AND COLUMN, RESPECTIVELY, OF THE OCCUPIED PLACE.


AFTER EACH PAIR OF MOVES I WILL PRINT THE GAME BOARD WITH
X'S INDICATING MY POSITIONS AND O'S INDICATING YOUR POS-
ITIONS.


AFTER I WIN, I WILL PRINT THE BOARD SHOWING THE FINAL
POSITIONS.


DO YOU WANT TO MOVE FIRST?YES
YOUR MOVE?111
MACHINE MOVES TO 411

O...  ....  ....  X...
....  ....  ....  ....
....  ....  ....  ....
....  ....  ....  ....

YOUR MOVE?134
MACHINE MOVES TO 414

O...  ....  ....  X..X
....  ....  ....  ....
...O  ....  ....  ....
....  ....  ....  ....

YOUR MOVE?413
MACHINE MOVES TO 114

O..X  ....  ....  X.OX
....  ....  ....  ....
...O  ....  ....  ....
....  ....  ....  ....

YOUR MOVE?
DONE
```

STORE 7

LISTING

QUBIC

```
9000  REM ***** QUBIC ***** DEMONSTRATION PROGRAM *****
9001  REM ***** VERSION-1.5 ***** 8/14/70 *****
9002  REM  PLAYS 3-DIMENSIONAL TIC-TAC-TOE
9003  DIM A$[72]
9004  LET Z9=0
9005  PRINT "THIS IS THE GAME OF QUBIC"
9006  PRINT "DO YOU WANT INSTRUCTIONS";
9007  INPUT A$
9008  IF A$="NO" THEN 9025
9009  IF A$="YES" THEN 9012
9010  PRINT "PLEASE TYPE 'YES' OR 'NO'";
9011  GOTO 9007
9012  PRINT
9013  PRINT "THE GAME IS TIC-TAC-TOE IN A 4 X 4 X 4 CUBE."
9014  PRINT "EACH MOVE IS INDICATED BY A 3 DIGIT NUMBER, WITH EACH"
9015  PRINT "DIGIT BETWEEN 1 AND 4 INCLUSIVE.  THE DIGITS INDICATE THE"
9016  PRINT "LEVEL, ROW, AND COLUMN, RESPECTIVELY, OF THE OCCUPIED PLACE."
9017  PRINT "AFTER EACH PAIR OF MOVES I WILL PRINT THE GAME BOARD WITH"
9018  PRINT "X'S INDICATING MY POSITIONS AND O'S INDICATING YOUR POS-"
9019  PRINT "ITIONS."
9020  PRINT "AFTER I WIN, I WILL PRINT THE BOARD SHOWING THE FINAL"
9021  PRINT "POSITIONS."
9022  DIM X[64],L[76],M[76,4],Y[16]
9023  DEF FNL(I)=X[M[I,1]]+X[M[I,2]]+X[M[I,3]]+X[M[I,4]]
9024  DEF FNM(M)=M+11*(M>10)+6*INT((M-1)/4)+60*INT((M-1)/16)
9025  MAT  READ Y,M
9026  MAT X=ZER
9027  PRINT "DO YOU WANT TO MOVE FIRST";
9028  INPUT A$
9029  IF A$="YES" THEN 9034
9030  IF A$="NO" THEN 9050
9031  PRINT "PLEASE TYPE 'YES' OR 'NO'";
9032  GOTO 9028
9033  GOSUB 9204
9034  PRINT "YOUR MOVE";
9035  INPUT J1
9036  GOSUB 9200
9037  LET K1=INT(J1/100)
9038  LET J2=(J1-K1*100)
9039  LET K2=INT(J2/10)
9040  LET K3=J1-K1*100-K2*10
9041  IF J1>444 OR J1<111 OR J1 <> INT(J1) THEN 9043
9042  IF K1 <= 4 AND K2 <= 4 AND K3 <= 4 THEN 9045
9043  PRINT "ILLEGAL MOVE, TRY AGAIN"
9044  GOTO 9035
9045  LET M=16*K1+4*K2+K3-20
9046  IF X[M]=0 THEN 9049
9047  PRINT "THAT SQUARE IS USED, TRY AGAIN"
9048  GOTO 9034
9049  LET X[M]=1
9050  GOSUB 9134
9051  LET S=T=0
9052  FOR I=1 TO 76
9053  IF L[I] <> 4 THEN 9061
9054  PRINT "CONGRATULATIONS!"
9055  PRINT "YOU WIN AS FOLLOWS";
9056  PRINT FNM(M[I,1]);FNM(M[I,2]);FNM(M[I,3]);FNM(M[I,4]);
9057  PRINT
9058  PRINT "FINAL POSITION:"
9059  GOSUB 9204
9060  GOTO 9123
9061  IF L[I] <> 15 THEN 9063
9062  LET S=I
9063  IF L[I] <> 3 THEN 9065
9064  LET T=I
9065  NEXT I
9066  IF S=0 THEN 9075
9067  LET I=S
9068  FOR J=1 TO 4
9069  LET M=M[I,J]
9070  IF X[M]>0 THEN 9074
9071  LET X[M]=5
9072  PRINT "MACHINE MOVES TO"FNM(M)", AND WINS AS FOLLOWS";
9073  GOTO 9056
9074  NEXT J
9075  IF T=0 THEN 9107
```

```
9076  [illegible] I=T
9077  FOR J=1 TO 4
9078  [illegible] M=M[I,J]
9079  IF X[M]>0 THEN 9083
9080  [illegible] X[M]=5
9081  PRINT "NICE TRY --MACHINE MOVES TO"FNM(M)
9082  GOTO 9033
9083  NEXT J
9084  FOR I=1 TO 76
9085  [illegible] L=FNL(I)
9086  IF INT(L) <> 2 THEN 9092
9087  IF L>2 THEN 9178
9088  FOR J=1 TO 4
9089  IF X[M[I,J]]>0 THEN 9091
9090  [illegible] X[M[I,J]]=.125
9091  NEXT J
9092  NEXT I
9093  GOSUB 9134
9094  FOR I=1 TO 76
9095  IF L[I]=.5 THEN 9188
9096  IF L[I]=1.375 THEN 9188
9097  NEXT I
9098  GOTO 9146
9099  FOR Z=1 TO 16
9100  IF X[Y[Z]]=0 THEN 9103
9101  NEXT Z
9102  GOTO 9138
9103  [illegible] M=Y[Z]
9104  [illegible] X[M]=5
9105  PRINT "MACHINE MOVES TO"FNM(M)
9106  GOTO 9033
9107  FOR I=1 TO 76
9108  [illegible] L=FNL(I)
9109  IF INT(L) <> 10 THEN 9115
9110  IF L>10 THEN 9178
9111  FOR J=1 TO 4
9112  IF X[M[I,J]]>0 THEN 9114
9113  [illegible] X[M[I,J]]=.125
9114  NEXT J
9115  NEXT I
9116  GOSUB 9134
9117  FOR I=1 TO 76
9118  IF L[I]=.5 THEN 9188
9119  IF L[I]=5.375 THEN 9188
9120  NEXT I
9121  GOSUB 9200
9122  GOTO 9084
9123  [illegible] Z9=Z9+1
9124  IF Z9=2 THEN 9218
9125  PRINT "WOULD YOU LIKE ANOTHER GAME";
9126  INPUT A$
9127  IF A$="YES" THEN 9026
9128  IF A$="NO" THEN 9131
9129  PRINT "PLEASE TYPE 'YES' OR 'NO'";
9130  GOTO 9126
9131  STOP
9132  REM
9133  REM
9134  FOR S=1 TO 76
9135  [illegible] L[S]=FNL(S)
9136  NEXT S
9137  RETURN
9138  FOR M=1 TO 64
9139  IF X[M]>0 THEN 9143
9140  [illegible] X[M]=5
9141  PRINT "MACHINE LIKES"FNM(M)
9142  GOTO 9033
9143  NEXT M
9144  PRINT "THE GAME IS A DRAW"
9145  GOTO 9123
9146  FOR K=1 TO 72 STEP 4
9147  [illegible] P=INT(L[K])+INT(L[K+1])+INT(L[K+2])+INT(L[K+3])
9148  IF P=4 OR P=9 THEN 9152
9149  NEXT K
9150  GOSUB 9200
9151  GOTO 9999
9152  [illegible] S=.125
```

```
9153  FOR I=K TO K+3
9154  GOTO 9189
9155  NEXT I
9156  LET S=0
9157  GOTO 9153
9158  DATA 1,49,52,4,13,61,64,16,22,39,23,38,26,42,27,43
9159  DATA 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20
9160  DATA 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38
9161  DATA 39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56
9162  DATA 57,58,59,60,61,62,63,64
9163  DATA 1,17,33,49,2,18,34,50,3,19,35,51,4,20,36,52
9164  DATA 5,21,37,53,6,22,38,54,7,23,39,55,8,24,40,56
9165  DATA 9,25,41,57,10,26,42,58,11,27,43,59,12,28,44,60
9166  DATA 13,29,45,61,14,30,46,62,15,31,47,63,16,32,48,64
9167  DATA 1,5,9,13,17,21,25,29,33,37,41,45,49,53,57,61
9168  DATA 2,6,10,14,18,22,26,30,34,38,42,46,50,54,58,62
9169  DATA 3,7,11,15,19,23,27,31,35,39,43,47,51,55,59,63
9170  DATA 4,8,12,16,20,24,28,32,36,40,44,48,52,56,60,64
9171  DATA 1,6,11,16,17,22,27,32,33,38,43,48,49,54,59,64
9172  DATA 13,10,7,4,29,26,23,20,45,42,39,36,61,58,55,52
9173  DATA 1,21,41,61,2,22,42,62,3,23,43,63,4,24,44,64
9174  DATA 49,37,25,13,50,38,26,14,51,39,27,15,52,40,28,16
9175  DATA 1,18,35,52,5,22,39,56,9,26,43,60,13,30,47,64
9176  DATA 49,34,19,4,53,38,23,8,57,42,27,12,61,46,31,16
9177  DATA 1,22,43,64,16,27,38,49,4,23,42,61,13,26,39,52
9178  FOR J=1 TO 4
9179  IF X[M[I,J]] <> .125 THEN 9187
9180  LET X[M[I,J]]=5
9181  IF L<5 THEN 9184
9182  PRINT "LET'S SEE YOU GET OUT OF THIS:  MACHINE MOVES TO";
9183  GOTO 9185
9184  PRINT "YOU FOX!  JUST IN THE NICK OF TIME, MACHINE MOVES TO";
9185  PRINT FNM(M[I,J])
9186  GOTO 9033
9187  NEXT J
9188  LET S=.125
9189  IF I-INT(I/4)*4>1 THEN 9192
9190  LET A=1
9191  GOTO 9193
9192  LET A=2
9193  FOR J=A TO 5-A STEP 5-2*A
9194  IF X[M[I,J]]=S THEN 9197
9195  NEXT J
9196  GOTO 9155
9197  LET X[M[I,J]]=5
9198  PRINT "MACHINE TAKES";
9199  GOTO 9185
9200  FOR I=1 TO 64
9201  LET X[I]=INT(X[I])
9202  NEXT I
9203  RETURN
9204  LET A$="0.X"
9205  PRINT
9206  FOR I1=1 TO 13 STEP 4
9207  FOR J1=I1 TO I1+48 STEP 16
9208  FOR K1=J1 TO J1+3
9209  LET L=ABS(X[K1]-2)
9210  PRINT A$[L,L];
9211  NEXT K1
9212  PRINT "  ";
9213  NEXT J1
9214  PRINT " "
9215  NEXT I1
9216  PRINT
9217  RETURN
9218  STOP
9219  END
```

TITLE: PLOTS A GIVEN FUNCTION ON THE TELETYPE

PLOT
A904-36104A

DESCRIPTION:

A program to plot a given function on the terminal. It checks for minumum and maximum Y values over the domain, excluding the undefined points, calculates the Y axis spacing, and plots the function.

INSTRUCTIONS:

Define the function in line 8900 by:

```
DEF FNF(X) = ...
```

Example: 8900 DEF FNF(X) = 25*COS(X)*SIN(X↑2/2)/(X↑2+1)

Type RUN and the program will request the following information:

1. left X end point
2. right X end point
3. desired X increment
4. points on the X-axis for which the function is undefined (for example, the denominator becomes zero)

The output will give:

1. minimum Y
2. maximum Y
3. Y-axis spacing
4. the plot with the Y-axis horizontal and the X-axis vertical on the paper.

SYSTEM SPECIFICATIONS:

2000A and Teletype.

SPECIAL CONSIDERATIONS:

The program will not handle functions where Y is a constant over the entire range.

Error halts and messages:

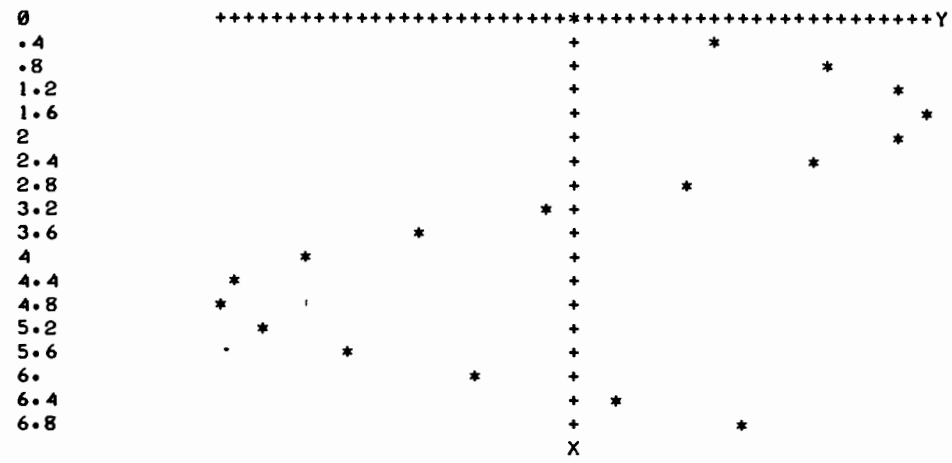
"THIS IS THE CONSTANT FUNCTION" - The value of the function is constant over its range and the program cannot plot it.

"DIVIDE BY ZERO ..." -- The function has an undefined point which was not indicated in the input.

ACKNOWLEDGEMENTS:

RUN

```
GET-$PLOT
8900 DEF FNF(X)=SIN(X)
RUN
PLOT
PLEASE INPUT THE FOLLOWING PARAMETERS:

LEFT X-ENDPOINT?0
RIGHT X-ENDPOINT?7
X-SPACING?.4
THE NUMBER OF UNDEFINED POINTS (IF NONE, ENTER 0)?0


THE MINIMUM VALUE OF THE FUNCTION IS-.996165

THE MAXIMUM VALUE OF THE FUNCTION IS .999574

THE SPACING ON THE Y-AXIS IS 3.99148E-02
```




```
DONE
```

LISTING

PLOT

```
8900  DEF FNF(X)=SIN(X)
9000  REM  *****  PLOT  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  PLOTS A FUNCTION ON THE TTY.
9003  LET R1=0
9004  LET L1=0
9005  LET Q1=0
9006  PRINT "NNPLEASE INPUT THE FOLLOWING PARAMETERS:"
9007  PRINT "NLEFT X-ENDPOINT";
9008  INPUT A
9009  PRINT "RIGHT X-ENDPOINT";
9010  INPUT B
9011  PRINT "X-SPACING";
9012  INPUT D
9013  PRINT "THE NUMBER OF UNDEFINED POINTS (IF NONE, ENTER 0)";
9014  INPUT N9
9015  IF N9=0 THEN 9018
9016  PRINT "ENTER THE UNDEFINED POINTS SEPARATED BY COMMAS"
9017  MAT  INPUT Z[N9]
9018  DEF FNG(X)=INT((Y7-L1)/D1+.5)+15
9019  LET L2=R2=FNF(A)
9020  FOR X=A TO B STEP D
9021  FOR I=1 TO N9
9022  IF X=Z[I] THEN 9028
9023  NEXT I
9024  IF FNF(X)>L2 THEN 9026
9025  LET L2=FNF(X)
9026  IF FNF(X)<R2 THEN 9028
9027  LET R2=FNF(X)
9028  NEXT X
9029  IF L2<0 THEN 9032
9030  LET R1=R2
9031  GOTO 9036
9032  IF R2>0 THEN 9034
9033  GOTO 9035
9034  LET R1=R2
9035  LET L1=L2
9036  LET D1=(R1-L1)/50
9037  IF L1<R1 THEN 9040
9038  PRINT "THIS IS THE FUNCTION Y=CONSTANT."
9039  STOP
9040  PRINT "NNTHE MINIMUM VALUE OF THE FUNCTION IS";L2
9041  PRINT "NTHE MAXIMUM VALUE OF THE FUNCTION IS";R2
9042  PRINT "NTHE SPACING ON THE Y-AXIS IS";D1
9043  PRINT "NNNNN"
9044  LET F=INT(-L1/D1+.5)+15
9045  IF A <= 0 THEN 9056
9046  IF A/D>6 THEN 9056
9047  LET Q1=1
9048  IF L1=0 THEN 9050
9049  PRINT TAB(F);"+"
9050  PRINT
9051  GOTO 9075
9052  FOR I=1 TO INT(A/D-.5)
9053  PRINT TAB(F);"+"
9054  NEXT I
9055  LET Q1=0
9056  FOR X=A TO B STEP D
9057  IF D<.0001 THEN 9060
9058  IF ABS(X)>.00001 THEN 9060
9059  LET X=0
9060  PRINT X,
9061  FOR P=1 TO N9
9062  IF X#Z[P] THEN 9072
9063  IF X#0 THEN 9070
9064  FOR I2=1 TO 50
9065  PRINT "+";
9066  NEXT I2
9067  LET Q=1
9068  PRINT "Y"
9069  GOTO 9103
9070  PRINT TAB(F);"+"
9071  GOTO 9103
9072  NEXT P
9073  IF X*(X+D)>0 THEN 9093
```

```
9074  IF X<-D/2 THEN 9093
9075  FOR I=0 TO 50
9076  IF Q1>0 THEN 9079
9077  LET Y7=FNF(X)
9078  IF FNG(X)=I+15 THEN 9082
9079  IF I+15=F THEN 9084
9080  PRINT "+";
9081  GOTO 9085
9082  PRINT "*";
9083  GOTO 9085
9084  PRINT "O";
9085  NEXT I
9086  IF I+15#F THEN 9088
9087  PRINT "+";
9088  PRINT "Y"
9089  LET Q=1
9090  IF (Q1+1)=1 THEN 9103
9091  IF (Q1+1)=2 THEN 9052
9092  IF (Q1+1)=3 THEN 9112
9093  IF X*(X-D)>0 THEN 9095
9094  IF X <= D/2 THEN 9075
9095  LET Y7=FNF(X)
9096  IF FNG(X)>F THEN 9102
9097  IF FNG(X)=F THEN 9100
9098  PRINT TAB(FNG(X));"*";TAB(F);"+"
9099  GOTO 9103
9100  PRINT TAB(F);"*"
9101  GOTO 9103
9102  PRINT TAB(F);"+";TAB(FNG(X));"*"
9103  NEXT X
9104  IF X >= 0 THEN 9113
9105  IF -X/D>6 THEN 9113
9106  FOR I=1 TO INT(-X/D-.5)
9107  PRINT TAB(F);"+"
9108  NEXT I
9109  LET Q1=2
9110  PRINT
9111  GOTO 9075
9112  PRINT TAB(F);"+"
9113  PRINT TAB(F);"X"
9114  IF Q=0 THEN 9116
9115  STOP
9116  PRINT
9117  PRINT
9118  PRINT
9119  FOR I=0 TO 50
9120  PRINT "+";
9121  NEXT I
9122  PRINT "Y"
9123  PRINT
9124  PRINT
9125  PRINT "                SINCE THE REAL Y-AXIS IS OFF THE GRAPH."
9126  STOP
9999  END
```

LIFE
A903-36138A

TITLE: A GAME OF CELLULAR GENERATION GROWTH

DESCRIPTION:

This is John Conway's Game of Life. Articles further describing his game may be found in the mathematical games section of Scientific American issues of October 70, November 70, and February 71. The game involves cellular generation growth. Each cell is one printed character. A cell is surrounded by eight other cells, in three directions: diagonally, horizontally, vertically. In each generation cells die and grow simultaneously. A cell dies from over population if it is surrounded by more than 3 living cells. It will also die of isolation if it has less than 2 cells surrounding it. New cells can be formed only in empty positions if and only if it has 3 cells around it. The maximum grid size is a 60 x 60 matrix. Each generation is printed as specified at the beginning of the program. Each cell is one printed character. The digit printed (a living cell) is the last digit of the generation it was born in. For example: if a cell was born in generation 14 it would be printed as a 4 thus allowing you to tell, within ten generations what was born during each generation. A dead cell is removed from the grid and replaced with a blank (not printed). It might be noted that the rate of growth in any direction is limited to one cell at any direction per generation.

INSTRUCTIONS:

The maximum grid size is 60 x 60 cells. Any life pattern exceeding this will either lose accuracy or terminate the program depending if it is too large horizontally or vertically. The program requires that two files be opened: "Open-A,62" "OPEN-B,62" At the start of the program you will be asked information about the printing of each generation (interval and start).

The row of the life pattern is simply the row in which certain cells are living. In entering a life pattern type a "1" for a living cell, blanks will be assumed to be empty cells. Enter your cells in quotes (otherwise left blanks will be ignored). Type "DONE" in response to the row number to terminate your input.

ERRORS: it is possible for a life pattern to grow off to one side of the grid. Error notices will be given before the number of the generation if this should occur. Adjustment errors of this type will simply move the pattern the specified direction automatically, thus always keeping the pattern within the grid. Note, if you input your pattern starting a row one, the program will move your pattern down two rows thus allowing for an upward growth, similarly to both the left and right sides of the grid.

When requesting the interval of print and start of print the first generation will ALWAYS be printed thus allowing you to check to see that your inputted pattern was correct.

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS:

REFERENCE: John Conway's Game of Life. Articles further describing his game may be found in the mathematical games section of Scientific American issues of October 1970, November 1970, and February 1971.

ACKNOWLEDGEMENTS: Richard Suslick
N.O.S.C.O.M.P.

```
RUN

OPEN-A,62
OPEN-B,62
RUN
LIFE


  T H I S   I S   L I F E   ------------WELCOME------------

DO YOU NEED INSTRUCTIONS?NO
IS INPUT TO BE FIRST GENERATION?YES
WHAT IS # OF LAST GENERATION TO BE COMPUTED?10
WHAT IS THE # OF THE FIRST GENERATION TO BE PRINTED?1
WHAT IS THE INTERVAL OF PRINTED GENERATIONS YOU WANT?1
WHAT IS THE # OF THE FIRST ROW OF LIFE PATTERN?26
ROW 26   ?"         111 111 111"
ROW 27   ?"         111 111 111"
ROW 28   ?"         111 111 111"
ROW 29   ?"  "
ROW 30   ?"         111 111 111"
ROW 31   ?DONE


GENERATION # 1

ROW 26              111 111 111
ROW 27              111 111 111
ROW 28              111 111 111
ROW 29
ROW 30              111 111 111


GENERATION # 2

ROW 25                2   2   2
ROW 26               1 1 1 1 1 1
ROW 27              2             2
ROW 28               1 1 1 1 1 1
ROW 29
ROW 30                1   1   1
ROW 31                2   2   2


GENERATION # 3

ROW 25                2   2   2
ROW 26               13   3   31
ROW 27              23          32
ROW 28
ROW 29                3   3   3


GENERATION # 4

ROW 25               42        24
ROW 26              4 3        3 4
ROW 27              234        432
ROW 28               4          4


GENERATION # 5

ROW 25               42        24
ROW 26              4  5      5  4
ROW 27              2 4        4 2
ROW 28              545        545


GENERATION # 6

ROW 25               42        24
ROW 26              4  5      5  4
ROW 27             62 46      64 26
ROW 28              5 5        5 5
ROW 29               6          6
```

```
GENERATION # 7

ROW 25              42          24
ROW 26            74  5       5  47
ROW 27            62 46       64 26
ROW 28            75 57       75 57
ROW 29              6           6


GENERATION # 8

ROW 25              842         248
ROW 26             7   5       5   7
ROW 27            8     8     8     8
ROW 28             7   7       7   7
ROW 29              868         868


GENERATION # 9

ROW 24              9           9
ROW 25              842         248
ROW 26             79995       59997
ROW 27            89   98     89   98
ROW 28             79997       79997
ROW 29              868         868
ROW 30              9           9


GENERATION # 10

ROW 24              090         090
ROW 25             0   0       0   0
ROW 26            0     0     0     0
ROW 27            8     8     8     8
ROW 28            0     0     0     0
ROW 29             0   0       0   0
ROW 30              090         090


DONE

LISTING

LIFE
1  REM *** HP TIME-SHARED BASIC PROGRAM LIBRARY **********************
2  REM
3  REM          LIFE:  A GAME OF CELLULAR GENERATION GROWTH
4  REM
5  REM          36138 (A903) REV A -- 7471
6  REM
7  REM *** CONTRIBUTED PROGRAM ***************************************
10  FILES A,B
20  DIM A$[62],B$[62],C$[62],X$[62]
30  PRINT
40  PRINT "  T H I S   I S   L I F E   ------------WELCOME-----------"
50  PRINT
60  M4=M5=S4=0
70  GOTO 1410
80  A$="                                                            "
90  FOR R=1 TO 62
100  PRINT #1,R;A$
110  PRINT #2,R;A$
120  NEXT R
130  PRINT "WHAT IS THE # OF THE FIRST ROW OF LIFE PATTERN";
140  INPUT R
150  F1=R-1
160  PRINT "ROW"R;TAB(8);
170  INPUT B$
180  IF B$="DONE" THEN 280
190  IF R>60 THEN 280
200  IF LEN(B$)=62 THEN 230
210  Q=62-LEN(B$)
220  B$[LEN(B$)+1,62]=A$[1,Q]
230  C$[1,1]=" "
240  C$[2,62]=B$[1,61]
```

```
250  PRINT #2,R;C$
260  R=R+1
270  GOTO 160
280  L1=R
290  PRINT "[redacted]"
300  T9=0
310  FOR Z=01 TO 02
320  IF T9#0 THEN 350
330  Z=Z-1
340  GOTO 810
350  IF F1<2 THEN 1910
360  IF L1>59 THEN 2080
370  READ #1,F1-1;B$
380  READ #1,F1;C$
390  READ H$
400  M4=M5=S4=0
410  IF H$#"1" THEN 430
420  RESTORE
430  FOR R=F1 TO L1
440  A$=B$
450  B$=C$
460  READ #1,R+1;C$
470  X$[1,1]=" "
480  FOR Q=2 TO 61
490  M=0
500  FOR N=Q-1 TO Q+1
510  IF A$[N,N]=" " THEN 530
520  M=M+1
530  NEXT N
540  IF B$[Q-1,Q-1]=" " THEN 560
550  M=M+1
560  IF B$[Q+1,Q+1]=" " THEN 580
570  M=M+1
580  FOR N=Q-1 TO Q+1
590  IF C$[N,N]=" " THEN 610
600  M=M+1
610  NEXT N
620  IF M#3 THEN 660
630  IF B$[Q,Q]#" " THEN 680
640  X$[Q,Q]=H$
650  GOTO 710
660  IF B$[Q,Q]=" " THEN 700
670  IF M#2 THEN 700
680  X$[Q,Q]=B$[Q,Q]
690  GOTO 710
700  X$[Q,Q]=" "
710  NEXT Q
720  IF X$[2,2]=" " THEN 740
730  M4=M4+1
740  IF X$[61,61]=" " THEN 760
750  M5=M5+1
760  X$[62,62]=" "
770  PRINT #2,R;X$
780  NEXT R
790  IF M4#0 THEN 1720
800  IF M5#0 THEN 1800
810  S9=1
820  T1=F1-1
830  IF F1-1 >= 1 THEN 850
840  T1=T1+1
850  T2=L1
860  GOSUB 1160
870  R1=R
880  T1=L1
890  S9=-1
900  T2=F1
910  GOSUB 1160
920  IF R >= R1 THEN 950
930  PRINT "LIFE NOW DEAD -- GENERATION #"Z
940  STOP
950  F1=R1-1
960  L1=R+1
970  IF T9=0 THEN 990
980  IF Z<E9 OR (Z-E9)/F9#INT((Z-E9)/F9) THEN 1090
990  T9=1
1000  PRINT "GENERATION #"Z;"N"
1010  FOR T1=R1 TO R
1020  READ #2,T1;X$
```

```
1030  FOR M=61 TO 2 STEP -1
1040  IF X$[M,M]#" " THEN 1060
1050  NEXT M
1060  PRINT "ROW"T1;TAB(10);X$[1,M]
1070  NEXT T1
1080  PRINT "N"
1090  FOR R=1 TO 62
1100  READ #2,R;X$
1110  PRINT #1,R;X$
1120  NEXT R
1130  NEXT Z
1140  DATA "2","3","4","5","6","7","8","9","0","1","X"
1150  END
1160  FOR R=T1+1 TO T2 STEP S9
1170  READ #2,R;X$
1180  IF X$#"                                                                  " THEN 1200
1190  NEXT R
1200  RETURN
1210  PRINT "IS INPUT TO BE FIRST GENERATION";
1220  INPUT A$
1230  IF A$#"NO" THEN 1330
1240  PRINT "WHAT IS THE # OF THE GENERATION TO BE INPUTED";
1250  INPUT O1
1260  O1=O1-1
1270  FOR R=2 TO O1-1
1280  READ H$
1290  IF H$#"1" THEN 1310
1300  RESTORE
1310  NEXT R
1320  GOTO 1340
1330  O1=2
1340  PRINT "WHAT IS # OF LAST GENERATION TO BE COMPUTED";
1350  INPUT O2
1360  PRINT "WHAT IS THE # OF THE FIRST GENERATION TO BE PRINTED";
1370  INPUT E9
1380  PRINT "WHAT IS THE INTERVAL OF PRINTED GENERATIONS YOU WANT";
1390  INPUT F9
1400  GOTO 80
1410  PRINT "DO YOU NEED INSTRUCTIONS";
1420  INPUT X$
1430  IF X$="YES" THEN 1460
1440  IF X$="NO" THEN 1210
1450  GOTO 1660
1460  PRINT "THIS IS JOHN CONWAY'S GAME OF LIFE.  ARTICLES FURTHER DESCRIBING"
1470  PRINT "HIS GAME MAY BE FOUND IN THE MATHEMATICAL GAMES SECTION OF"
1480  PRINT "SCIENTIFIC AMERICAN ISSUES: NOOCTOBER 70ONNOVEMBER 70ONFEBRUARY 71"
1490  PRINT "THE GAME INVOLVES CELLULAR GENERATION GROWTH.  EACH CELL IS"
1500  PRINT "ONE PRINTED CHARACTER.  A CELL IS SURROUNDED BY EIGHT OTHER"
1510  PRINT "CELLS, IN THREE DIRECTIONS: DIAGONALLY,HORIZONTALLY,VERTICALLY."
1520  PRINT "IN EACH GENERATION CELLS DIE AND GROW SIMULTANEOUSLY."
1530  PRINT "A CELL DIES FROM OVER POPULATION IF IT IS SURROUNDED BY MORE"
1540  PRINT "THAN 3 LIVING CELLS.  IT WILL ALSO DIE OF ISOLATION IF IT HAS"
1550  PRINT "LESS THAN 2 CELLS SURROUNDING IT.  NEW CELLS CAN BE FORMED ONLY"
1560  PRINT "IN EMPTY POSITIONS IF AND ONLY IF IT HAS 3 CELLS AROUND IT."
1570  PRINT "THE MAXIMUM GRID SIZE IS A 60X60 MATRIX."
1580  PRINT "INPUT YOUR CELLS IN QUOTES --- SPACING AND PRINTING A "
1590  PRINT "  '1' WHERE YOU WANT A LIVING CELL.  BLANK POSITIONS WILL"
1600  PRINT "BE ASSUMED TO BE EMPTY CELLS.  WHEN YOU ARE FINISHED "
1610  PRINT "TYPE 'DONE' IN RESPONSE TO THE ROW NUMBER"
1620  PRINT "------NOTE-----NOTHIS PROGRAM REQUIRES THAT YOU OPEN 2 FILES.  IF "
1630  PRINT "I COME BACK WITH A MESSAGE STATING: YOU HAVE REQUESTED A NON-EXISTANT"
1640  PRINT "FILE.  TYPE:NOOPEN-A,62NOOPEN-B,62NO1 FILES A,BNOTHEN TRY RENNUNING"" THE PROGRAM."
1650  PRINT "YOU NEED NOT ASK FOR INSTRUCTIONS AGAIN.""NONGOOD LUCK......HERE WE GONNN"
1660  READ #1,1
1670  GOTO 1210
1680  PRINT "PLEASE TYPE A 'GYESG' OR 'GNOG' ANSWER."
1690  GOTO 1410
1700  PRINT "BOTH LEFT AND RIGHT ADJUSTMENT ERRORS...PROGRAM MAY NOT BE ACCURATE"" FROM HERE ON"
1710  GOTO 810
1720  PRINT "LEFT ADJUSTMENT ERROR, CORRECTION BEING MADE A SPACE TO RIGHT"
1730  FOR D=F1 TO L1
1740  READ #1,D;A$
1750  A$[1,1]=" "
1760  A$[2,62]=A$[1,62]
1770  PRINT #1,D;A$
1780  NEXT D
1790  GOTO 1860
1800  PRINT "RIGHT ADJUSTMENT ERROR, CORRECTION BEING MADE A SPACE TO LEFT"
```

```
1810  FOR D=F1 TO L1
1820  READ #1,D;A$
1830  A$[1,62]=A$[2,62]
1840  PRINT #1,D;A$
1850  NEXT D
1860  READ #1,F1-1;B$
1870  M4=M5=0
1880  READ #1,F1;C$
1890  S4=1
1900  GOTO 430
1910  IF L1=60 THEN 2210
1920  B$[1,62]=" "
1930  PRINT "TOP ADJUSTMENT ERROR--CORRECTION BEING MADE A ROW LOWER"
1940  READ #1,1;A$
1950  PRINT #1,1;B$
1960  FOR R=1 TO 59
1970  READ #1,R+1;B$
1980  PRINT #1,R+1;A$
1990  A$=B$
2000  NEXT R
2010  B$[1,62]=" "
2020  F1=F1+1
2030  L1=L1+1
2040  FOR R=1 TO 62
2050  PRINT #2,R;B$
2060  NEXT R
2070  GOTO 350
2080  PRINT "BOTTOM ADJUSTMENT ERROR--CORRECTION BEING MADE A ROW UP"
2090  READ #1,60;A$
2100  B$[1,62]=" "
2110  PRINT #1,60;B$
2120  FOR R=60 TO 2
2130  READ #1,R-1;B$
2140  PRINT #1,R-1;A$
2150  A$=B$
2160  NEXT R
2170  L1=L1-1
2180  F1=F1-1
2190  B$[1,62]=" "
2200  GOTO 2040
2210  PRINT "BOTH TOP AND BOTTOM ADJUSTMENT ERRORS PROGRAM SPECIFICATIONS ARE"
2220  PRINT " BEING EXCEEDED ===PROGRAM TERMINATION===  **SORRY**"
2230  END
```

ROOTER
A311-36024A

TITLE: FINDS THE ROOTS OF POLYNOMIALS

DESCRIPTION:

This program finds the roots of a polynomial using Barstow's Method.

INSTRUCTIONS:

Before running the program supply data as follows:

9900 DATA $N, A_N, A_{N-1}, \ldots, A_1, A_\emptyset$

99xx DATA $\emptyset$

where N is the order of the polynomial

A_i is the coefficient of the ith term of the polynomial of the form

$$A_N X^N + A_{N-1} X^{N-1} + \ldots + A_1 X + A_\emptyset$$

This program will solve for the roots of as many polynomials as desired, and will terminate execution when reading a value for N of zero ($\emptyset$).

In cases where the program is not converging to a solution the user will be asked if he wishes to continue or go to the next polynomial.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

There are some forms of polynomials for which this program cannot find the roots. If this condition occurs the program will indicate this and continue to the next polynomial.

For high order polynomials the running time may be excessive since many iterations may be required.

ACKNOWLEDGEMENTS:

RUN

```
GET-$ROOTER
9900 DATA 3
9901 DATA 1,6,11,6
9902 DATA 2
9903 DATA 1,0,1
9904 DATA 0
RUN
ROOTER

POLYNOMIAL NUMBER 1     IS OF ORDER 3

   COEFFICIENTS (IN DESCENDING ORDER) ARE:

       1      6      11     6

   THE ROOTS ARE:

      -3.
      -.999998      AND  -2.


POLYNOMIAL NUMBER 2     IS OF ORDER 2

   COEFFICIENTS (IN DESCENDING ORDER) ARE:

       1      0      1

   THE ROOTS ARE:

       0     + J * 1      AND    0     - J * 1


DONE
```

LISTING

```
ROOTER

9000  REM  *****  ROOTER  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  FINDS THE ROOTS OF POLYNOMIALS
9003  DIM A[26],B[26],X[26]
9004  LET G8=0
9005  PRINT
9006  LET C1=0
9007  LET D1=1
9008  LET O=0
9009  LET G8=G8+1
9010  PRINT
9011  READ N
9012  IF N=0 THEN 9133
9013  PRINT
9014  PRINT "POLYNOMIAL NUMBER";G8;"IS OF ORDER";N
9015  PRINT
9016  PRINT "   COEFFICIENTS (IN DESCENDING ORDER) ARE:"
9017  PRINT
9018  PRINT "      ";
9019  FOR I=1 TO N+1
9020  READ A[I]
9021  PRINT A[I];
9022  LET B[I]=A[I]
9023  NEXT I
9024  PRINT
9025  PRINT
9026  PRINT "   THE ROOTS ARE:"
9027  PRINT
9028  IF N <= 2 THEN 9093
9029  IF A[N+1]=0 THEN 9101
9030  IF (N/2-INT(N/2))=0 THEN 9033
9031  GOSUB 9109
9032  GOTO 9028
9033  IF ABS(A[N-1])<1.E-25 THEN 9037
9034  LET P=A[N]/A[N-1]
```

```
9035  LET Q=A[N+1]/A[N-1]
9036  GOTO 9039
9037  LET P=A[N]
9038  LET Q=A[N+1]
9039  FOR I=1 TO N+1
9040  LET X[I]=A[I]
9041  NEXT I
9042  GOSUB 9104
9043  FOR I=1 TO N-1
9044  LET B[I]=X[I]
9045  NEXT I
9046  LET R=X[N]
9047  LET S=A[N+1]-P*X[N]-Q*X[N-1]
9048  GOSUB 9104
9049  LET X[N]=P*X[N-1]-Q*X[N-2]
9050  LET D=X[N-1]↑2-X[N]*X[N-2]
9051  IF ABS(D)>1.E-25 THEN 9054
9052  PRINT "SOLUTIONS UNOBTAINABLE WITH THIS PROGRAM."
9053  GOTO 9005
9054  LET P1=P+(R*X[N-1]-S*X[N-2])/D
9055  LET Q1=Q+(S*X[N-1]-R*X[N])/D
9056  IF ABS(P)>1.E-25 THEN 9060
9057  IF ABS(P1)>1.E-25 THEN 9060
9058  IF ABS(Q)>1.E-25 THEN 9061
9059  GOTO 9062
9060  IF ABS(P1/P-1)>.000001 THEN 9062
9061  IF ABS(Q1/Q-1)<.000001 THEN 9076
9062  LET P=P1
9063  LET Q=Q1
9064  LET C1=C1+1
9065  IF C1=D1*25 THEN 9067
9066  GOTO 9039
9067  PRINT
9068  PRINT "THE SOLUTION DID NOT CONVERGE AFTER";C1;"ITERATIONS TO"
9069  PRINT "CONTINUE THE SOLUTION FOR 25 MORE ITERATIONS TYPE 1 OTHERWISE"
9070  PRINT "TYPE 0."
9071  INPUT K1
9072  IF K1=1 THEN 9074
9073  GOTO 9005
9074  LET D1=D1+1
9075  GOTO 9039
9076  FOR I=2 TO N-1
9077  LET A[I]=B[I]
9078  NEXT I
9079  LET N=N-2
9080  LET D=P*P-4*Q
9081  IF D<0 THEN 9088
9082  LET D=SQR(D)
9083  PRINT "       "$(-P+D)/2$" AND  "$(-P-D)/2
9084  LET C1=0
9085  LET D1=1
9086  IF N-2>0 THEN 9029
9087  GOTO 9093
9088  LET D=SQR(-D)
9089  PRINT "       "$-P/2$"+ J *"$D/2$" AND  "$-P/2$"- J *"$D/2
9090  LET C1=0
9091  LET D1=1
9092  IF N-2>0 THEN 9029
9093  IF N=1 THEN 9099
9094  IF N=0 THEN 9005
9095  LET P=B[2]/B[1]
9096  LET Q=B[3]/B[1]
9097  LET N=0
9098  GOTO 9080
9099  PRINT "       "$-B[2]/B[1]
9100  GOTO 9005
9101  PRINT "        0.00000"
9102  LET N=N-1
9103  GOTO 9028
9104  LET X[2]=X[2]-P*X[1]
9105  FOR I=3 TO N
9106  LET X[I]=X[I]-P*X[I-1]-Q*X[I-2]
9107  NEXT I
9108  RETURN
9109  IF B[2]=0 THEN 9112
9110  LET X=-B[2]/B[1]
9111  GOTO 9113
9112  LET X=-B[N+1]/B[1]
```

```
9113  LET F=0
9114  LET F1=0
9115 FOR I=1 TO N+1
9116 LET J=N-I+2
9117 IF B[J]=0 THEN 9121
9118 LET F=B[J]*X↑(I-1)+F
9119 IF I-1=0 THEN 9121
9120 LET F1=(I-1)*B[J]*X↑(I-2)+F1
9121 NEXT I
9122  LET X1=X-F/F1
9123  IF ABS(X/X1-1)<.000001 THEN 9126
9124  LET X=X1
9125  GOTO 9113
9126  PRINT "      ";X1
9127  LET N=N-1
9128 FOR I=2 TO N+1
9129 LET A[I]=B[I]+X1*A[I-1]
9130 LET B[I]=A[I]
9131 NEXT I
9132  RETURN
9133  STOP
9900  DATA 3,1,6,11,6,2,1,0,1,0
9999  END
```

TITLE: SOLVES SIMULTANEOUS LINEAR EQUATIONS

SIMEQN
A314-36026A

DESCRIPTION:

This program solves systems consisting of N linear equations in N unknowns.

INSTRUCTIONS:

To use, coefficients in the equations are entered in data-statements, starting in statement number 9901 with the first coefficient of the first equation, and ending with the N-th coefficient of the N-th equation. All zero coefficients must be entered in their proper place. The right-side constant terms of the equations are then entered in subsequent data statements. If additional cases with the same coefficient matrix but different right sides are to be run, they may all be run at once by simply entering additional data statements with the right-side values of the other cases. The data statement in line 9900 (preceding all the above) is used to specify the number of systems to be solved, and the number of equations (and hence variables) in the system. Thus, the two systems:

```
          3X + 5Y -2Z = 9          3X + 5Y -2Z = 19
          7X +  Y     =-3          7X +  Y     = -3
           X - 7Y + 9Z = 14          X - 7Y +9Z =  8   could be solved
                                                       by typing:
9900 DATA 2,3
9901 DATA 3,5,-2,7,1,0,1,-7,9
9902 DATA 9,-3,14,19,-3,8          Solutions are proofed by multiplying
                                   the vector by the original coefficient
                                   matrix.
```

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

"NEARLY SINGULAR MATRIX IN LINE 9055" means solution of the system is not possible with the program.

ACKNOWLEDGEMENTS:

RUN

```
GET-$SIMEQN
9900 DATA 2,3
9901 DATA 3,5,-2,7,1,0,1,-7,9
9902 DATA 9,-3,14,19,-3,8
RUN
SIMEQN

LINEAR SYSTEM SOLVER

DO YOU WANT INSTRUCTIONS (YES OR NO)?NO

SOLUTION FOR LINEAR SYSTEM OF ORDER 3

COEFFICIENT MATRIX

 3              5              -2

 7              1               0

 1             -7               9
DEPENDENT VARIABLE VECTOR FOR CASE 1

 9             -3               14

SOLUTION VECTOR FOR CASE 1

                VARIABLES

  X1             X2              X3
-1.07447       4.52128         5.19149

PROOF OF SOLUTION FOR CASE 1

 9.00001       -3               14.


DEPENDENT VARIABLE VECTOR FOR CASE 2

 19            -3               8

SOLUTION VECTOR FOR CASE 2

                VARIABLES

  X1             X2              X3
-1.48936       7.42553         6.82979

PROOF OF SOLUTION FOR CASE 2

 19.           -3               7.99999


DONE
```

LISTING

```
SIMEQN

9000  REM  *****  SIMEQN  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  SOLVES SIMILTANEOUS LINEAR EQUATIONS
9003  DIM A[25,25],B[25,25],X[25,1],Y[1,25],Z[25,1],A$[3],B$[72]
9004  PRINT "LINEAR SYSTEM SOLVER"
9005  PRINT "ONDO YOU WANT INSTRUCTIONS (YES OR NO)";
9006  INPUT A$
9007  IF A$="NO" THEN 9046
9008  GOTO 9046
9009  PRINT
9010  PRINT "     THIS PROGRAM SOLVES SYSTEMS CONSISTING"
9011  PRINT "OF N LINEAR EQUATIONS IN N UNKOWNS. TO USE,"
9012  PRINT "COEFFICIENTS IN THE EQUATIONS ARE ENTERED"
9013  PRINT "IN DATA STATEMENTS, STARTING IN STATEMENT"
9014  PRINT "NUMBER 9901  WITH THE FIRST COEFFICIENT OF THE "
9015  PRINT "FIRST EQUATION, AND ENDING WITH THE N-TH"
```

```
9016  PRINT "COEFFICIENT OF THE N-TH EQUATION.  ALL ZERO"
9017  PRINT "COEFFICIENTS MUST BE ENTERED IN THEIR PROPER"
9018  PRINT "PLACE.  THE RIGHT-SIDE CONSTANT TERMS OF"
9019  PRINT "THE EQUATIONS ARE THEN ENTERED IN SUBSEQUENT"
9020  PRINT "DATA STATEMENTS.  IF ADDITIONAL CASES WITH"
9021  PRINT "THE SAME COEFFICIENT MATRIX BUT DIFFERENT"
9022  PRINT "RIGHT SIDES ARE TO BE RUN, THEY MAY ALL BE"
9023  PRINT "RUN AT ONCE BY SIMPLY ENTERING ADDITIONAL"
9024  PRINT "DATA STATEMENTS WITH THE RIGHT-SIDE VALUES"
9025  PRINT "OF THE OTHER CASES. THE DATA STATEMENT IN LINE"
9026  PRINT "9900 (PRECEDING ALL THE ABOVE) IS USED TO SPECIFY"
9027  PRINT "THE NUMBER OF SYSTEMS TO BE SOLVED, AND THE"
9028  PRINT "NUMBER OF EQUATIONS (AND HENCE VARIABLES)"
9029  PRINT "IN THE SYSTEM.  THUS, THE TWO SYSTEMS:"
9030  PRINT
9031  PRINT "  3X + 5Y - 2Z =  9     3X + 5Y - 2Z = 19"
9032  PRINT "  7X +  Y      = -3     7X +  Y      = -3"
9033  PRINT "   X - 7Y + 9Z = 14      X - 7Y + 9Z =  8"
9034  PRINT
9035  PRINT "COULD BE SOLVED BY TYPING:"
9036  PRINT
9037  PRINT
9038  PRINT "    9900  DATA 2,3"
9039  PRINT "    9901  DATA 3,5,-2,7,1,0,1,-7,9"
9040  PRINT "    9902  DATA 9,-3,14,19,-3,8"
9041  PRINT "    RUN"
9042  PRINT
9043  PRINT "SOLUTIONS ARE PROOFED BY MULTIPLYING THE"
9044  PRINT "VECTOR BY THE ORIGINAL COEFFICIENT MATRIX."
9045  STOP
9046  READ L,M
9047  LET B$="1 2 3 4 5 6 7 8 9 10111213141516171819202122232425"
9048  PRINT
9049  PRINT "SOLUTION FOR LINEAR SYSTEM OF ORDER";M
9050  PRINT "ONCOEFFICIENT MATRIXON"
9051  MAT  READ A[M,M]
9052  MAT  PRINT A
9053  PRINT
9054  MAT B=ZER[M,M]
9055  MAT B=INV(A)
9056  MAT Y=ZER[1,M]
9057  MAT Z=ZER[M,1]
9058  LET Q=0
9059  LET Q=Q+1
9060  PRINT "ONDEPENDENT VARIABLE VECTOR FOR CASE";Q
9061  PRINT
9062  MAT  READ X[M,1]
9063  MAT Y=TRN(X)
9064  MAT  PRINT Y
9065  MAT Z=B*X
9066  MAT Y=TRN(Z)
9067  PRINT
9068  PRINT
9069  PRINT "SOLUTION VECTOR FOR CASE";Q
9070  PRINT "ON";TAB(17);"VARIABLESON"
9071  FOR K=1 TO M
9072  PRINT "  X";B$[2*K-1,2*K],
9073  NEXT K
9074  PRINT
9075  MAT  PRINT Y
9076  PRINT "PROOF OF SOLUTION FOR CASE"Q
9077  PRINT
9078  MAT X=A*Z
9079  MAT Y=TRN(X)
9080  MAT  PRINT Y
9081  IF Q<L THEN 9059
9082  STOP
9900  DATA 2,3
9901  DATA 3,5,-2,7,1,0,1,-7,9
9902  DATA 9,-3,14,19,-3,8
9999  END
```

SALSIM
A720-36072A

TITLE: SALARY SCHEDULE COST SIMULATOR

DESCRIPTION:

This program computes the total cost, average salary per teacher, cost of training and experience increments, and various other related costs for a proposed salary schedule. Output also includes optional reproduction of personnel matrix with marginal totals and average number of years teaching experience.

INSTRUCTIONS:

This program requires two files to be opened (ML0005 and ML0017).

In preparing the data tapes type all figures for step 1 in the first data statement, type all figures for step 2 in the next data statement, etc. (Steps and lanes are defined as follows:)

	LANE		
STEP	7000=	7200	7400 . . .
	7400		
	7800		
	:		
	:		

The question "what number do you want to multiply your schedule by?" allows two options.

Option 1 — To multiply the entire schedule by a percentage increase. (Example: Type in 1.03 to multiply the entire schedule. by 3%, 1.05 to multiply by 5%, etc.)

Option 2 — To use an index instead of a proposed schedule. If you want to use an index in place of the proposed schedule just follow the same format for preparing the data tape substituting indices for actual salaries.

Because two sets of data are appended to the program it is important that they are numbered correctly.

The "PERMAT" tape (personnel matrix) must be numbered from 4000 to 6900.

The "PROSCH" tape (proposed schedule) must be numbered from 7000 to 9900.

Because these two tapes are appended you must also include an "end"statement (9999 end) either on the end of the "PROSCH" tape or by itself after the two tapes have been appended.

Both the proposed schedule and the personnel matrix may not exceed 15 steps and 14 lanes. Both the proposed schedule and the personnel matrix must have the same Number of steps and lanes.

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS: See attached copy for additional information.

ACKNOWLEDGEMENTS: TIES
St. Paul, Minnesota 55113

SALSIM

(SALary schedule cost SIMulator)

Purpose

To compute the total and related cost for a proposed salary schedule.

Input

1. A copy of the proposed salary schedule.
2. A personnel matrix for your district (number of teachers in each salary category).

Output

1. Optional -- A step-by-lane reproduction of the personnel matrix with marginal totals and the average number of years teaching experience.
2. A cost matrix showing the cost in each category with step and lane marginal totals.
3. The total cost of the proposed schedule.
4. The cost if all teachers were placed on step one.
5. The cost of experience increments.
6. The cost if all teachers were placed in lane one.
7. The cost of training increments.
8. The ratio of cost of experience increments to the cost of training increments. (A number greater than 1 means that more money is being spent on experience than on training. A number equal to 1 means that money being spent on experience is equal to money being spent on training. A number less than 1 means that more money is being spent on training than on experience).
9. The average salary for teachers .

Options

There is now a question within the program that asks WHAT NUMBER DO YOU WANT TO MULTIPLY YOUR SCHEDULE BY? This question gives you two options.

1. To multiply your whole schedule by a percentage increase. (Example: Type in 1.03, when the question is asked, to multiply your whole schedule by 3%, 1.05 to multiply by 5%, etc.)
2. To use an index instead of a proposed schedule. When using an index in place of the proposed schedule follow FORMAT B IN SECTION I UNDER OFF-LINE PROCEDURE. When the above question is asked type in the BA base figure you want to multiply your index by. (Type in 7500 to multiply the entire schedule by $7500, 8000 to multiply by $8000, etc.)
3. Type in a 1 if you do not want to multiply your proposed schedule (i.e. if you do not want to take advantage of these options.)

Restrictions

Both the proposed schedule and the personnel matrix may not exceed 15 steps and 14 lanes. Both the proposed schedule and the personnel matrix must have the same number of steps and the same number of lanes.

PRELIMINARY OFF-LINE PROCEDURE FOR RUNNING ALL SALARY PROGRAMS

The following two sections I and II, are instructions for preparing tapes off-line.

To run SALSIM follow sections I and II.

I. Proposed Schedule - PROPS1

You must type a tape of your proposed schedule using the following format.

Type each step of your proposed schedule in one data statement.

NOTE: DATA STATEMENTS FOR THIS FILE MUST BEGIN AT STATEMENT NUMBER 7000 AND CONTINUE AT ANY INTERVAL TO 9000. THE LAST STATEMENT MUST BE A 9999 END.

Format A

```
7000 DATA 7500, 7700, 7900, 8100, 8400, 8600, 8800
7100 DATA 7900, 8100, 8300, 8500, 8900, 9100, 9300
7200 DATA 8300, 8500, 8700, 8900, 9400, 9600, 9800
 .    .
 .    .
 .    .
8700 DATA 10900, 11300, 11700, 12100, 13500, 14500, 15800
9999 END
```

If you want to use an index in place of the proposed schedule you must follow the format below.

Format B

```
7000 DATA 1,1.05, 1.1, 1.15, 1.2, 1.25, 1.3
7100 DATA 1.06, 1.12, 1.18, 1.24, 1.3, 1.36, 1.42
7200 DATA 1.12, 1.19, 1.26, 1.33, 1.4, 1.47, 1.54
  .    .
  .    .
  .    .
8700 DATA 1.87, 1.98, 2.09, 2.2, 2.31, 2.42, 2.53
9999 END
```

II. Personnel Matrix - PERMAT

You must type a tape of your personnel matrix using the following format. Type each step of your personnel matrix in one data statement.
NOTE: DATA STATEMENTS FOR THIS FILE MUST BEGIN AT STATEMENT NUMBER 5000 AND CONTINUE AT ANY INTERVAL TO 6900.

```
5000 DATA 10, 4, 2, 0, 3, 0, 0
5100 DATA  5, 6, 3, 1, 2, 0, 0
5200 DATA  6, 3, 6, 4, 7, 1, 0
  .   .
  .   .
  .   .
6700 DATA  4, 7, 12, 16, 8, 5, 3
```

PRELIMINARY ON-LINE PROCEDURE FOR RUNNING SALARY PROGRAMS

1. Call the H/P Time Sharing System.
2. When you hear a high-pitched tone, place phone in coupler.
3. Type in any number (0-9) and hit carriage return.
4. When computer asks you to log in, type in your appropriate district user ID, hit carriage return, and wait for READY message.
5. The first time you run SALSIM you must type in the following two statement.

   OPE-ML00005,5
   OPE-ML00017,7

   You need never type in these statements again. (In fact it would be best if you took a pen and crossed out this step after you have done it.)

6. If you have never run SALSIM, proceed to step 7, otherwise type in KIL-PROPS1 and hit carriage return.
7. Type in TAPE, hit carriage return and begin reading in the proposed schedule tape.
8. When the tape has been read in, type in NAM-PROPS1 and hit carriage return.
9. Type in SAV and hit carriage return.
10. Type in SCR and hit carriage return.
11. If you have never run SALSIM, proceed to step 12, otherwise type in KIL-PERMAT and hit carriage return.
12. Type in TAPE, hit carriage return, and begin reading in personnel matrix tape.
13. When tape has been read in, type in NAM-PERMAT and hit carriage return.
14. Type in SAV and hit carriage return.
15. Type in SCR and hit carriage return.

PROCEDURE FOR RUNNING - SALSIM

1. Type in GET-PERMAT and hit carriage return.
2. Type in APP-PROPS1 and hit carriage return.
3. Type in RUN and hit carriage return.

The computer will ask you to PLEASE TYPE IN THE NUMBER OF STEPS THEN THE NUMBER OF LANES SEPARATED BY A COMMA. When you have responded the computer will print DONE.

4. Type in GET-SALSIM and hit carriage return.
5. Type in RUN and hit carriage return.

The computer will now ask the following questions:

PLEASE TYPE IN THE NUMBER OF STEPS THEN THE NUMBER OF LANES SEPARATED BY A COMMA.

WHAT NUMBER DO YOU WANT TO MULTIPLY YOUR SCHEDULE BY? IF YOU DO NOT WANT TO MULTIPLY YOUR SCHEDULE TYPE IN A 1.

PLEASE ENTER YOUR LANE HEADINGS. MAXIMUM HEADING LENGTH IS 7 CHARACTERS. PLEASE TYPE ONLY ONE LANE HEADING AFTER EACH QUESTION MARK THAT FOLLOWS.
EXAMPLE:

```
? BA
? BA + 30
? MA
? MA + 30
? PH.D.
```

DO YOU WISH TO ADVANCE ALL TEACHERS ONE STEP FOR PROJECTED COST COMPUTATIONS? (TYPE 1 FOR NO, 2 FOR YES).

DO YOU WANT A PRINTOUT OF THE PERSONNEL MATRIX? (TYPE 1 FOR YES, 2 FOR NO).

When the program has finished running it will print DONE. If you want to run the program again type in RUN and hit carriage return. If you want to run one of the other programs go to the respective PROCEDURE FOR RUNNING section for that program. If you are done running programs type in BYE and hit carriage return.

RUN

SAMPLE OUTPUT - SALSIM

```
DO YOU WISH TO ADVANCE ALL TEACHERS ONE STEP FOR PROJECTED COST COMPUTATIONS?
(TYPE 1 FOR YES, 2 FOR NO)
?1

*****PERSONNEL MATRIX (TRAINING BY EXPERIENCE)*****

LANE 1      LANE 2      LANE 3      LANE 4      LANE 5
LANE 6      LANE 7      LANE 8      LANE 9

 0           60          8           5           0
 0           0           0           0           73

 0           57          12          3           1
 0           1"          0           0           74

 0           29          9           0           0
 0           0           0           0           38

 0           17          6           1           1
 1           1           0           0           27

 0           12          2           0           0
 3           1           0           1           19

 0           15          2           2           1
 1           1           1           0           23

 0           6           4           1           0
 0           0           0           0           11

 1           0           9           6           1
 2           1           0           1           21

 2           0           8           7           0
 2           0           0           0           19

 1           3           0           0           1
 1           0           1           0           7

 1           11          6           9           0
 5           7           7           4           50

 5           210         66          34          5
 15          12          9           6           362

THE AVERAGE STEP (ROW) ATTAINED BY YOUR CURRENT STAFF IS 4.73
```

SAMPLE OUTPUT - SALSIM

$$$

*****COST BY CATEGORY FOR THE PROPOSED SALARY SCHEDULE*****

LESS	BA	BA+15	BA+30	BA+45
$ 0	$ 0	$ 0	$ 0	$ 0
$ 0	$ 498000	$ 69600	$ 45500	$ 0
$ 0	$ 501600	$ 110400	$ 28800	$ 10000
$ 0	$ 269700	$ 87300	$ 0	$ 0
$ 0	$ 166600	$ 61200	$ 10600	$ 11000
$ 0	$ 123600	$ 21400	$ 0	$ 0
$ 0	$ 163500	$ 22600	$ 23400	$ 12100
$ 0	$ 69000	$ 47600	$ 12300	$ 0
$ 9900	$ 0	$ 112500	$ 77400	$ 13300
$ 21000	$ 0	$ 105600	$ 95200	$ 0
$ 22400	$ 189000	$ 83400	$ 128700	$ 14700
$ 53300	$ 1981000	$ 721600	$ 421900	$ 61100

$$$

SAMPLE OUTPUT - SALSIM

$$

*****COST BY CATEGORY FOR THE PROPOSED SALARY SCHEDULE(CONT)*****

MA	MA+15	MA+30	MA+45	TOTAL
$ 0	$ 0	$ 0	$ 0	$ 0
$ 0	$ 0	$ 0	$ 0	$ 613100
$ 0	$ 10800	$ 0	$ 0	$ 661600
$ 0	$ 0	$ 0	$ 0	$ 357000
$ 11400	$ 11800	$ 0	$ 0	$ 272600
$ 35700	$ 12300	$ 0	$ 13100	$206100
$ 12500	$ 12900	$ 13300	$ 0	$ 260300
$ 0	$ 0	$ 0	$ 0	$ 128900
$ 27600	$ 14200	$ 0	$ 15000	$ 269908
$ 29000	$ 0	$ 0	$ 0	$ 250800
$ 91200	$ 109200	$ 128000	$ 65600	$ 832200
$ 207400	$ 171200	$ 141300	$ 93700	$ 3852500

$$

SAMPLE OUTPUT - SALSIM

THE TOTAL COST FOR THE PROPOSED SALARY SCHEDULE IS $ 3852500

THE TOTAL COST FOR THIS SALARY SCHEDULE WOULD BE $ 2958600
IF ALL TEACHERS WERE PLACED ON STEP 1

THE COST OF THE EXPERIENCE INCREMENTS FOR THIS SCHEDULE IS $ 893900

THE TOTAL COST FOR THIS SALARY SCHEDULE WOULD BE $ 2746000
IF ALL TEACHERS WERE PLACED IN LANE 1

THE COST OF TRAINING INCREMENTS FOR THIS SCHEDULE IS $ 1106500

THE RATIO OF THE COST OF EXPERIENCE INCREMENTS TO THE COST OF
TRAINING INCREMENTS IS .81

THE AVERAGE SALARY PER TEACHER FOR THE PROPOSED SALARY SCHEDULE IS
$ 10642.3

LISTING

SALSIM

```
10  FILES ML0005,ML0017
11  PRINT "PLEASE TYPE IN THE NUMBER OF STEPS THEN THE NUMBER OF LANES"
12  PRINT "SEPARATED BY A COMMA."
13  INPUT A,B
20  C=A*B
25  FOR I=1 TO C
26  READ R
30  PRINT #1;R, END
35  NEXT I
50  FOR I=1 TO C
55  READ R
60  PRINT #2;R, END
65  NEXT I
69  PRINT
70  PRINT "WHAT NUMBER DO YOU WANT TO MULTIPLY YOUR SCHEDULE BY?"
71  PRINT "(SEE INSTRUCTIONS FOR DETAILS.)  IF YOU DO NOT WANT TO"
72  PRINT "MULTIPLY YOUR SCHEDULE, TYPE IN A  1 ."
73  INPUT Q9
90  DIM Z$[7]
110  DIM X[15,14],Y[16,15],Z[16,15],M[15],N[15]
120  DIM A$[7],B$[7],C$[7],D$[7],E$[7],F$[7],G$[7],H$[7],I$[7],J$[7]
130  DIM K$[7],L$[7],M$[7],N$[7],O$[7]
150  PRINT
160  PRINT "PLEASE ENTER YOUR LANE HEADINGS.  MAXIMUM HEADING "
170  PRINT "LENGTH IS  7  CHARACTERS.  PLEASE TYPE ONLY ONE "
180  PRINT "LANE HEADING AFTER EACH QUESTION MARK THAT FOLLOWS."
190  L=0
200  FOR O=1 TO B
210  INPUT Z$
220  L=L+1
230  GOTO L OF 260,280,300,320,340,360,380,400,420,440,455,465,475,485,495
240  IF L>B THEN 2358
245  NEXT O
250  GOTO 500
260  A$=Z$
270  GOTO 240
280  B$=Z$
290  GOTO 240
300  C$=Z$
310  GOTO 240
320  D$=Z$
330  GOTO 240
340  E$=Z$
350  GOTO 240
360  F$=Z$
370  GOTO 240
380  G$=Z$
390  GOTO 240
400  H$=Z$
410  GOTO 240
420  I$=Z$
430  GOTO 240
440  J$=Z$
450  GOTO 240
455  K$=Z$
460  GOTO 240
465  L$=Z$
470  GOTO 240
475  M$=Z$
480  GOTO 240
485  N$=Z$
490  GOTO 240
495  O$=Z$
499  GOTO 240
500  MAT X=ZER
502  PRINT
505  MAT Z=ZER
510  MAT Y=ZER
515  READ #1,1
520  FOR I=1 TO A
530  FOR J=1 TO B
540  READ #1;Y[I,J]
550  IF  END #1 THEN 590
560  NEXT J
570  NEXT I
580  GOTO 595
```

```
590   PRINT "STOP THE COMPUTER HAS NOT READ THE 'PROPAFILE PROPERLY'."
595   READ #2,1
600   FOR I=1 TO A
610   FOR J=1 TO B
620   READ #2;X[I,J]
625   X[I,J]=INT((X[I,J]*Q9)+.5)
630   IF  END #2 THEN 670
640   NEXT J
650   NEXT I
660   GOTO 680
670   PRINT "STOP THE COMPUTER HAS NOT READ THE 'PERMAT FILE PROPERLY'."
680   PRINT "DO YOU WISH TO ADVANCE ALL TEACHERS ONE STEP FOR"
690   PRINT "PROJECTED COST COMPUTATIONS? (TYPE 1 FOR YES, 2 FOR NO.)"
700   INPUT Q4
710   IF Q4=2 THEN 810
720   FOR I=2 TO A-1
730   FOR J=1 TO B
740   LET Z[I,J]=X[I,J]*Y[I-1,J]
750   NEXT J
760   NEXT I
770   FOR J=1 TO B
780   Z[A,J]=X[A,J]*(Y[A,J]+Y[A-1,J])
790   NEXT J
800   GOTO 860
810   FOR I=1 TO A
820   FOR J=1 TO B
830   LET Z[I,J]=X[I,J]*Y[I,J]
840   NEXT J
850   NEXT I
860   P1=P2=0
870   FOR I=1 TO A
880   FOR J=1 TO B
890   Y[I,B+1]=Y[I,B+1]+Y[I,J]
900   NEXT J
910   LET P2=Y[I,B+1]*I+P2
920   NEXT I
930   FOR J=1 TO B
940   FOR I=1 TO A
950   LET Y[A+1,J]=Y[A+1,J]+Y[I,J]
960   NEXT I
970   LET P1=P1+Y[A+1,J]
980   NEXT J
990   LET Y[A+1,B+1]=P1
995   PRINT
1000  PRINT "DO YOU WANT A PRINTOUT OF THE PERSONNEL MATRIX?"
1010  PRINT "(TYPE 1 FOR YES, 2 FOR NO.)"
1020  INPUT D3
1030  IF D3=2 THEN 1220
1040  PRINT
1045  PRINT
1050  PRINT "      *****PERSONNEL MATRIX (TRAINING BY EXPERIENCE)*****"
1060  PRINT
1070  FOR J=1 TO B
1080  PRINT "LANE"J,
1090  NEXT J
1100  PRINT
1110  PRINT
1120  FOR I=1 TO A+1
1130  FOR J=1 TO B+1
1140  PRINT Y[I,J],
1150  NEXT J
1160  PRINT
1170  PRINT
1180  NEXT I
1190  PRINT
1200  PRINT "THE AVERAGE STEP (ROW) ATTAINED BY YOUR CURRENT STAFF"
1210  PRINT "IS"INT((P2/P1)*100+.5)/100
1220  PRINT
1230  PRINT
2145  PRINT
2348  PRINT
2350  L=L+1
2351  Z$="TOTAL"
2352  GOTO 230
2355  PRINT
2358  PRINT
2360  PRINT "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
2370  PRINT
```

```
2380  PRINT
2390  PRINT
2400  PRINT "     *****COST BY CATEGORY FOR THE PROPOSED SALARY SCHEDULE*****"
2410  PRINT
2420  G=0
2430  FOR I=1 TO A
2440  FOR J=1 TO B
2450  LET Z[I,B+1]=Z[I,B+1]+Z[I,J]
2460  NEXT J
2470  LET G=Z[I,B+1]+G
2480  NEXT I
2490  FOR J=1 TO B
2500  FOR I=1 TO A
2510  LET Z[A+1,J]=Z[A+1,J]+Z[I,J]
2520  NEXT I
2530  NEXT J
2540  LET Z[A+1,B+1]=G
2580  PRINT
2590  PRINT
2600  PRINT "  "A$,"  "B$,"  "C$,"  "D$,"  "E$
2610  PRINT "  -------","  -------","  -------","  -------","  -------"
2615  PRINT
2620  PRINT
2630  FOR I=1 TO A+1
2640  FOR J=1 TO 5
2650  PRINT "$"Z[I,J],
2660  NEXT J
2670  PRINT
2680  PRINT
2690  NEXT I
2700  PRINT
2720  PRINT "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
2730  PRINT
2740  PRINT
2750  PRINT
2760  PRINT
2765  IF B <= 5 THEN 3380
2770  PRINT
2780  PRINT "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
2782  PRINT
2783  PRINT
2784  PRINT
2785  PRINT "*****COST BY CATEGORY FOR THE PROPOSED SALARY SCHEDULE (CONT)*****"
2786  PRINT
2787  PRINT
2788  PRINT
2790  PRINT "  "F$,"  "G$,"  "H$,"  "I$,"  "J$
2791  IF B <= 9 THEN 2800
2793  FOR J=6 TO 10
2795  PRINT "  -------",
2796  NEXT J
2798  GOTO 2808
2800  FOR J=6 TO B+1
2805  PRINT "  -------",
2806  NEXT J
2808  REM
2810  PRINT
2812  PRINT
2820  IF B >= 10 THEN 2900
2830  FOR I=1 TO A+1
2840  FOR J=6 TO B+1
2850  PRINT "$"Z[I,J],
2860  NEXT J
2870  IF B=9 THEN 2875
2872  PRINT
2875  PRINT
2877  PRINT
2880  NEXT I
2885  PRINT
2890  GOTO 3010
2900  FOR I=1 TO A+1
2910  FOR J=6 TO 10
2920  PRINT "$"Z[I,J],
2930  NEXT J
2940  PRINT
2950  PRINT
2980  NEXT I
2990  PRINT
```

```
3010 PRINT "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
3020 PRINT
3030 PRINT
3040 PRINT
3045 IF B <= 9 THEN 3380
3050 PRINT
3080 PRINT "$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
3090 PRINT
3100 PRINT
3110 PRINT
3120 PRINT "*****COST BY CATEGORY FOR THE PROPOSED SALARY SCHEDULE (CONT)*****"
3130 PRINT
3135 PRINT
3140 PRINT
3150 PRINT "  "K$,"  "L$,"  "M$,"  "N$,"  "O$
3160 FOR J=11 TO B+1
3170 PRINT "  -------",
3180 NEXT J
3185 IF B=14 THEN 3200
3190 PRINT
3200 PRINT
3210 PRINT
3220 FOR I=1 TO A+1
3230 FOR J=11 TO B+1
3240 PRINT "$"Z[I,J],
3250 NEXT J
3260 IF B=14 THEN 3280
3270 PRINT
3280 PRINT
3290 PRINT
3300 NEXT I
3310 PRINT
3330 PRINT "$$$$$$$$$D$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$"
3340 PRINT
3350 PRINT
3360 PRINT
3370 PRINT
3380 PRINT
3390 PRINT "THE TOTAL COST FOR THE PROPOSED SALARY SCHEDULE IS $"G
3400 PRINT
3410 PRINT
3420 G1=G2=0
3430 FOR J=1 TO B
3440 M[J]=Y[A+1,J]*X[1,J]
3450 LET G1=G1+M[J]
3460 NEXT J
3470 PRINT "THE TOTAL COST FOR THIS SALARY SCHEDULE WOULD BE $"G1
3480 PRINT "IF ALL TEACHERS WERE PLACED ON STEP 1"
3490 PRINT
3500 PRINT
3510 PRINT "THE COST OF THE EXPERIENCE INCREMENTS FOR THIS SCHEDULE IS $"G-G1
3520 FOR I=1 TO A
3530 N[I]=Y[I,B+1]*X[I,1]
3540 LET G2=G2+N[I]
3550 NEXT I
3560 PRINT
3570 PRINT
3580 PRINT "THE TOTAL COST FOR THIS SALARY SCHEDULE WOULD BE $"G2
3590 PRINT "IF ALL TEACHERS WERE PLACED IN LANE 1"
3600 PRINT
3610 PRINT
3620 PRINT "THE COST OF TRAINING INCREMENTS FOR THIS SCHEDULE IS $"G-G2
3630 PRINT
3640 PRINT
3650 PRINT "THE RATIO OF THE COST OF EXPERIENCE INCREMENTS TO THE"
3660 PRINT "COST OF TRAINING INCREMENTS IS"INT(((G-G1)/(G-G2))*100+.5)/100
3670 PRINT
3680 PRINT
3690 PRINT "THE AVERAGE SALARY PER TEACHER FOR THE PROPOSED"
3700 PRINT "SALARY SCHEDULE IS $"INT((G/P1)*100+.5)/100
3710 PRINT
3720 PRINT
3730 PRINT
3740 END
```

TITLE:	TOP MANAGEMENT DECISION GAME	DECSN A606-36065B
DESCRIPTION:	This program furnishes the simulated business conditions and the mechanics for operating a business game for any number from 10 to 60 participants. The participants form into teams representing ficticious companies and make decisions on price, promotion, production, capacity, research, incentives, and training in a one product market. The program provides a set of interrelated market and internal conditions that approximate real conditions, even including some random perturbation. The team decisions are converted into results fast enough so the results can be given back to the teams during the same class period, enabling the teams to make up to three sets of decisions during a two or three hour period. This quick feedback of results has been found to have excellent educational reinforcing characteristics.(See "ECONOMIC BACKGROUND" for further discussion.)	
INSTRUCTIONS:	See attached.	
SYSTEM SPECIFICATIONS:	2000A and Teletype.	
SPECIAL CONSIDERATIONS:	None	
ACKNOWLEDGEMENTS:	Professor Joseph Nordstrom Bowling Green University	

ECONOMIC BACKGROUND

This game gives the participant practice in making top level management decisions under time pressure. The decisions to be made call for attention to the inter-dependencies among the various decision areas, in other words, to the need for integrated policy thinking. Through the use of a Hewlett-Packard table top computer (Model 2114A with an 8K word memory) which can be brought right into the classroom, the results of participant decisions can be made available to them in a very short time after the decisions are made. In fact, during a two hour class period, the participants can play two or three periods of the game, getting their results back each period only minutes after turning in their decisions. Finally, the simulated business situation programmed into the computer is considerably more complicated than would be possible for a paper and pencil game with the same turn-around speed. All the calculations described below are performed as automatic functions of the computer program.

As is the case in the market place, some carry-over exists in this game from period to period. For example, the promotion contracted for in one period will also affect sales in the following periods. The same is true of expenditures for Research and Training.

This game, departing from the practices present in most similar games, makes the participant teams compute their own accounting statements. The computer printout does not supply these figures. The participants are thus forced to consider accounting relationships more than otherwise. Experience has shown that this feature is a valuable part of this game.

The participant should develop a sense of the market as he plays. He should attempt to "psych-out" the demand relationships as functions of his decisions. It is to his interest, therefore, to adopt somewhat more extreme strategies in this game than would be safe under real business conditions. In this way, he can learn without cost, and in a short time, lessons that might cost much more, both in time and money in the real market. It is worth pointing out especially here the role of strategy in business as opposed to decision. The participant will learn little from a policy of changing decision rationale frequently. He will learn much more from the results of having made a series of decisions according to a certain rationale that he wishes to test, i.e., a strategy. This is true in real life and is true in this game.

Market Demand

In general, demand is a function of price and promotion; the higher the promotion or the lower the price, the higher will be the demand.* The participant must be concerned not only with demand relations, however, but also with production costs at various levels of capacity. Inventory costs also must be considered. These factors can be controlled to some degree by attention to investment in training, incentive and research activities, but the final results will be dependent on all these elements acting together. Finally, there are forces acting on demand that are outside the control of the participants, i.e., the general market trends. The game starts off with a growth trend for a few periods to invite attention to necessary increases in capacity. Then there is a market decline for a few periods, inviting attention to inventory costs and overextended production capability. Finally, there is another rise in demand. The result affords an excellent opportunity for the participant to practice forecasting talents.

---

*With price the demand relationship is continuous. With promotion, however, the demand rises with promotion expense up to a maximum of $1,000,000 expense per period, whereupon the promotion effect saturates and no further increase in demand results from increased promotional outlay.

The market trend is given to the game by the following relationship:

$$F2 = 1 + .2P - .036P^2 + .0019P^3$$

where P is the number of the period being played.

The relation of demand to price and promotion is given by the following:

$$F1 = \frac{75}{P} \times \frac{300+S}{1500+S}$$

where P is the price and S is the promotion expense. In the case of total demand, mean price and promotion figures are used. In the case of team demand, the team's price and promotion figures are used.

For the total demand, the relation is:

$$D = N \times F1 \times F2 \times 60{,}000$$

where N is the number of teams. F1 gives the effects of price and promotion, and F2 gives the effect of the general market trend. The base demand is seen to be 60,000 units per team.

In addition to the factors mentioned above, there is a random perturbation of demand figures, so that a team's demand will not conform precisely to the functions noted above. This perturbation produces up to ten percent variation from the defined functions and can be thought of as the result of extraneous market conditions.

The Production Decision

The production cost is constant in any given period up to a production level which is 5000 units less than full capacity. Above this point, there is a per unit increase of 70 percent for production cost. For the participant this will result in gradually increasing average costs as he approaches and exceeds capacity. It might be noted that the participant may assume that he cannot produce above capacity. This is an erroneous assumption. Production in excess of capacity can be justified theoretically on the basis of creating a night shift, or farming some of the work out, etc.

The Capacity Decision

The capacity decision is made three periods in advance of the availability of the facilities contracted for. The facilities are not paid for until they are ready. The payment results in a reduction of the cash, but does not result in a commensurate reduction of profits in the period in which the facilities become available. The reduction in profits comes about through a steady state increase in administrative or overhead costs, so that, on a period by period basis, the cost is amortized. The amount of return on investment for money put into increased plant capacity will be favorable if this extra capacity is used, but it will just be extra expense if not used.

The Research Decision

Because it has been found advisable in this game for all teams to consider that they are selling the same product, research in product design is not appropriate. Therefore, it is assumed that research input is for the purpose of improving the process and that success in research will result in lower production costs. The research expenditures create a probability of breakthrough, the more expenditure the more

the probability. Only one breakthrough is possible in any one period and it lowers the production costs by 1.5 percent for every period from the point of breakthrough on. New breakthroughs increase the cost saving by the same factor. Obviously, the more production that is scheduled, the larger will be the resulting saving. The assignment of probabilities of breakthrough results from a random number simulation in the program. The characteristics of this probability function are such that a steady $80,000 per period investment in inventory yields the best return on investment.

The Incentive Decision

It is assumed that trying to apply a wage incentive to either the skilled or unskilled classifications will result in no improvement at all, due to the practical difficulties involved. However, applying a wage incentive program for the semi-skilled workers will result in some substantial improvement. The improvement will take the form of increase in apparent capacity, so that, when a team is producing at or above stated capacity, the production costs will be less, enough so that a satisfactory return on investment (in the incentive plan) occurs. It is assumed that unit production costs will not be affected, because the form of the incentive is such that labor will get as much per unit of production under incentive as before. Therefore, savings will result from a reduction in the costs of above capacity production, in the manner stated.

The Training Decision

The training decision assumes that there will be a lower production cost associated with increased training expense. The effect of this training input, however, will attenuate over time so that the effect will be far less two or three periods hence than it is directly after instituting the program.

In Summary

The carry over of effects (Research, Training, etc.) occurs thru the continual updating of the last (data statement) matrix in the printout, as mentioned in the Instructions for the Referee. It is important either to carry out this updating thru entering the new data by hand from the last matrix or by entering it by tape in the manner described.

It should be noted that this game was developed for 12 periods of play. If the instructor plays many more periods, it would be advisable to move from [P = 12] back to [P = 5, P = 6, etc.]. If this is not done, the instructor will find the market tendency rising at a rate without bound.

In general, there is no existing equity relationship in the case, because there is no fixed asset item nor is long term indebtedness or equity mentioned. Some instructors using the game may wish to add these figures, making it possible to develop balance sheet relationships and financial ratios which are not possible under present conditions. This will be easy to accomplish.

INSTRUCTIONS FOR THE REFEREE

The referee should first make sure that the Basic compiler is in the computer (Hewlett-Packard Model 2114A, 8K memory) and operative. He should then read in the game tape. After initializing the game program according to INSTRUCTIONS FOR INITIALIZING, the game will be ready to play.

The participants in the game should be divided into teams, ideally no fewer than three, nor more than seven participants in each team. It is best when there are at least three teams and the computer program will not handle more than eight teams. Each team should be encouraged to select a chairman (or president), an accountant, and appoint members to represent the marketing, personnel, and production functions.

The referee should then make sure that each team has an official team booklet, with copies of Forms I through IV arranged in a set for three years (four periods each year). The official set should have initial data (as per copy attached to this set of instructions) entered on the forms. This should include data regarding production, capacity, inventory value, administrative cost, and cash balance. Each member of each team should have a set of PLAYERS' INSTRUCTIONS, a copy of Chart I showing the past twenty-four periods of sales experience for his team, and copies of Forms I through IV that he can use for calculations. It is advisable to make this material available for study sometime before initiating play of the game.

The referee should then explain the philosophy of the game, pointing out that it is up to the teams to find out how the market reacts to their decisions and explaining the decisions to be made in the play for the first period on Form I. It is wise at this point to discuss the basic nature of the game, the fact that the teams are interdependent, the fact that market response will be dependent on the team decisions, to some degree, but that the market response will also be determined by general economic trends and by some random variation. Also, it can be pointed out that one quarter's decisions will affect results not only for that quarter, but for future quarters as well.

Receiving the Decisions

Upon receiving the booklets (with decisions) from the teams, the referee should verify the calculations and the entries, making sure that enough lead time is given for decisions regarding new production and capability. He should check profit calculations and should assure himself that each team has entered the cost of negative cash balance, if the team incurred some.

Entering Data

The referee then types the decisions into the computer program as data. (See INSTRUCTIONS FOR ENTERING DATA.) Decisions as to price, promotiom, production, capacity, research, and incentives are typed into memory locations from 351 to 358 (as needed), team one's decisions being typed into 351, team two's into 352, etc. In the event that some team other than team one is the first to turn in its decisions, that team's decisions can be entered as soon as the form arrives. For example, if team four is the first to turn in its decisions, the decisions can be entered into memory location 354. Decisions as to training are entered into memory locations 361 through 368 (as needed) in the same way that data was entered in the 351-358 block. The referee should check the means for entering training data carefully before proceeding. It should be noted that the data for incentive wages and for training must be coded before entry.

For incentive wages, the questions involve only whether the team has paid the full cost of the incentive plan and whether the plan is for the semi-skilled workers or not. A plan for another group of workers produces no effect at all. A plan, fully paid for, for semi-skilled workers, increases plant capacity, thereby causing less production expense when the plant is working near or above capacity. In coding the plan, the number entered should be 30 or more if the plan is for semi-skilled workers and less than 30 if the plan is for one of the other two classes of employees, or if there is no plan.

As for training, the number entered is a function of the number of periods since a training program was installed. In the first period, whether or not a training program was installed, this number will be (0). This is because the results of the training program are not apparent in the period for which it is installed. In the next period, if a training program was installed in the first period, the number should be (6). One period after the introduction of an additional new training program, six should be added to the number which appears as the last item of data in subject team's line in the last matrix printed out from the previous period (the matrix characterized as data statements). The data for the present period should be corrected by this increase. Note that, in entering such data, the whole data line must be entered, even though most of the data is as it was. For example, if team 4 had initiated a training program last period, and this period's data line had been "364 DATA 70,1,12,525,80,3.235", it should be re-entered as "364 DATA 70,1,12,525,80,9.235".

Running the Program

First, the "ON" button for the tape punch should be pressed and the "HERE IS" button should also be pressed. This will avoid any residual punching on the tape. Now, press the "OFF" button on the tape punch. At this point, after making sure that the program is initialized and all new data is fed in, type "RUN" and press the carriage return button. This should cause three matrices to be printed out along with two pieces of summary information. At the end of the second piece of summary information, "Total Promotion, $XXX", there will be a pause of one second. During this second, press the "ON" button of the tape punch. All the new data for data block 361 through 36n will be typed out and punched onto the tape (n depending on the number of teams). At the end of this series of data statements there will be another pause of one second. At this point press the "OFF" button. This will insure that only the data statements are present on the tape. The computer will then type "READY". After this has happened, press the "ON" button again and press "HERE IS". This will give you some blank tape at the end of the data. Then press the "OFF" button. Immediately then put this piece of tape in the tape reader and read in the new data. The computer is now initialized for the next period of play. The first matrix printed out will simply show the team decisions for the referee's verification.

The referee should then insert the "sales" figures (in units), the "Prod Cost" figures (in dollars), and the "Admin Cost" figures (in dollars) from the second matrix in the appropriate places on the team forms. The last column, "unit cost", will be useful in the final game analysis. He can then hand the official booklets back to the teams. During the second play of the game, the teams will need assistance in completing the forms especially the Income Statement. It is probably appropriate to explain the forms to the participants as a group, going down through the necessary calculations.

Summarizing Results

After each period is decided, the referee should post on the blackboard: 1) the prices charged by each team last period; 2) a combined sales total for all teams; 3) the total amount charged by all teams for promotion. After each four periods post the year's profit for each team. Each period represents three months.

At the end of each four periods (i.e., each year) the referee will calculate a total profit for each team for the year. He will charge the team an income tax payable in the following period of play. This income tax will be 50% of the total profit calculated. This figure will be entered on Form II on line 15, labeled "Tax". The team must treat this as an expense in said period.

Upon completing the game for any one day's play, the referee should retain all official team booklets, allowing the participants to keep their calculation sheets. He should also read out the present stage of the game onto tape. This he does by first typing "PLIST" on the teletype, waiting a second and then pressing the "ON" button for the tape punch. The tape will be furnished with blank leader and follower in the process and can be used to initiate play of the game for the next period. Only the new decision data and the new period number will have to be furnished.

Game Analysis

Upon completion of all the plays of the game, the referee should post records of the performances of all teams. A suitable form for such a presentation is embodied in Chart II with a series of trend lines for each team depicting various criteria useful in analyzing the game experience. The teams should be encouraged to analyze their own experiences for the benefit of the other teams in the analysis session. The referee can comment as he feels appropriate. The data for the presentation can be found in the official team booklets and in the printouts from the computer.

Note: (1) It may be useful at times to experiment with the game in order to determine how the total demand function behaves. For such purpose, the experimenter may wish to print out only a selected portion of the total printout. He can eliminate printing Matrix A for example, simply by one instruction, "9 GO TO 17". Similarly he can eliminate printing the second matrix by typing "285 GO TO 330". The last (data statement) matrix can be omitted by typing "379 GO TO 400". When these matrices are again desired, simply type "9" then return, "285" then return, and "379" then return.

Note: (2) A copy of the referee's data sheet is included at the conclusion. It is useful to enter team decision data on this sheet before entering the data into the computer. In this way errors in entering data can be avoided. Further, information on this sheet will be useful in the final game analysis.

INSTRUCTIONS FOR INITIALIZING GAME

1. The letter "N" represents the number of teams in the game. In order to set up the program for a given play of the game, this number will have to be inserted in the following manner:

   Type "4 LET N = (the number of teams)"

   For example, if the number of teams is to be 5, the instruction is:

   "4 LET N = 5"

2. Next, the period of play must be inserted. Assuming that this is the first period, this is done as follows:

   Type "5 LET P = 1"

   For the third period of play it would be:

   "5 LET P = 3"

   This instruction will give a market trend to the demand function throughout the game. As originally set up, this trend will call for rising total demand (all other things being equal) during the initial four periods. The demand will then level off and drop until the ninth period. The demand will then level off and rise again.*

3. The tape is set up initially for eight teams. For this reason, data will have to be omitted for any teams above the actual number playing. This will have to be done in two different data blocks, 351 to 358 and 361 to 368. The omissions should be from the higher numbers in each case to the lower numbers. For example, if the actual number of teams is 5, simply type the following numbers, pressing "return" after each number:

   356, 357, 358, 366, 367, 368

4. The data representing cumulative effects of past decisions are already entered in the initial tape. These entries won't have to be changed for the first period's play.

   The game is now ready for the first period.

---

*If the instructor contemplates playing many more than twelve periods, it would be wise to move from period 12 to period 5 and then to period 6, etc. Using period numbers much higher than 12 will introduce demands that will probably be too high for practical purposes.

PLAYER INSTRUCTIONS

(To be given to all players)

You are a member of a closely knit management team that is competing directly with several companies for a share of an industrial market. All of the companies are selling a product that is technically similar. Price and promotional effort are the key elements affecting volume. Profits result from a careful assessment of market demand, competitor's activities, and sound production and expense planning and control.

As in any business, a number of forms must be used to communicate your decisions and to report the company's position. Each period you must determine the: (1) product price, (2) promotion expenditure, (3) amount to be spent on plant expansion, (4) volume of raw material to be placed into production, (5) amount of research investment, (6) amount of expenditure for an incentive program, and (7) amount of investment in a training program. These decisions will be fed into a computer simulation representing a real market situation, and the results will be given back by the referee. Your team's results will be determined by (1) your decisions, (2) your competitors' decisions, and (3) the market conditions (affected by some trend indices). Additionally, there will be some random variation.

Selling Price (Line 1)

All other things being equal, the higher your promotion outlay, the more units of your product the market will absorb and the lower the price the more units of your product the market will absorb. However, the number of units sold by your firm will depend on the price and promotion outlays set by your firm in relationship to competitor's actions. There will be cyclical, seasonal and random influence in total market demand, as well as influence from the average price charged by all competitors and the total amount they spend on promotion. Orders must be filled from currently available merchandise; and inventory deficiency results in lost sales.

Selling prices can only be changed in one dollar increments with a maximum variation of two dollars per unit from one period to the next. Assume that the last price charged was $30.00 per unit and your company sold 59,000 units.

Over the past several years each competitor has maintained an equal share of the market. Chart I portrays your company's sales volume over the last twenty-four periods. Your marketing research staff has reported that they expect the upward trend to continue.

Promotion Budget (Line 2)

This is the amount spent for advertising and personal sales effort. The budgeted amount cannot be altered more than $100,000 from one period to the next, and changes made in $50,000 steps. Your promotion budget last period was $450,000. The effect of promotional effort is somewhat cumulative. That is, there will be some effect on sales in later periods due to this period's promotion.

Production (Line 3)

During any period you may begin the ordering and production cycle for any number of raw materials units. The complete cycle requires two periods: one period for the raw materials to arrive after they have been ordered and one period to change the inputs into finished goods. Therefore, if a stock of finished inventory is needed for sale during period five for example, the raw material order must be placed no later than the beginning of the third period. Therefore, the production decision must be made for the third period.

Two periods ago, 60,000 units of raw material were ordered. They can be sold during this first game period. Last period, 65,000 units were ordered and will be saleable during period II. If you wish to have additional inventory available for sale during the third period, enter the desired quantity now on line three, period III.

There is a 10% inventory carrying charge each period. This charge is based on cost value of the ending inventory. Your ending inventory last period was 41,000 units, valued at $310,000. Thus the carrying charge would have been $31,000 last period.

Your manufacturing cost is about $10 per unit when production is near plant capacity. Above capacity production leads to overtime rates and other charges; if your plant operates much under its full potential, the $700,000 fixed charges will raise the unit costs. However, this should not be construed as forbidding you to produce above capacity.

Plant Capacity Additions (Lines 4, 5, and 6)

Initially, your plant has a 75,000 unit per period capacity. Every $60,000 spent for expansion will increase the plant's capacity 1,000 units. An expansion program initiated during one period is not completed until three periods have passed. Payments are not made for plant additions until the new space is available for use.

Your plant will have a 77,000 unit capacity during Period III. If you believe more capacity will be needed during the fourth period, this expansion program must be started now in the first period. Enter this decision on lines 7, 8, and 9 in the space provided under Period IV.

Research and Development (Line 7)

An investment may be made for research and development during any period. The more money that is put into research, the greater the probability of a breakthrough. For any investment made there is a period of delay due to the time needed for research before any results are realized. If a breakthrough does occur, the advantages will be realized through a reduction in total production cost for each period after the research investment repays itself. Repetitive breakthroughs are possible if research investments are repeated. The same total amount invested over time as a steady state input will give a greater possibility of breakthrough than if it is invested all in one period. That is, crash research programs, while effective, are more expensive than regular research investment. Investments in research must be made in multiples of $20,000. There is an investment maximum of $160,000 per period.

Incentive Wage Program (Line 8)

An incentive program may be installed for all levels of the production force: unskilled, semi-skilled, and skilled. Any one, all, or a combination of these segments may be put on incentive during any period. The costs of the program include an initial cost for determining each job's productivity measurement, for establishing evaluation methods, and for making accounting adjustments. This cost is $50,000 for any or all groups of workers able to be put on incentive. There will also be a steady-state cost of $10,000 per period for each skill level on incentive. This is needed to maintain the control, evaluation, and accounting procedures. The advantage of this program's establishment is that it may substitute for additions to the plant capacity. The increased production advantage of the program discontinues if the payments cease. As 60% of the work force is semi-skilled, the benefits of this group being put on incentive would be evidenced soonest -- during the period in which introduced. If this program is introduced, enter the amount of incentive expenditure on Line 4 of Form I. The total unit production, including increments added by incentives, will be taken into account when the computer calculates the production cost.

Training Program (Line 9)

It has been determined that the introduction and use of an extensive training program for production workers will result in lower total production costs whether production is at full capacity or not. If it is decided to begin this program, the expense will be one investment of $30,000. This expense will include the cost of instructors and educational material. It will take a period before the details of the program's setup are complete and the instructors are trained. Then there will be a reduction in production costs. The amount of total production cost reduction will exist from time to first effect, in decreasing amount from period to period.

Negative Cash Balance

At the end of every period in which your cash balance is negative, you will be charged an extra 5% of the amount by which it is negative. Make your calculations on scratch paper first to see if you will have a negative cash balance. Then add this cost if so. This is the cost of borrowing money to cover debts.

Income Tax

At the end of each four periods (1 year) the referee will calculate an income tax to be paid in the following period. It will be entered in Form II, line 15 and also in the space provided.

Completing the Income Statement

Step 1: The unit sales will be entered in Form II, line one, by the judge. Multiply the unit sales figure by the price charges by the company this period. Enter the dollar sales volume on line two.

Step 2: Line three, Beginning Inventory, is the same as line six, Ending Inventory, from the previous period.

Step 3: Line four, Production Cost, is entered on Form II by the judge.

Step 4: Line five, Merchandise Available for Sale, is the sum of lines three and four.

Step 5: Multiply the Unit Sales, line one, by the average unit cost (Form IV, line six) and enter the product on line seven as the Cost of Goods Sold. Form IV is provided as a worksheet to aid in calculating the number of units of ending inventory and also the average unit cost.

Step 6: Subtract line 7, Cost of Goods Sold, from line 5, Merchandise Available for Sale, and enter on line 6, Ending Inventory.

Step 7: Subtract line 7, Cost of Goods Sold, from line 2, Sales, and enter the difference on line 8, Gross Margin.

Step 8: Enter the Promotion Expense on line 9, from Form I, line 2.

Step 9: Enter the Research Expense on line 12 from Form I, line 7.

Step 10: Enter the Incentive Cost on line 10, from Form I, line 8.

Step 11: Enter the Training Expense on line 11 from Form I, line 9.

Step 12: Inventory Carrying Charge, line 13, is 10% of line 6, Ending Inventory.

Step 13: The Overhead is provided by the judge. It is a function of capacity.

Step 14: Add lines 9 through 15 and subtract the total from line 8. Enter the difference on line 16.

Negative Cash Balance

Step 1: Complete the Cash Available Statement - Form III. The "cash end this period" is the result of subtracting the sum of lines 4 and 5 from the sum of lines 1, 2, and 3.

Step 2: If there is a negative cash balance at the end of the period, enter 5% of that figure as a penalty on the Income Statement, Form II, line 15. Reduce the Net Income (or increase the Net Loss) for the company for every period that there is a negative cash balance on Form III.

Average Unit Cost (Form IV, Line 6)

Calculate the average unit cost by dividing the value of total merchandise for sale (Form II, line 5) by the number of units available for sale (Form IV, line 30). This figure should be entered on line 6 of Form IV.

JUDGE'S FORM

Class____________________ Date____________________

Number of Teams____________________ Judge____________________

Location	Designation	Team #	Price	Prom.	Prod.	Cap.	Research	Incentive
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							

JUDGE'S FORM

Class Management 460 (B) Date 2/18/70

Number of Teams 5 Judge Nordstrom

Location	Designation	Team #	Price	Prom.	Prod.	Cap.	Research	Incentive
351	DATA	1	30	500	65	75	100	30
352	DATA	2	30	850	100	80	160	20
353	DATA	3	31	500	55	75	30	10
354	DATA	4	29	650	90	80	10	0
355	DATA	5	26	500	65	75	80	30
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							
	DATA							

FORM I MANAGEMENT DECISIONS

Company_____ Year_____

	I	II	III	IV
1. Selling Price				
2. Promotion Budget				
3. Production, Units				
4. Plant Cap. Add'ns, Units				
5. Plant Cap. Add'ns, $				
6. Cum. Cap. Add'ns, $				
7. Research Inv't, $				
8. Incentive Prog. Exp., $				
9. Training Prog. Inv't, $				

FORM II INCOME STATEMENT

1. Sales, Units				
2. Sales, $				
3. Begin, Inv'y, $				
4. Production Cost, $				
5. Mdse. Av. for Sale, $				Year's Profit
6. Ending Inv'y, $				
7. Cost of Goods Sold, $				
8. Gross Margin, $				Income Tax
9. Promotion Exp., $				
10. Incentive Cost, $				
11. Training Exp., $				Net After Tax
12. Research Exp., $				
13. Inv'y Carrying Charge, $				
14. Overhead, $				
15. Cash Shortage Charge, $				
16. Net Income (loss), $				

FORM I MANAGEMENT DECISIONS

Company X Year 197X

	I	II	III	IV
1. Selling Price				
2. Promotion Budget				
3. Production, Units	60,000	65,000		
4. Plant Cap. Add'ns, Units			2,000	
5. Plant Cap. Add'ns, $			120,000	
6. Cum. Cap. Add'ns, $	75,000	75,000	77,000	
7. Research Inv't, $				
8. Incentive Prog. Exp., $				
9. Training Prog. Inv't, $				

FORM II INCOME STATEMENT

1. Sales, Units					
2. Sales, $					
3. Begin. Inv'y, $	310,000				
4. Production Cost, $					
5. Mdse. Av. for Sale, $					Year's Profit
6. Ending Inv'y, $					
7. Cost of Goods Sold, $					
8. Gross Margin, $					Income Tax
9. Promotion Exp., $					
10. Incentive Cost, $					
11. Training Exp., $					Net After Tax
12. Research Exp., $					
13. Inv'y Carrying Charge, $					
14. Overhead, $	700,000				
15. Cash Shortage Charge, $					
16. Net Income (loss), $					

FORM III CASH AVAILABLE

	I	II	III	IV
1. Cash, End of Last Per.				
2. Inv'y End Last Period				
3. Net Income This Period				
4. Paid for Add'l Plant Cap.				
5. Inv'y End This Period				
6. Cash End This Period				

FORM IV INVENTORY WORKSHEET

1. Beginning Inv'y				
2. Units This Period				
3. Total Units for Sale				
4. Unit Sales, This Per.				
5. Ending Inv'y, Units				
6. Av. Unit Cost (II5/IV3)				

FORM III CASH AVAILABLE

	I	II	III	IV
1. Cash, End of Last Per.	660,000			
2. Inv'y End Last Period	310,000			
3. Net Income This Period				
4. Paid for Add'l Plant Cap.			120,000	
5. Inv'y End This Period				
6. Cash End This Period				

FORM IV INVENTORY WORKSHEET

1. Beginning Inv'y	31,000			
2. Units This Period	60,000			
3. Total Units for Sale	91,000			
4. Unit Sales, This Per.				
5. Ending Inv'y, Units				
6. Av. Unit Cost (II5/IV3)				

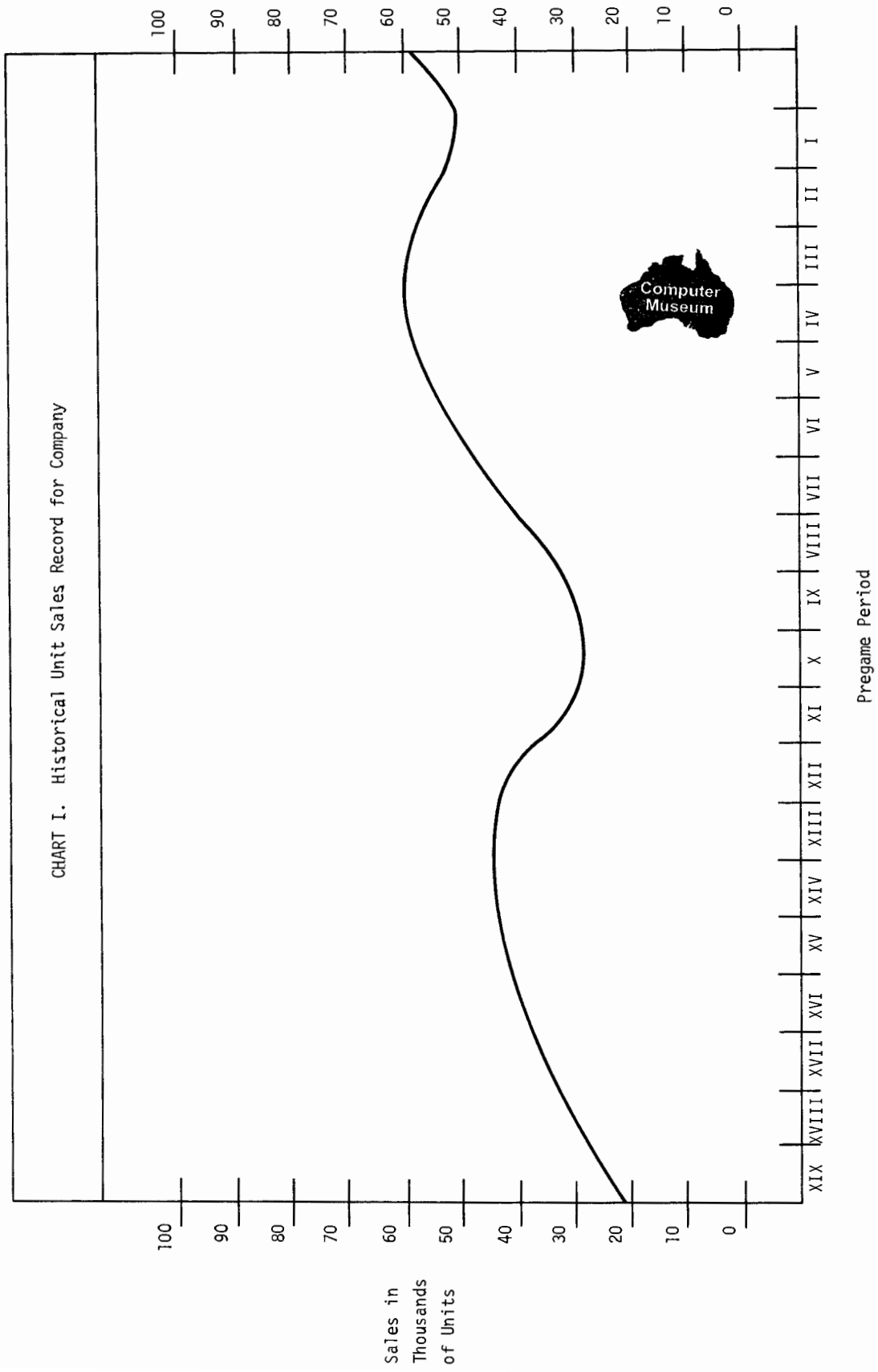
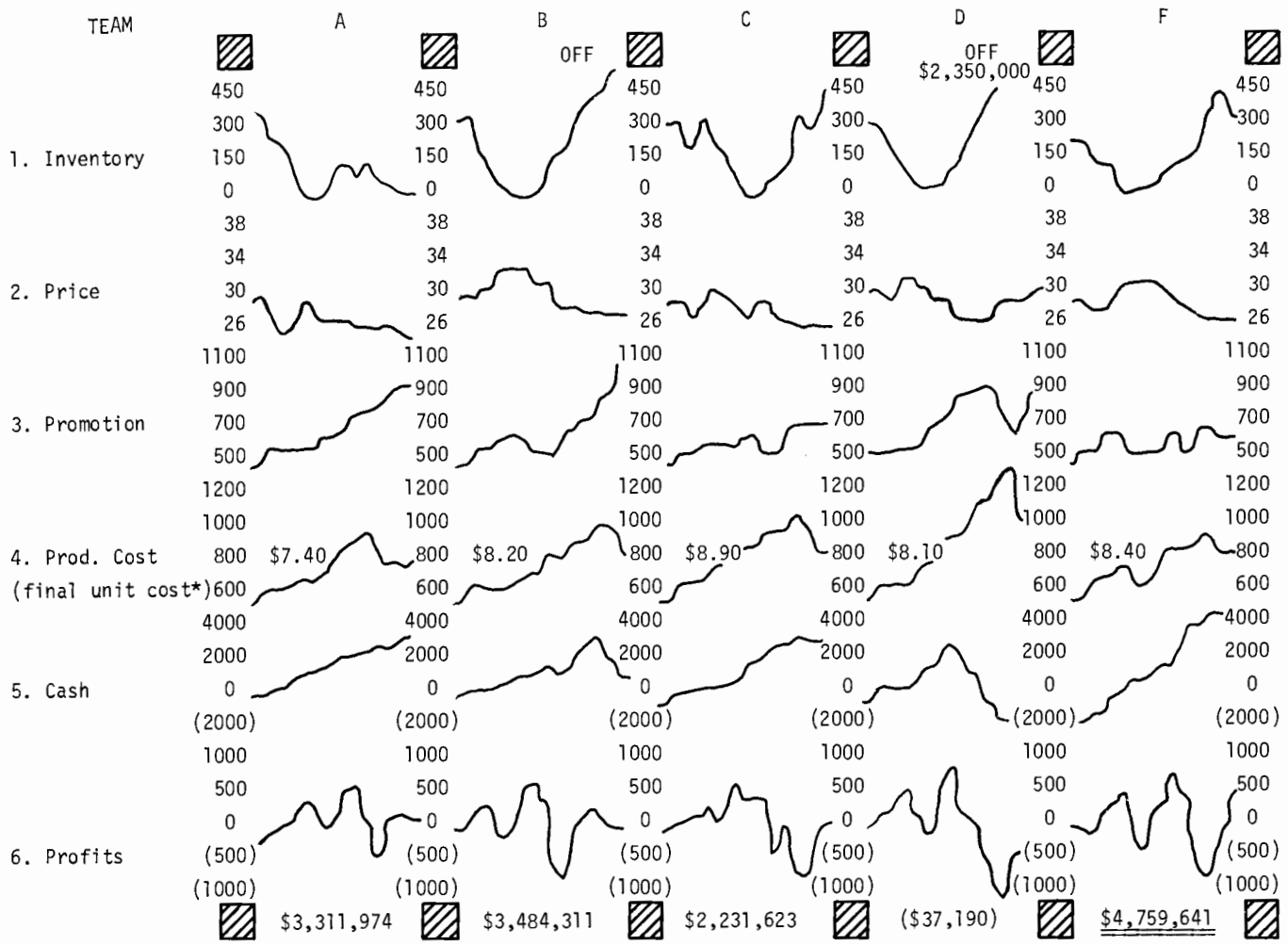


CHART I. Historical Unit Sales Record for Company
Computer Museum
Sales in Thousands of Units
0
10
20
30
40
50
60
70
80
90
100
XIX
XVIII
XVII
XVI
XV
XIV
XIII
XII
XI
X
IX
VIII
VII
VI
V
IV
III
II
I
Pregame Period

CHART II FINAL ANALYSIS SHEET



CONTRIBUTED PROGRAM **BASIC**

KR20

TITLE: ITEM ANALYSIS AND KUDER-RICHARDSON FORMULA 20 RELIABILITY A408-36137A

DESCRIPTION:

This program may be used to do an item analysis on teacher-constructed tests to determine the difficulty, discrimination index, and PQ value for each item, and the average difficulty, average discrimination index, and Kuder-Richardson Formula 20 Reliability for the test.

INSTRUCTIONS:

After determining the number of students in the upper 27% and the number in the lower 27% of all the students who took the test, the teacher tabulates the number of correct responses to each item on the test for each of these two groups.

DATA: line 350: number of items on the test, number of people in either the high or low group (27% of all those taking the test).

in following data lines, list the number of correct responses for the high group on item #1, no. of correct responses for the low group on item no. 1; then correct responses for the high group on item no. 2, no. of correct responses for the low group on item no. 2, etc.

last data line (line 400) must be the variance (standard deviation squared) for the test obtained previously using all test scores.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

NONE

ACKNOWLEDGEMENTS:

Donald E. Gettinger
Stillwater Senior High School

RUN

```
RUN
KR20

TEST ITEM       HIGH  LOW   DIFFICULTY        DISCR. INDEX        PQ
 1               44    27    .622807           .298246          .234918
 2               52    42    .824561           .175439          .14466
 3               50    11    .535088           .684211          .248769
 4               49    32    .710526           .298246          .205679
 5               18    2     .175439           .280702          .14466
 6               22    12    .298246           .175439          .209295
 7               56    26    .719298           .526316          .201908
 8               56    29    .745614           .473684          .189674
 9               54    32    .754386           .385965          .185288
 10              56    29    .745614           .473684          .189674
 11              41    13    .473684           .491228          .249307
 12              54    37    .798246           .298246          .16105
 13              57    47    .912281           .175439          8.00246E-02
 14              57    36    .815789           .368421          .150277
 15              55    35    .789474           .350877          .166205
 16              55    48    .903509           .122807          8.71807E-02
 17              51    27    .684211           .421053          .216066
 18              52    15    .587719           .649123          .242305
 19              50    18    .596491           .561404          .240689
 20              15    8     .201754           .122807          .16105
 21              57    52    .95614            8.77193E-02      .041936
 22              53    31    .736842           .385965          .193906
 23              55    40    .833333           .263158          .138889
 24              56    21    .675439           .614035          .219221
 25              55    21    .666667           .596491          .222222
 26              47    14    .535088           .578947          .248769
 27              54    9     .552632           .789474          .24723
 28              45    18    .552632           .473684          .24723
 29              27    11    .333333           .280702          .222222
 30              55    10    .570175           .789474          .245075
 31              48    16    .561404           .561404          .24623
 32              51    22    .640351           .508772          .230302
 33              19    14    .289474           8.77193E-02      .205679
 34              22    10    .280702           .210526          .201908

SUM OF PQ= 6.6195              VARIANCE= 29.963
AVERAGE DIFFICULTY IS  .619969
AVERAGE DISCRIMINATION INDEX IS  .398865
KUDER-RICHARDSON FORMULA 20 RELIABILITY= .802686

DONE
```

LISTING

```
KR20
10  REM ***  HP TIME-SHARED BASIC PROGRAM LIBRARY  **********************
11  REM
12  REM          KR20:  ITEM ANALYSIS AND KUDER-RICHARDSON FORMULA 20
13  REM                 RELIABILITY
14  REM          36137 (A408) REV A -- 7/71
15  REM
16  REM ***  CONTRIBUTED PROGRAM  ***************************************
17  REM
100  REM ITEM ANALYSIS
110  REM K=NO. ITEMS, N=NO. PEOPLE IN EITHER GROUP
120  READ K,N
130  LET S=0
140  PRINT "TEST ITEM","HIGH  LOW   DIFFICULTY       DISCR. INDEX",
150  PRINT "    PQ"
160  LET D1=0
170  LET D2=0
180  FOR I=1 TO K
190  READ H,L
200  PRINT I,H;L;(H+L)/(2*N),(H-L)/N,
210  LET D1=D1+(H+L)/(2*N)
220  LET D2=D2+(H-L)/N
230  LET P=((H+L)/(2*N))*((2*N-H-L)/(2*N))
240  PRINT P
```

```
250  LET S=S+P
260  NEXT I
270  REM V=VARIANCE      (S.D.↑2)
280  READ V
290  LET R=(K/(K-1))*(1-S/V)
300  PRINT
310  PRINT "SUM OF PQ="S,"VARIANCE="V
320  PRINT "AVERAGE DIFFICULTY IS "D1/K
330  PRINT "AVERAGE DISCRIMINATION INDEX IS "D2/K
340  PRINT "KUDER-RICHARDSON FORMULA 20 RELIABILITY="R
350  DATA 34,57
360  DATA 44,27,52,42,50,11,49,32,18,2,22,12,56,26,56,29,54,32
370  DATA 56,29,41,13,54,37,57,47,57,36,55,35,55,48,51,27,52,15
380  DATA 50,18,15,8,57,52,53,31,55,40,56,21,55,21,47,14,54,9
390  DATA 45,18,27,11,55,10,48,16,51,22,19,14,22,10
400  DATA 29.963
410  END
```

SCORES
A408-36136A

TITLE: COMPUTES MEAN, STANDARD DEVIATION, AND STANDARD SCORES FOR TEST SCORES

DESCRIPTION:

Program finds the mean and standard deviation for a set of scores, and the deviation, Z-score, and T-score for each of the individual scores.

INSTRUCTIONS:

DATA: First line (line 370) is number of scores.

List the scores on the following data line(s).

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS:

Program assumes a normal distribution of scores.

ACKNOWLEDGEMENTS: Donald E. Gettinger
Stillwater Senior High School

RUN

```
RUN
SCORES

MEAN =  41.3333
STANDARD DEVIATION =  7.66522

SCORE          DEVIATION      Z-SCORE        T-SCORE
-----          ---------      -------        -------
 50             8.66666        1.13065        61.3065
 50             8.66666        1.13065        61.3065
 50             8.66666        1.13065        61.3065
 48             6.66666        .869729        58.6973
 48             6.66666        .869729        58.6973
 44             2.66666        .347891        53.4789
 43             1.66666        .217432        52.1743
 42             .666664        8.69726E-02    50.8697
 42             .666664        8.69726E-02    50.8697
 42             .666664        8.69726E-02    50.8697
 41            -.333336       -4.34868E-02    49.5651
 35            -6.33334        -.826243       41.7376
 30            -11.3333        -1.47854       35.2146
 29            -12.3333        -1.609         33.91
 26            -15.3333        -2.00038       29.9962

DONE
```

LISTING

```
SCORES
10  REM ***  HP TIME-SHARED BASIC PROGRAM LIBRARY  **********************
11  REM
12  REM          SCORES:  COMPUTES MEAN, STANDARD DEVIATION AND STANDARD
13  REM                   SCORES FOR TEST SCORES
14  REM          36136 (A408) REV A -- 7/71
15  REM
16  REM ***  CONTRIBUTED PROGRAM  ***************************************
17  REM
100  REM STATISTICS PROGRAMS: MEAN, STANDARD DEVIATION, STANDARD SCORES
110  DIM X[100],D[100],Z[100],T[100]
120  LET X=0
130  READ N
140  FOR I=1 TO N
150  READ X[I]
160  LET X=X+X[I]
170  NEXT I
180  LET M=X/N
190  PRINT "MEAN = ";M
200  LET D=0
210  FOR I=1 TO N
220  LET D[I]=X[I]-M
230  LET D=D+D[I]↑2
240  NEXT I
250  LET S=SQR(D/N)
260  PRINT "STANDARD DEVIATION = ";S
270  FOR I=1 TO N
280  LET Z[I]=D[I]/S
290  LET T[I]=10*D[I]/S+50
300  NEXT I
310  PRINT
320  PRINT "SCORE","DEVIATION","Z-SCORE","T-SCORE"
330  PRINT "-----","---------","-------","-------"
340  FOR I=1 TO N
350  PRINT X[I],D[I],Z[I],T[I]
360  NEXT I
370  DATA 15
380  DATA 50,50,50,48,48,44,43,42,42,42,41,35,30,29,26
390  END
```

INTGRT

TITLE: COMPUTES NUMERIC INTEGRAL OF A FUNCTION — A310-36021A

DESCRIPTION:

This program uses Simpson's Rule for finding the numeric integral of a function.

INSTRUCTIONS:

The function should be defined by a

```
DEF FNF(X)=...
```

statement with a line number less than 9000. The following data should be supplied in line 9900:

```
A = lower X bound
B = upper X bound
E = acceptable error
```

To use this routine as a subroutine, change line 9017 to RETURN, delete line 9003. Now define the function and data in the main program. To stop printout, delete line 9016. The value of the integral is given by the variable S.

Variables used: A,B,E,H,M,S,T,X

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

None

ACKNOWLEDGEMENTS:

```
RUN

GET-$INTGRT
8900 DEF FNF(X)=SIN(X)
9900 DATA 0,3.142-159,.001
RUN
INTGRT

INTEGRAL =  2.

DONE


LISTING

INTGRT

8900  DEF FNF(X)=SIN(X)
9000  REM  *****  INTGRT  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  COMPUTES NUMERIC INTEGRAL OF A FUNCTION
9003  READ A,B,E
9004  LET H=B-A
9005  LET T=(FNF(A)+FNF(B))*H
9006  LET M=0
9007  LET T=(T+M)/2
9008  LET M=0
9009  FOR X=A+H/2 TO B STEP H
9010  LET M=M+FNF(X)
9011  NEXT X
9012  LET M=M*H
9013  LET S=(T+2*M)/3
9014  LET H=H/2
9015  IF ABS(T-M)/ABS(S)>E THEN 9007
9016  PRINT "INTEGRAL = "S
9017  STOP
9900  DATA 0,3.14159,.001
9999  END
```

CONTRIBUTED PROGRAM **BASIC**

EUCLID
A301-36035B

TITLE: COMPUTES LARGEST COMMON FACTOR OF TWO INTEGERS

DESCRIPTION: This program finds the largest common factor of two integer numbers using the Euclidean Algorithm.

INSTRUCTIONS: The program will request the two integers during execution and print out the result.

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS: None

ACKNOWLEDGEMENTS: Richard Klein
Los Angeles City Unified School District

RUN

```
RUN
EUCLID


WHAT ARE YOUR TWO INTEGERS?48,32
LARGEST COMMON FACTOR IS  16

DONE
```

LISTING

```
EUCLID
9000  REM  ***  HP TIME-SHARED BASIC PROGRAM LIBRARY  ********************
9001  REM
9002  REM          EUCLID:  36035  (A321)  REV B -- 9/71
9003  REM          COMPUTES LARGEST COMMON FACTOR OF TWO INTEGERS
9004  REM
9005  REM  ***  CONTRIBUTED PROGRAM  ************************************
9006  PRINT
9007  PRINT "WHAT ARE YOUR TWO INTEGERS";
9008  INPUT A,B
9009  X1=A MAX B
9010  X2=A MIN B
9011  DEF FNA(X)=X1/X2
9012  X3=X2
9015  X2=(X2*(FNA(1)-INT(FNA(1))))
9020  X2=X2+.5
9025  X2=INT(X2)
9030  X1=X3
9035  IF X2#0 THEN 9012
9040  PRINT "LARGEST COMMON FACTOR IS "X1
9999  END
```

FACTOR
A301-36037A

TITLE: FINDS PRIME FACTORS OF POSITIVE INTEGERS

DESCRIPTION: This program will find the prime factors of a number.

INSTRUCTIONS: The program will request the number to be factored and print out all prime factors and their multiplicity.

Input a zero (Ø) or negative number to terminate execution.

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS: The number to be factored must be a positive integer less than 32768.

ACKNOWLEDGEMENTS:

```
RUN

GET-$FACTOR
RUN
FACTOR

PROGRAM TO FIND PRIME FACTORS OF A POSITIVE INTEGER.
TO TERMINATE EXECUTION INPUT A '0'.

WHAT NUMBER IS TO BE FACTORED?77

THE PRIME FACTORS OF 77   ARE:
PRIME           MULTIPLICITY
-----           ------------
 7               1
 11              1

WHAT NUMBER IS TO BE FACTORED?147

THE PRIME FACTORS OF 147  ARE:
PRIME           MULTIPLICITY
-----           ------------
 3               1
 7               2

WHAT NUMBER IS TO BE FACTORED?0

DONE
```

LISTING

FACTOR

```
9000  REM  *****  FACTOR  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  FINDS PRIME FACTORS
9003  PRINT "PROGRAM TO FIND PRIME FACTORS OF A POSITIVE INTEGER."
9004  PRINT "TO TERMINATE EXECUTION INPUT A '0'."
9005  PRINT
9006  PRINT "WHAT NUMBER IS TO BE FACTORED";
9007  INPUT A
9008  IF A <= 32767 THEN 9012
9009  PRINT "SORRY!THIS PROGRAM IS ONLY DESIGNED TO FACTOR NUMBERS"
9010  PRINT "OF 5 DIGITS UP TO 32767 OR LESS!PLEASE TRY AGAIN"
9011  GOTO 9005
9012  LET D=A
9013  PRINT
9014  IF A=2 THEN 9044
9015  LET Q=0
9016  IF A>0 THEN 9018
9017  STOP
9018  LET C=2
9019  GOSUB 9023
9020  FOR C=3 TO SQR(A) STEP 2
9021  GOSUB 9023
9022  GOTO 9039
9023  LET B=0
9024  IF A=C*INT(A/C) THEN 9026
9025  GOTO 9029
9026  LET A=A/C
9027  LET B=B+1
9028  GOTO 9024
9029  IF B<1 THEN 9038
9030  IF Q=1 THEN 9037
9031  LET Q=1
9032  PRINT "THE PRIME FACTORS OF";D;"ARE:"
9033  PRINT
9034  PRINT "PRIME","MULTIPLICITY"
9035  PRINT "-----","------------"
9036  PRINT
9037  PRINT C,B
9038  RETURN
9039  NEXT C
9040  IF A=1 THEN 9005
9041  IF Q=0 THEN 9044
9042  PRINT A,1
9043  GOTO 9005
9044  PRINT "THE NUMBER";A;"IS PRIME"
9045  GOTO 9005
9046  STOP
9999  END
```

CALCOM/CALPLT
A301-36131A

TITLE: CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT

DESCRIPTION:

CALCOM and CALPLT allow the user to perform immediate mode calculations and other functions. The two programs are identical other than for the GRAPH command, which utilizes the HP 7200A Plotter with CALPLT, or the printing terminal with CALCOM.

The sample run utilized CALPLT (and the HP 7200A Plotter).

INSTRUCTIONS:

See Page 2.

SYSTEM SPECIFICATIONS:

2000B and Teletype

SPECIAL CONSIDERATIONS:

There is a heirarchy of operators with factorialization being performed first followed by the min and max functions, then exponentiation, multiplication and division, and finally addition and subtraction. Paranthesis may be used at any time to override the order in which the operations are performed.

In addition to performing direct calculations, the user may retain the results of a calculation as a variable consisting of a single letter.

Variables may be used in later calculations once they have been defined. Undefined variables are set to zero.

By using a backslash \ (shift L) the user may perform more than one calculation per line. The different calculations are performed from left to right in the command string.

ACKNOWLEDGEMENTS:

Steve Poulsen
OMSI

INSTRUCTIONS

The following symbols, commands, and functions are available:

SYMBOL	MEANING	EXAMPLE
+	Addition	2+5=7
-	Subtraction	5-2=3
*	Multiplication	2*5=10
/	Division	2/5=.4
↑ or ^	Exponentiation	2↑5=32
%	Root function A%B=B↑(1/A)	2%5=2.236
<	MIN function. Value is lesser number on either side	2<5=2
>	MAX function. Value is greater number on either side	2>5=5
!	Factorialization of number preceding !	5!=120
?	Value is supplied by user	W=?+3*?/2
\	Allows more than one command per line	2+5\FACTOR 314*W
← or _	Deletes preceding character	2+3_5=7

COMMAND	MEANING
BASE n	Changes input and output to base n
BASE	Changes input and output back to base 10
DEGREES	Allow trig functions to be evaluated in degrees
FACTOR	Prime factors number following command
GRAPH	Graphs functions following command on teleprinter (or plotter)
LIST	Lists variables not equal to zero
RADIANS	Allows trig functions to be evaluated in radians
SAME	Repeats last command string
SCRATCH	Sets all variables equal to zero
STOP	Stops the running of CALC
ZERO	Approximates the points at which the equation following the command is equal to zero

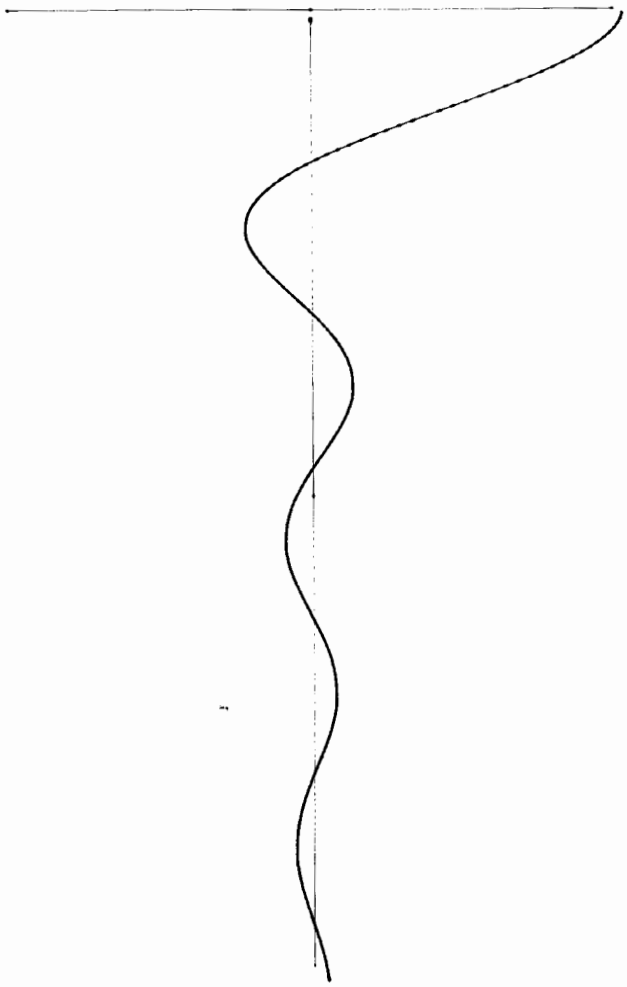
FUNCTION NAME	MEANING
ABS	Absolute value of number
COS	Cosine of angle
COT	Cotangent of angle
CSC	Cosecant of angle
EXP	"e" raised to a real power
INT	Integer part of number
LOG	Natural logarithm of number
RND	Random number between 0 and 1
SEC	Secant of angle
SIN	Sine of angle
TAN	Tangent of angle

Arc functions are called by placing the prefix ARC in front of the function such as: ARCSIN, ARCCOT, or ARCCSC.

Hyperbolic functions are called with the prefix HYP such as: HYPSIN, HYPCOS, HYPSEC, ARCHYPCOT, ARCHYPCSC, HYPARCTAN, or HYPARCCOS.

```
RUN

RUN
CALPLT

INTERPRETIVE CALCULATOR




0  2+3*(5/2)
 9.5
0  Q=2+3*(5/2)
0  Q*2
 19
0  2+3*(5/2)\Q*2
 9.5
 19
0  ?+3*(5/2)
INPUT DATA?2
 9.5
0  ?+3*(?/2)
INPUT DATA?2
INPUT DATA?5
 9.5
0  INT(2+3*(5/2))
 9
0  ABS(2-3*(5/2))
 5.5
0  COS(3.14159
COS(3.14159
           ↑
MISSING RIGHT PARENTHESIS
 3.14159
0  COS(3.14159)
-1.
0  ARCTAN(-1)
-.785398
0  FACTOR (52/2)
 2    * 13
0  Q=1024
0  BASE 2\Q
10000000000
0  10010*1001
10100010
0  BASE
0  A=2+2\B=2*3\C=2/5
0  LIST
A                4
B                6
C                .4
Q                1024
0  SCRATCH
0  LIST
0  2*?+?↑2
INPUT DATA?2
INPUT DATA?3
 13
0  SAME

INPUT DATA?1
INPUT DATA?2
 6
0  ZERO Y=X↑3-X↑2-10*X-8
LOWER LIMIT OF SEARCH?-8
UPPER LIMIT OF SEARCH?8
-2    -1     4
0  GRAPH Y=(SIN(X))/X
LOWER LIMIT OF X?0
UPPER LIMIT OF X?20
X INCREMENT?.1
X OFFSET?0
Y SCALING FACTOR?10
PLTL
 100   5000
 5000     5000
 9900     5000
PLTT
PLTL
 0     100
 0     5000
 0     9900
PLTT
PLTL
 0     4.99950E+33
 49    9991
 99    9966
 149   9924
 199   9867
 249   9793
 299   9704
 349   9601
 399   9483
 449   9351
 499   9206
 549   9050
 599   8883
 649   8705
 699   8519
 749   8324
 799   8123
 849   7916
 899   7704
 949   7490
 999   7273
 1049     7055
 1099     6837
 1149     6620
 1199     6407
 1249     6196
 1299     5991
 1349     5791
 1399     5598
 1449     5412
 1499     5235
 1549     5067
 1599     4908
 1649     4761
 1699     4624
 1749     4498
 1799     4385
 1849     4284
 1899     4195
 1949     4118
 1999     4054
 2049     4002
 2099     3962
 2149     3934
 2199     3918
 2249     3913
 2299     3920
 2349     3936
 2399     3962
 2449     3997
 2499     4041
 2549     4092
 2599     4150
 2649     4214
 2699     4284
 2749     4358
 2799     4436
 2849     4516
 2899     4599
 2949     4683
 2999     4767
 3049     4850
 3099     4933
 3149     5013
 3199     5091
 3249     5165
 3299     5235
 3349     5302
 3399     5363
 3449     5419
 3499     5469
 3549     5513
```

```
3599      5551
3649      5582
3699      5607
3749      5625
3799      5636
3849      5641
3899      5640
3949      5632
3999      5618
4049      5598
4099      5573
4149      5543
4199      5508
4249      5469
4299      5426
4349      5380
4399      5332
4449      5281
4499      5228
4549      5175
4599      5121
4649      5066
4699      5013
4749      4960
4799      4909
4849      4859
4899      4813
4949      4768
4999      4728
5049      4690
5099      4656
5149      4627
5199      4602
5249      4581
5299      4564
5349      4553
5399      4545
5449      4543
5499      4545
5549      4552
5599      4562
5649      4577
5699      4596
5749      4619
5799      4645
5849      4674
5899      4706
5949      4740
5999      4776
6049      4814
6099      4853
6149      4893
6199      4933
6249      4973
6299      5013
6349      5052
6399      5090
6449      5126
6499      5161
6549      5194
6599      5224
6649      5251
6699      5276
6749      5297
6799      5315
6849      5330
6899      5341
6949      5349
6999      5353
7049      5354
7099      5351
7149      5344
7199      5335
7249      5322
7299      5306
7349      5287
7399      5266
7449      5242
7499      5216
7549      5189
7599      5159
7649      5129
7699      5098
7749      5066
7799      5034
7849      5002
7899      4970
7949      4939
7999      4910
8049      4881
8099      4854
8149      4828
8199      4805
8249      4784
8299      4765
8349      4749
8399      4735
8449      4725
8499      4717
8549      4712
8599      4710
8649      4711
8699      4714
8749      4721
8799      4730
8849      4742
8899      4756
8949      4772
8999      4791
9049      4811
9099      4833
9149      4857
9199      4881
9249      4907
9299      4933
9349      4960
9399      4986
9449      5013
9499      5039
9549      5064
9599      5089
9649      5112
9699      5134
9749      5155
9799      5173
9849      5190
9899      5205
9948      5217
9998      5228
PLTT
0   STOP

DONE
```



LISTING

```
CALCOM
10   COM X,V[36]
11   REM
12   REM ***  HP TIME-SHARED BASIC PROGRAM LIBRARY  ************************
13   REM
14   REM          CALCOM:  KEYBOARD ENTRY CALCULATOR PROGRAM
15   REM
16   REM          36131 (A301) REV A -- 7/71  (PART 1 OF 2)
17   REM
18   REM ***  CONTRIBUTED PROGRAM  ****************************************
19   REM
20   REM          6/1/71 VERSION BY STEVE POULSEN OF OMSI
40   DIM P[70],A$[72],B$[72],S$[52],C$[37],F$[45]
50   S$="+-*/↑%<>!()\?0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ."
60   C$="BASDEGFACGRAHELLISQUARADSAMSCRSTOZER"
70   F$="ABSARCCOSCOTCSCEXPHYPINTLOGRNDSECSGNSINTAN"
80   MAT P=ZER
90   N3=0
100   MAT V=ZER
120   B=10
130   M1=O1=Q=Q1=E1=Q2=I1=D1=F=S1=A1=H1=P2=S3=X=0
140   P7=R1=R0=0
150   P=T1=L1=N1=O2=L=1
160   PRINT "INTERPRETIVE CALCULATOR"
170   PRINT "NNN"
180   PRINT "[0]  ";
190   T2=100
200   ENTER T2,T2,B$
```

```
210  IF T2 <> -256 THEN 240
220  PRINT "GGGG";
230  GOTO 190
240  P=1
250  M1=LEN(B$)
260  PRINT
270  IF P>M1 THEN 180
280  IF B$[P,P] <> " " THEN 310
290  P=P+1
300  GOTO 270
310  IF P+2>M1 THEN 420
320  O1=0
330  FOR Q=1 TO 36 STEP 3
340  IF B$[P,P+2] <> C$[Q,Q+2] THEN 400
350  O1=INT(Q/3)+1
360  P=P+1
370  IF P>M1 THEN 400
380  IF B$[P,P]="\" THEN 400
390  IF B$[P,P] <> " " THEN 360
400  NEXT Q
410  IF O1 <> 0 THEN 450
420  GOSUB 700
430  GOSUB 1010
440  GOTO 910
450  GOTO O1 OF 460,510,4470,4080,680,530,680,600,620,650,670,4720
460  GOSUB 1400
470  IF X>1 AND X<37 THEN 490
480  X=10
490  B=INT(X)
500  GOTO 910
510  T1=3.14159/180
520  GOTO 910
530  FOR Q7=B+14 TO LEN(S$)-1
540  IF V[Q7-14]=0 THEN 580
550  PRINT S$[Q7,Q7],
560  X=V[Q7-14]
570  GOSUB 1010
580  NEXT Q7
590  GOTO 910
600  T1=1
610  GOTO 910
620  B$=A$
630  S3=1
640  GOTO 240
650  MAT V=ZER
660  GOTO 910
670  STOP
680  PRINT "NOT YET IMPLEMENTED"
690  GOTO 910
700  Q=P-1
710  Q=Q+1
720  E1=0
730  IF Q>M1 THEN 890
740  IF B$[Q,Q]="\" THEN 890
750  IF B$[Q,Q] <> "=" THEN 710
760  Q2=Q
770  Q=Q-1
780  IF Q<P THEN 830
790  IF B$[Q,Q]=" " THEN 770
800  FOR Q1=14+B TO LEN(S$)-1
810  IF B$[Q,Q]=S$[Q1,Q1] THEN 870
820  NEXT Q1
830  P=Q
840  GOSUB 980
850  PRINT "ILLEGAL VARIABLE ON LEFT OF EQUAL SIGN"
860  GOTO 910
870  E1=Q1
880  P=Q2+1
890  GOSUB 1400
900  RETURN
910  A$=B$
920  IF B$[P,P] <> "\" THEN 950
930  P=P+1
940  GOTO 270
950  P=P+1
960  IF P>M1 THEN 180
970  GOTO 920
980  PRINT B$
```

```
990  PRINT TAB(P-1);"↑"
1000  RETURN
1010  E1=E1-14
1020  IF E1 <= .0001 THEN 1060
1030  V[E1]=X
1040  E1=0
1050  RETURN
1060  IF B <> 10 THEN 1090
1070  PRINT X
1080  RETURN
1090  IF X >= 0 THEN 1120
1100  PRINT "-";
1110  X=ABS(X)
1120  I1=INT(X)
1130  D1=X-I1
1140  IF X <> 0 THEN 1170
1150  PRINT "0"
1160  RETURN
1170  IF ABS(INT(LOG(X)/LOG(B)))<70 THEN 1200
1180  PRINT X;"(BASE 10)"
1190  RETURN
1200  A$=""
1210  B1=60
1220  X=I1
1230  B1=B1-1
1240  Q=X-INT(X/B)*B
1250  A$[B1,B1]=S$[Q+14,Q+14]
1260  X=INT(X/B)
1270  IF X>0 AND B1>2 THEN 1230
1280  X=D1
1290  A$=A$[B1,B1+LEN(A$)]
1300  B1=LEN(A$)
1310  IF D1=0 THEN 1380
1320  A$[B1,B1]="."
1330  Q=INT(X*B)
1340  B1=B1+1
1350  A$[B1,B1]=S$[Q+14,Q+14]
1360  X=X*B-Q
1370  IF B1<72 AND X <> 0 THEN 1330
1380  PRINT A$
1390  RETURN
1400  P=P-1
1410  N1=L1=1
1415  B1=0
1420  X=S1=P7=L=0
1430  P=P+1
1440  IF P <= M1 THEN 1520
1450  O2=0
1460  GOSUB 2690
1470  GOSUB 3150
1480  IF L1 <= 1 THEN 1510
1490  GOSUB 980
1500  PRINT "MISSING RIGHT PARENTHESIS"
1510  RETURN
1520  IF B$[P,P]=" " THEN 1430
1530  FOR Q=1 TO LEN(S$)
1540  IF B$[P,P]=S$[Q,Q] THEN 1590
1550  NEXT Q
1560  GOSUB 980
1570  PRINT "ILLEGAL CHARACTER"
1580  GOTO 1430
1590  C=Q
1600  IF C=12 THEN 1450
1610  IF C <> 11 THEN 1810
1620  O2=0
1630  GOSUB 2690
1650  N1=2
1660  IF L1>1 THEN 1700
1670  GOSUB 980
1680  PRINT "EXTRA RIGHT PARENTHESIS"
1690  GOTO 1430
1700  GOSUB 1770
1710  H1=INT(A/10)
1720  A1=A-H1*10
1730  GOSUB 1770
1740  L=INT(A)
1750  F=(A-L)*100
```

```
1755  GOSUB 3150
1758  GOSUB 2690
1760  GOTO 1430
1770  IF L1 <= 1 THEN 1670
1780  A=P[L1]
1790  L1=L1-1
1800  RETURN
1805  IF L=0 THEN 1890
1810  IF C <> 10 THEN 2040
1820  A=X
1830  X=0
1840  L=L+1
1850  GOSUB 1970
1860  A=O2
1870  O2=0
1880  GOSUB 1970
1890  A=L+F/100
1900  F=L=0
1910  GOSUB 1970
1920  A=H1*10+A1
1930  GOSUB 1970
1940  H1=A1=0
1950  N1=1
1960  GOTO 1420
1970  L1=L1+1
1980  IF L1<70 THEN 2020
1990  GOSUB 980
2000  PRINT "EXPRESSION TOO COMPLEX"
2010  GOTO 1620
2020  P[L1]=A
2030  RETURN
2040  GOSUB N1 OF 2060,2550
2050  GOTO 1430
2060  IF C=50 THEN 2150
2070  IF C<14 OR C>13+B THEN 2250
2080  IF P7>0 THEN 2120
2090  X=X*B+C-14
2100  S1=N1=1
2110  RETURN
2120  X=X+B↑(-P7)*(C-14)
2130  P7=P7+1
2140  GOTO 2100
2150  IF P7>0 THEN 2200
2160  P7=1
2170  S1=1
2180  N1=1
2190  RETURN
2200  GOSUB 980
2210  PRINT "ILLEGAL DECIMAL POINT"
2220  GOTO 2160
2230  N1=1
2240  RETURN
2250  IF S1 <> 1 THEN 2300
2260  P=P-1
2270  S1=P7=0
2280  N1=2
2290  RETURN
2300  IF C<13+B OR C>50 THEN 2490
2310  IF P+2>M1 THEN 2350
2320  FOR F=1 TO 14*3-1 STEP 3
2330  IF B$[P,P+2]=F$[F,F+2] THEN 2380
2340  NEXT F
2350  F=0
2360  X=V[C-14]
2370  GOTO 2270
2380  P=P+2
2390  F=INT(F/3)+1
2400  IF F <> 2 THEN 2440
2410  F=0
2420  A1=1
2430  GOTO 2230
2440  IF F <> 7 THEN 2230
2450  F=0
2460  H1=1
2470  GOTO 2230
2480  F=0
2490  IF C <> 13 THEN 2530
2500  PRINT "INPUT DATA";
```

```
2510  INPUT X
2520  GOTO 2270
2530  X=0
2540  GOTO 2260
2550  P7=S1=0
2560  IF C <> 9 THEN 2640
2570  Q=1
2580  FOR Q1=1 TO X
2590  Q=Q*Q1
2600  NEXT Q1
2610  X=Q
2620  N2=2
2630  RETURN
2640  IF C>8 THEN 2670
2650  O2=C
2660  GOTO 2690
2670  O2=3
2680  P=P-1
2690  IF L>0 THEN 2800
2700  IF O2=0 THEN 2790
2710  A=X
2720  X=0
2730  L=L+1
2740  GOSUB 1970
2750  A=O2
2760  O2=0
2770  GOSUB 1970
2780  N1=1
2790  RETURN
2800  GOSUB 1770
2810  IF INT((A+1)/2) >= INT((O2+1)/2) THEN 2840
2820  GOSUB 1970
2830  GOTO 2710
2840  O7=A
2850  GOSUB 1770
2860  L=L-1
2870  N3=A
2880  GOSUB O7 OF 2900,2920,2940,2960,3030,3080,3110,3130
2890  GOTO 2690
2900  X=N3+X
2910  RETURN
2920  X=N3-X
2930  RETURN
2940  X=N3*X
2950  RETURN
2960  IF X#0 THEN 3010
2970  IF P2#0 THEN 2990
2980  PRINT "DIVISION BY ZERO"
2990  X=1.E+30
3000  RETURN
3010  X=N3/X
3020  RETURN
3030  IF N3>0 THEN 3040
3032  IF X=INT(X) THEN 3040
3034  IF P2>0 THEN 2990
3036  PRINT "NEGATIVE NUMBER TO REAL POWER - - WARNING ONLY"
3038  N3=ABS(N3)
3040  X=N3↑X
3050  RETURN
3060  X=1
3070  RETURN
3080  IF N3=0 OR X=0 THEN 3060
3082  X0=1
3083  GOTO 2*(N3#INT(N3))+(P2>0)+1 OF 3084,3084,3086,2990
3084  X9=1+4*((N3/2)=INT(N3/2))+2*SGN(1+SGN(X))+(P2>0)
3085  GOTO X9 OF 3098,3098,3094,3094,3086,2990,3094,3090
3086  PRINT "ROOT OF NEGATIVE NUMBER - - WARNING ONLY"
3088  GOTO 3094
3090  B1=B1+1
3092  X0=1-2*((B0/(2↑B1))=INT(B0/(2↑B1)))
3094  X=ABS(X)↑(1/N3)*X0
3096  RETURN
3098  X0=-1
3099  GOTO 3094
3100  RETURN
3110  X=X MIN N3
3120  RETURN
3130  X=X MAX N3
```

```
3140  RETURN
3150  IF F <= 0 THEN 3190
3152  IF A1=0 THEN 3160
3154  X=X/T1
3160  F=INT(F*10+.5)/10
3170  IF F>9 THEN 3210
3180  GOSUB F OF 3230,3240,3250,3370,3490,3610,3240,3630,3650
3182  IF A1=0 THEN 3190
3184  X=X/T1
3190  A1=F=H1=0
3200  RETURN
3210  GOSUB F-9 OF 3670,3690,3810,3830,3950
3220  GOTO 3182
3230  X=ABS(X)
3240  RETURN
3250  X=X*T1
3260  GOTO 4-2*(A1=0)-(H1=0) OF 3350,3330,3310,3290
3290  X=LOG(X+SQR(X↑2-1))
3300  RETURN
3310  X=ATN(SQR(1-X↑2)/X)
3320  RETURN
3330  X=(EXP(X)+EXP(-X))/2
3340  RETURN
3350  X=COS(X)
3360  RETURN
3370  X=X*T1
3380  GOTO 4-2*(A1=0)-(H1=0) OF 3470,3450,3430,3410
3410  X=(LOG(X+1)-LOG(X-1))/X
3420  RETURN
3430  X=ATN(1/X)
3440  RETURN
3450  X=(EXP(X)+EXP(-X))/(EXP(X)-EXP(-X))
3460  RETURN
3470  X=1/TAN(X)
3480  RETURN
3490  X=X*T1
3500  GOTO 4-2*(A1=0)-(H1=0) OF 3590,3550,3570,3530
3530  X=LOG((1/X)+SQR((1/X↑2)+1))
3540  RETURN
3550  X=2/(EXP(X)-EXP(-X))
3560  RETURN
3570  X=ATN(1/SQR(X↑2-1))
3580  RETURN
3590  X=1/SIN(X)
3600  RETURN
3610  X=EXP(X)
3620  RETURN
3630  X=INT(X)
3640  RETURN
3650  X=LOG(X)
3660  RETURN
3670  X=RND(-X)
3680  RETURN
3690  X=X*T1
3700  GOTO 4-2*(A1=0)-(H1=0) OF 3790,3770,3750,3730
3730  X=LOG((1/X)+SQR((1/X↑2)-1))
3740  RETURN
3750  X=ATN(SQR(X↑2-1))
3760  RETURN
3770  X=2/(EXP(X)+EXP(-X))
3780  RETURN
3790  X=1/COS(X)
3800  RETURN
3810  X=SGN(X)
3820  RETURN
3830  X=X*T1
3840  GOTO 4-2*(A1=0)-(H1=0) OF 3930,3910,3890,3870
3870  X=LOG(X+SQR(X↑2+1))
3880  RETURN
3890  X=ATN(X/SQR(1-X↑2))
3900  RETURN
3910  X=(EXP(X)-EXP(-X))/2
3920  RETURN
3930  X=SIN(X)
3940  RETURN
3950  X=X*T1
3960  GOTO 4-2*(A1=0)-(H1=0) OF 4050,4030,4010,3990
3990  X=(LOG(1+X)-LOG(1-X))/2
```

```
4000  RETURN
4010  X=ATN(X)
4020  RETURN
4030  X=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))
4040  RETURN
4050  X=TAN(X)
4060  RETURN
4070  REM GRAPHING ROUTINE
4080  Q3=P
4085  P2=1
4090  PRINT "LOWER LIMIT OF X";
4100  INPUT G2
4110  PRINT "UPPER LIMIT OF X";
4120  INPUT G3
4130  PRINT "X INCREMENT";
4140  INPUT G4
4150  PRINT "X OFFSET";
4160  INPUT G5
4170  PRINT "Y SCALING FACTOR";
4180  INPUT G6
4182  GOSUB 700
4184  GOSUB 1010
4186  B3=2↑B1
4190  IF G6 <> 0 THEN 4210
4200  G6=.85
4210  FOR X7=INT(G2/G4)*G4 TO INT(G3/G4)*G4 STEP G4
4220  FOR G7=1 TO 72
4230  A$[G7,G7]=" "
4240  NEXT G7
4250  G8=0
4260  IF ABS(2*G5)>35 THEN 4290
4270  G8=35+2*G5
4280  A$[G8,G8]="."
4290  IF ABS(X7)>.00001 THEN 4360
4300  A$[10,10]="Y"
4310  FOR G7=11 TO 61 STEP 2
4320  A$[G7,G7+1]=". "
4330  NEXT G7
4340  A$[62,62]="Y"
4350  G8=63
4360  FOR B0=1 TO B3
4365  P=Q3
4370  V[33]=X7
4380  GOSUB 700
4390  GOSUB 1010
4400  Y5=INT(35+2*V[34]*G6+G5)
4410  IF Y5>72 OR Y5<1 THEN 4425
4415  G8=G8 MAX Y5
4420  A$[Y5,Y5]="*"
4425  NEXT B0
4430  PRINT A$[1,G8 MIN 72]
4440  NEXT X7
4445  P2=0
4450  GOTO 910
4460  REM PRIME FACTORING ROUTINE
4470  GOSUB 1400
4480  X=INT(ABS(X))
4490  IF X=0 THEN 4700
4500  R5=SQR(X)
4510  C3=0
4520  X5=2
4530  GOTO 4590
4540  C3=0
4550  IF X5>2 THEN 4570
4560  X5=1
4570  X5=X5+2
4580  IF R5<X5 THEN 4700
4590  Q3=X/X5
4600  IF Q3 <> INT(Q3) THEN 4640
4610  X=Q3
4620  C3=C3+1
4630  IF X>1 THEN 4590
4640  IF C3=0 THEN 4550
4650  IF C3=1 THEN 4680
4660  PRINT X5;"↑";C3;"*";
4670  GOTO 4690
4680  PRINT X5;"*";
4690  IF X>1 THEN 4540
```

```
4700  PRINT X
4710  GOTO 910
4720  PRINT "LOWER LIMIT OF SEARCH";
4730  INPUT Z8
4740  PRINT "UPPER LIMIT OF SEARCH";
4750  INPUT Z9
4760  Q5=P
4770  V[33]=Z8
4780  I7=1
4790  Q3=0
4800  P=Q5
4810  GOSUB 700
4820  GOSUB 1010
4830  IF V[34] <> 0 THEN 4870
4840  PRINT V[33];
4850  V[33]=INT(1+V[33])
4860  GOTO 4780
4870  V[33]=V[33]+I7
4880  Y5=V[34]
4890  P=Q5
4900  GOSUB 700
4910  GOSUB 1010
4920  IF V[34]=0 THEN 4840
4930  IF SGN(V[34]) <> SGN(Y5) THEN 5000
4940  IF Q3=0 THEN 4980
4950  I7=I7/2
4960  Q3=Q3+1
4970  IF Q3>25 THEN 4840
4980  IF V[33]>Z9 THEN 5050
4990  GOTO 4870
5000  I7=I7/2
5010  V[33]=V[33]-I7
5020  Q3=Q3+1
5030  IF Q3>25 THEN 4840
5040  GOTO 4890
5050  PRINT
5060  GOTO 910
5070  END
```

```
CALPLT
10  COM X,V[36]
11  REM
12  REM  ***  HP TIME-SHARED BASIC PROGRAM LIBRARY  **********************
13  REM
14  REM          CALPLT:  KEYBOARD ENTRY CALCULATOR PROGRAM WITH 7200A
15  REM                   GRAPHIC PLOTTER OUTPUT
16  REM          36131 (A301) REV A -- 7/71  (PART 2 OF 2)
17  REM
18  REM  ***  CONTRIBUTED PROGRAM  ***************************************
20  REM *** CALCOM ***; 6/1/71 VERSION BY STEVE POULSEN OF OMSI
40  DIM P[70],A$[72],B$[72],S$[52],C$[37],F$[45]
50  S$="+-*/↑%<>!()\?0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ."
60  C$="BASDEGEACGRAHELLISQUARADSAMSCRSTOZER"
70  F$="ABSARCCOSCOTCSCEXPHYPINTLOGRNDSECSGNSINTAN"
80  MAT P=ZER
90  N3=0
100  MAT V=ZER
120  B=10
130  M1=O1=Q=Q1=F1=Q2=I1=D1=F=S1=A1=H1=P2=S3=X=0
140  P7=B1=R0=0
150  P=T1=L1=N1=O2=L=1
160  PRINT "INTERPRETIVE CALCULATOR"
170  PRINT "NNN"
180  PRINT "[0]  ";
190  T2=100
200  ENTER T2,T2,B$
210  IF T2 <> -256 THEN 240
220  PRINT "GGGG";
230  GOTO 190
240  P=1
250  M1=LEN(B$)
260  PRINT
270  IF P>M1 THEN 180
280  IF B$[P,P] <> " " THEN 310
290  P=P+1
300  GOTO 270
310  IF P+2>M1 THEN 420
```

```
320  O1=0
330  FOR Q=1 TO 36 STEP 3
340  IF B$[P,P+2] <> C$[Q,Q+2] THEN 400
350  O1=INT(Q/3)+1
360  P=P+1
370  IF P>M1 THEN 400
380  IF B$[P,P]="\" THEN 400
390  IF B$[P,P] <> " " THEN 360
400  NEXT Q
410  IF O1 <> 0 THEN 450
420  GOSUB 700
430  GOSUB 1010
440  GOTO 910
450  GOTO O1 OF 460,510,4470,4080,680,530,680,600,620,650,670,4720
460  GOSUB 1400
470  IF X>1 AND X<37 THEN 490
480  X=10
490  B=INT(X)
500  GOTO 910
510  T1=3.14159/180
520  GOTO 910
530  FOR Q7=B+14 TO LEN(S$)-1
540  IF V[Q7-14]=0 THEN 580
550  PRINT S$[Q7,Q7],
560  X=V[Q7-14]
570  GOSUB 1010
580  NEXT Q7
590  GOTO 910
600  T1=1
610  GOTO 910
620  B$=A$
630  S3=1
640  GOTO 240
650  MAT V=ZER
660  GOTO 910
670  STOP
680  PRINT "NOT YET IMPLEMENTED"
690  GOTO 910
700  Q=P-1
710  Q=Q+1
720  E1=0
730  IF Q>M1 THEN 890
740  IF B$[Q,Q]="\" THEN 890
750  IF B$[Q,Q] <> "=" THEN 710
760  Q2=Q
770  Q=Q-1
780  IF Q<P THEN 830
790  IF B$[Q,Q]=" " THEN 770
800  FOR Q1=14+B TO LEN(S$)-1
810  IF B$[Q,Q]=S$[Q1,Q1] THEN 870
820  NEXT Q1
830  P=0
840  GOSUB 980
850  PRINT "ILLEGAL VARIABLE ON LEFT OF EQUAL SIGN"
860  GOTO 910
870  E1=Q1
880  P=Q2+1
890  GOSUB 1400
900  RETURN
910  A$=B$
920  IF B$[P,P] <> "\" THEN 950
930  P=P+1
940  GOTO 270
950  P=P+1
960  IF P>M1 THEN 180
970  GOTO 920
980  PRINT B$
990  PRINT TAB(P-1);"↑"
1000  RETURN
1010  E1=E1-14
1020  IF E1 <= .0001 THEN 1060
1030  V[E1]=X
1040  E1=0
1050  RETURN
1060  IF B <> 10 THEN 1090
1070  PRINT X
1080  RETURN
1090  IF X >= 0 THEN 1120
```

```
1100  PRINT "-";
1110  X=ABS(X)
1120  I1=INT(X)
1130  D1=X-I1
1140  IF X <> 0 THEN 1170
1150  PRINT "0"
1160  RETURN
1170  IF ABS(INT(LOG(X)/LOG(B)))<70 THEN 1200
1180  PRINT X;"(BASE 10)"
1190  RETURN
1200  A$=""
1210  B1=60
1220  X=I1
1230  B1=B1-1
1240  Q=X-INT(X/B)*B
1250  A$[B1,B1]=S$[Q+14,Q+14]
1260  X=INT(X/B)
1270  IF X>0 AND B1>2 THEN 1230
1280  X=D1
1290  A$=A$[B1,B1+LEN(A$)]
1300  B1=LEN(A$)
1310  IF D1=0 THEN 1380
1320  A$[B1,B1]="."
1330  Q=INT(X*B)
1340  B1=B1+1
1350  A$[B1,B1]=S$[Q+14,Q+14]
1360  X=X*B-Q
1370  IF B1<72 AND X <> 0 THEN 1330
1380  PRINT A$
1390  RETURN
1400  P=P-1
1410  N1=L1=1
1415  B1=0
1420  X=S1=P7=L=0
1430  P=P+1
1440  IF P <= M1 THEN 1520
1450  O2=0
1460  GOSUB 2690
1470  GOSUB 3150
1480  IF L1 <= 1 THEN 1510
1490  GOSUB 980
1500  PRINT "MISSING RIGHT PARENTHESIS"
1510  RETURN
1520  IF B$[P,P]=" " THEN 1430
1530  FOR Q=1 TO LEN(S$)
1540  IF B$[P,P]=S$[Q,Q] THEN 1590
1550  NEXT Q
1560  GOSUB 980
1570  PRINT "ILLEGAL CHARACTER"
1580  GOTO 1430
1590  C=Q
1600  IF C=12 THEN 1450
1610  IF C <> 11 THEN 1810
1620  O2=0
1630  GOSUB 2690
1650  N1=2
1660  IF L1>1 THEN 1700
1670  GOSUB 980
1680  PRINT "EXTRA RIGHT PARENTHESIS"
1690  GOTO 1430
1700  GOSUB 1770
1710  H1=INT(A/10)
1720  A1=A-H1*10
1730  GOSUB 1770
1740  L=INT(A)
1750  F=(A-L)*100
1755  GOSUB 3150
1758  GOSUB 2690
1760  GOTO 1430
1770  IF L1 <= 1 THEN 1670
1780  A=P[L1]
1790  L1=L1-1
1800  RETURN
1805  IF L=0 THEN 1890
1810  IF C <> 10 THEN 2040
1820  A=X
1830  X=0
```

```
1840  L=L+1
1850  GOSUB 1970
1860  A=O2
1870  O2=0
1880  GOSUB 1970
1890  A=L+F/100
1900  F=L=0
1910  GOSUB 1970
1920  A=H1*10+A1
1930  GOSUB 1970
1940  H1=A1=0
1950  N1=1
1960  GOTO 1420
1970  L1=L1+1
1980  IF L1<70 THEN 2020
1990  GOSUB 980
2000  PRINT "EXPRESSION TOO COMPLEX"
2010  GOTO 1620
2020  P[L1]=A
2030  RETURN
2040  GOSUB N1 OF 2060,2550
2050  GOTO 1430
2060  IF C=50 THEN 2150
2070  IF C<14 OR C>13+B THEN 2250
2080  IF P7>0 THEN 2120
2090  X=X*B+C-14
2100  S1=N1=1
2110  RETURN
2120  X=X+B↑(-P7)*(C-14)
2130  P7=P7+1
2140  GOTO 2100
2150  IF P7>0 THEN 2200
2160  P7=1
2170  S1=1
2180  N1=1
2190  RETURN
2200  GOSUB 980
2210  PRINT "ILLEGAL DECIMAL POINT"
2220  GOTO 2160
2230  N1=1
2240  RETURN
2250  IF S1 <> 1 THEN 2300
2260  P=P-1
2270  S1=P7=0
2280  N1=2
2290  RETURN
2300  IF C<13+B OR C>50 THEN 2490
2310  IF P+2>M1 THEN 2350
2320  FOR F=1 TO 14*3-1 STEP 3
2330  IF B$[P,P+2]=F$[F,F+2] THEN 2380
2340  NEXT F
2350  F=0
2360  X=V[C-14]
2370  GOTO 2270
2380  P=P+2
2390  F=INT(F/3)+1
2400  IF F <> 2 THEN 2440
2410  F=0
2420  A1=1
2430  GOTO 2230
2440  IF F <> 7 THEN 2230
2450  F=0
2460  H1=1
2470  GOTO 2230
2480  F=0
2490  IF C <> 13 THEN 2530
2500  PRINT "INPUT DATA";
2510  INPUT X
2520  GOTO 2270
2530  X=0
2540  GOTO 2260
2550  P7=S1=0
2560  IF C <> 9 THEN 2640
2570  Q=1
2580  FOR Q1=1 TO X
2590  Q=Q*Q1
2600  NEXT Q1
2610  X=Q
```

```
2620  N2=2
2630  RETURN
2640  IF C>8 THEN 2670
2650  O2=C
2660  GOTO 2690
2670  O2=3
2680  P=P-1
2690  IF L>0 THEN 2800
2700  IF O2=0 THEN 2790
2710  A=X
2720  X=0
2730  L=L+1
2740  GOSUB 1970
2750  A=O2
2760  O2=0
2770  GOSUB 1970
2780  N1=1
2790  RETURN
2800  GOSUB 1770
2810  IF INT((A+1)/2) >= INT((O2+1)/2) THEN 2840
2820  GOSUB 1970
2830  GOTO 2710
2840  O7=A
2850  GOSUB 1770
2860  L=L-1
2870  N3=A
2880  GOSUB O7 OF 2900,2920,2940,2960,3030,3080,3110,3130
2890  GOTO 2690
2900  X=N3+X
2910  RETURN
2920  X=N3-X
2930  RETURN
2940  X=N3*X
2950  RETURN
2960  IF X#0 THEN 3010
2970  IF P2#0 THEN 2990
2980  PRINT "DIVISION BY ZERO"
2990  X=1.E+30
3000  RETURN
3010  X=N3/X
3020  RETURN
3030  IF N3>0 THEN 3040
3032  IF X=INT(X) THEN 3040
3034  IF P2>0 THEN 2990
3036  PRINT "NEGATIVE NUMBER TO REAL POWER - - WARNING ONLY"
3038  N3=ABS(N3)
3040  X=N3↑X
3050  RETURN
3060  X=1
3070  RETURN
3080  IF N3=0 OR X=0 THEN 3060
3082  X0=1
3083  GOTO 2*(N3#INT(N3))+(P2>0)+1 OF 3084,3084,3086,2990
3084  X9=1+4*((N3/2)=INT(N3/2))+2*SGN(1+SGN(X))+(P2>0)
3085  GOTO X9 OF 3098,3098,3094,3094,3086,2990,3094,3090
3086  PRINT "ROOT OF NEGATIVE NUMBER - - WARNING ONLY"
3088  GOTO 3094
3090  B1=B1+1
3092  X0=1-2*((B0/(2↑B1))=INT(B0/(2↑B1)))
3094  X=ABS(X)↑(1/N3)*X0
3096  RETURN
3098  X0=-1
3099  GOTO 3094
3100  RETURN
3110  X=X MIN N3
3120  RETURN
3130  X=X MAX N3
3140  RETURN
3150  IF F <= 0 THEN 3190
3152  IF A1=0 THEN 3160
3154  X=X/T1
3160  F=INT(F*10+.5)/10
3170  IF F>9 THEN 3210
3180  GOSUB F OF 3230,3240,3250,3370,3490,3610,3240,3630,3650
3182  IF A1=0 THEN 3190
3184  X=X/T1
3190  A1=F=H1=0
3200  RETURN
```

```
3210  GOSUB F-9 OF 3670,3690,3810,3830,3950
3220  GOTO 3182
3230  X=ABS(X)
3240  RETURN
3250  X=X*T1
3260  GOTO 4-2*(A1=0)-(H1=0) OF 3350,3330,3310,3290
3290  X=LOG(X+SQR(X↑2-1))
3300  RETURN
3310  X=ATN(SQR(1-X↑2)/X)
3320  RETURN
3330  X=(EXP(X)+EXP(-X))/2
3340  RETURN
3350  X=COS(X)
3360  RETURN
3370  X=X*T1
3380  GOTO 4-2*(A1=0)-(H1=0) OF 3470,3450,3430,3410
3410  X=(LOG(X+1)-LOG(X-1))/X
3420  RETURN
3430  X=ATN(1/X)
3440  RETURN
3450  X=(EXP(X)+EXP(-X))/(EXP(X)-EXP(-X))
3460  RETURN
3470  X=1/TAN(X)
3480  RETURN
3490  X=X*T1
3500  GOTO 4-2*(A1=0)-(H1=0) OF 3590,3550,3570,3530
3530  X=LOG((1/X)+SQR((1/X↑2)+1))
3540  RETURN
3550  X=2/(EXP(X)-EXP(-X))
3560  RETURN
3570  X=ATN(1/SQR(X↑2-1))
3580  RETURN
3590  X=1/SIN(X)
3600  RETURN
3610  X=EXP(X)
3620  RETURN
3630  X=INT(X)
3640  RETURN
3650  X=LOG(X)
3660  RETURN
3670  X=RND(-X)
3680  RETURN
3690  X=X*T1
3700  GOTO 4-2*(A1=0)-(H1=0) OF 3790,3770,3750,3730
3730  X=LOG((1/X)+SQR((1/X↑2)-1))
3740  RETURN
3750  X=ATN(SQR(X↑2-1))
3760  RETURN
3770  X=2/(EXP(X)+EXP(-X))
3780  RETURN
3790  X=1/COS(X)
3800  RETURN
3810  X=SGN(X)
3820  RETURN
3830  X=X*T1
3840  GOTO 4-2*(A1=0)-(H1=0) OF 3930,3910,3890,3870
3870  X=LOG(X+SQR(X↑2+1))
3880  RETURN
3890  X=ATN(X/SQR(1-X↑2))
3900  RETURN
3910  X=(EXP(X)-EXP(-X))/2
3920  RETURN
3930  X=SIN(X)
3940  RETURN
3950  X=X*T1
3960  GOTO 4-2*(A1=0)-(H1=0) OF 4050,4030,4010,3990
3990  X=(LOG(1+X)-LOG(1-X))/2
4000  RETURN
4010  X=ATN(X)
4020  RETURN
4030  X=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))
4040  RETURN
4050  X=TAN(X)
4060  RETURN
4070  REM GRAPHING ROUTINE
4080  Q3=P
4085  P2=1
4090  PRINT "LOWER LIMIT OF X";
```

```
4100  INPUT G2
4110  PRINT "UPPER LIMIT OF X";
4120  INPUT G3
4130  PRINT "X INCREMENT";
4140  INPUT G4
4150  PRINT "X OFFSET";
4160  INPUT G5
4170  PRINT "Y SCALING FACTOR";
4180  INPUT G6
4182  GOSUB 700
4184  GOSUB 1010
4186  B3=2↑B1
4190  B5=9999/(G3-G2)
4192  B6=G6*B5
4194  B7=-G2*B5
4196  B8=5000+G5*B6
4200  PRINT "PLTLON";100;INT(B8);"ON";5000;INT(B8);"ON"9900;INT(B8)
4205  PRINT "PLTTONPLTLON";INT(B7);100;"ON";INT(B7);5000;"ON";INT(B7);9900
4207  PRINT "PLTTONPLTL"
4210  FOR X7=INT(G2/G4)*G4 TO INT(G3/G4)*G4 STEP G4
4220  FOR G7=1 TO 72
4230  A$[G7,G7]=" "
4240  NEXT G7
4250  G8=0
4260  IF ABS(2*G5)>35 THEN 4290
4270  G8=35+2*G5
4280  A$[G8,G8]="."
4290  IF ABS(X7)>.00001 THEN 4360
4300  A$[10,10]="Y"
4310  FOR G7=11 TO 61 STEP 2
4320  A$[G7,G7+1]=". "
4330  NEXT G7
4340  A$[62,62]="Y"
4350  G8=63
4360  FOR B0=1 TO B3
4365  P=Q3
4370  V[33]=X7
4380  GOSUB 700
4390  GOSUB 1010
4400  Y5=INT(35+2*V[34]*G6+G5)
4410  IF Y5>72 OR Y5<1 THEN 4425
4415  G8=G8 MAX Y5
4420  A$[Y5,Y5]="*"
4425  NEXT B0
4430  PRINT INT(B5*X7+B7);INT(B8+V[34]*B6)
4440  NEXT X7
4441  PRINT "PLTT"
4445  P2=0
4450  GOTO 910
4460  REM PRIME FACTORING ROUTINE
4470  GOSUB 1400
4480  X=INT(ABS(X))
4490  IF X=0 THEN 4700
4500  R5=SQR(X)
4510  C3=0
4520  X5=2
4530  GOTO 4590
4540  C3=0
4550  IF X5>2 THEN 4570
4560  X5=1
4570  X5=X5+2
4580  IF R5<X5 THEN 4700
4590  Q3=X/X5
4600  IF Q3 <> INT(Q3) THEN 4640
4610  X=Q3
4620  C3=C3+1
4630  IF X>1 THEN 4590
4640  IF C3=0 THEN 4550
4650  IF C3=1 THEN 4680
4660  PRINT X5;"↑";C3;"*";
4670  GOTO 4690
4680  PRINT X5;"*";
4690  IF X>1 THEN 4540
4700  PRINT X
4710  GOTO 910
4720  PRINT "LOWER LIMIT OF SEARCH";
4730  INPUT Z8
4740  PRINT "UPPER LIMIT OF SEARCH";
```

```
4750  INPUT Z9
4760  Q5=P
4770  V[33]=Z8
4780  I7=1
4790  Q3=0
4800  P=Q5
4810  GOSUB 700
4820  GOSUB 1010
4830  IF V[34] <> 0 THEN 4870
4840  PRINT V[33];
4850  V[33]=INT(1+V[33])
4860  GOTO 4780
4870  V[33]=V[33]+I7
4880  Y5=V[34]
4890  P=Q5
4900  GOSUB 700
4910  GOSUB 1010
4920  IF V[34]=0 THEN 4840
4930  IF SGN(V[34]) <> SGN(Y5) THEN 5000
4940  IF Q3=0 THEN 4980
4950  I7=I7/2
4960  Q3=Q3+1
4970  IF Q3>25 THEN 4840
4980  IF V[33]>Z9 THEN 5050
4990  GOTO 4870
5000  I7=I7/2
5010  V[33]=V[33]-I7
5020  Q3=Q3+1
5030  IF Q3>25 THEN 4840
5040  GOTO 4890
5050  PRINT
5060  GOTO 910
5070  END
```

GOLF
A903-36102A

TITLE: A GOLF GAME

DESCRIPTION:

This program is a game of computerized golf in which one player plays against obstacles such as sandtraps and slicing with numbers.

INSTRUCTIONS:

Input variables as follows:

```
Line 9   input H  up to 2  digits
Line 19  input T  up to 1  digit
Line 49  input C  up to 2  digits
Line 115 input W  up to 3  digits
Line 169 input I  up to 3  digits
Line 195 input A$ up to 9  characters, a'yes'or'no'answer
```

SYSTEM SPECIFICATIONS: 2000A and Teletype

SPECIAL CONSIDERATIONS: None

ACKNOWLEDGEMENTS: Richard Nelson
Bloomington, Minnesota

RUN

```
GET-GOLF
RUN
GOLF
WELCOME TO THE TIES TIMESHARING 18 HOLE CHAMPIONSHIP COURSE

TO GET A DESCRIPTION OF CLUBS, ETC. TYPE 0 FOR A CLUB NK. WHEN REQUESTED

WHAT IS YOUR HANDICAP ?10
DIFFICULTIES AT GOLF INCLUDE:
   0=HOOK, 1=SLICE, 2=POOR DISTANCE, 4=TRAP SHOTS, 5=PUTTING
WHICH IS YOUR WORST ?0

READY TO GO ?YES
YOU ARE AT TEE OFF HOLE 1     , DISTANCE 361  YARDS PAR 4

ON YOUR RIGHT IS ADJACENT FAIRWAY.
ON YOUR LEFT IS ADJACENT FAIRWAY.

WHAT CLUB DO YOU WANT ?0
----------------------------------------------------------------------
HERE'S YOUR BAG OF CLUBS

  WOODS (FULL SWING ONLY)
     1   DRIVER
     2   BRASSIE
     3   SPOON

  IRONS (FULL SWING ONLY)
     12   TWO IRON
     .
     .
     19   NINE IRON

  IRONS (LESS THEN FULL SWING)
     22   TWO IRON - PARTIAL SWING
     .
     .
     29   NINE IRON - PARTIAL SWING


WHEN YOU REACH THE GREEN IT WILL BE ASSUMED THAT YOU ARE
USING A PUTTER. THE PUTT POTENCY NO. REFERS TO THE STRENGTH
WITH WHICH THE BALL IS PUTTED. USE NUMBERS GREATER THAN
ZERO, INCREASING THE NUMBER FOR GREATER DISTANCE.

YOU WILL BE ASKED FOR 'PERCENT FULL SWING' ON CLUBS 22-29.
THIS SHOULD BE A NUMBER FROM 1 TO 99.
----------------------------------------------------------------------

WHAT CLUB DO YOU WANT ?1
YOU HOOKED-
           SHOT WENT 247  YARDS - IS 118  YARDS FROM HOLE.
BALL IS 31   YARDS OFF LINE IN ROUGH.

WHAT CLUB DO YOU WANT ?27
PERCENT FULL SWING ?65

SHOT WENT 98   YARDS - IS 22   YARDS FROM HOLE.
BALL IS 10   YARDS OFF LINE IN TRAP.

WHAT CLUB DO YOU WANT ?29
PERCENT FULL SWING ?80
ON GREEN 1    FEET FROM PIN. PUT POTENCY NUMBER ?1
PASSED BY CUP.
ON GREEN 2    FEET FROM PIN. PUT POTENCY NUMBER ?1
YOU H*O*L*E*D IT

YOUR SCORE ON HOLE 1    WAS 5

READY TO GO ?NO

TOTAL SCORE FOR 1    HOLES WAS 5
 1    OVER PAR

DONE
```

STORE 5

LISTING

GOLF

```
1  DIM L[11],A$[9]
2  PRINT "WELCOME TO THE [illegible]ING 18 HOLE CHAMPIONSHIP COURSE"
3  GOSUB 234
4  X=3
5  N=.8
6  S1=S2=T5=0
7  F=1
8  PRINT "WHAT IS YOUR HANDICAP ";
9  INPUT H
10  IF H>[illegible] OR H<0 THEN 220
11  IF H>9 THEN 14
12  PRINT "OH-OH, A HOT SHOT!"
13  GOTO 16
14  IF H<20 THEN 16
15  PRINT "ANYONE FOR TENNIS?"
16  PRINT "DIFFICULTIES AT GOLF INCLUDE:"
17  PRINT "  0=HOOK, 1=SLICE, 2=POOR DISTANCE, 4=TRAP SHOTS, 5=PUTTING"
18  PRINT "WHICH IS YOUR WORST ";
19  INPUT T
20  IF T>5 OR T<0 OR T=3 THEN 18
21  FOR Z=1 TO (H+1)/(T+1)*10
22  R=RND(0)
23  NEXT Z
24  J=Q=K=L[1]=0
25  S2=S2+S1
26  IF F=1 THEN 40
27  PRINT "YOUR SCORE ON HOLE"F-1"WAS"S1
28  GOTO P-S1+3 OF 39,39,31,33,35
29  PRINT "KEEP YOUR HEAD DOWN."
30  GOTO 39
31  PRINT "A PAR. NICE GOING."
32  GOTO 39
33  PRINT "A BIRDIE, AND TWEET-TWEET TO YOU."
34  GOTO 39
35  IF P=3 THEN 38
36  PRINT "A GREAT BIG EAGLE."
37  GOTO 39
38  PRINT "A HOLE IN *[illegible]O*[illegible]N*[illegible]E*."
39  IF F=19 THEN 208
40  S1=0
41  IF S1=0 THEN 193
42  IF L[1]<1 THEN 146
43  X=1
44  IF L[1]>5 THEN 150
45  PRINT "SHOT WENT"D1"YARDS - IS"D2"YARDS FROM HOLE."
46  PRINT "BALL IS"INT(0)"YARDS OFF LINE IN ";
47  GOSUB 217
48  PRINT "WHAT CLUB DO YOU WANT ";
49  INPUT C
50  IF C THEN 70
51  FOR T6=1 TO 69
52  PRINT "-";
53  NEXT T6
54  PRINT
55  IF C=9 THEN 48
56  PRINT "HERE'S YOUR BAG OF CLUBS[illegible]  WOODS (FULL SWING ONLY)"
57  PRINT "     1   DRIVER[illegible]     2   BRASSIE[illegible]     3   SPOON[illegible]"
58  PRINT "  IRONS (FULL SWING ONLY)[illegible]     12   TWO IRON[illegible]     .[illegible]     ."
59  PRINT "     19   NINE IRONONN  IRONS (LESS THEN FULL SWING)"
60  PRINT "     22   TWO IRON - PARTIAL SWING[illegible]     .[illegible]     ."
61  PRINT "     29   NINE IRON - PARTIAL SWING[illegible]"
62  PRINT "WHEN YOU REACH THE GREEN IT WILL BE ASSUMED THAT YOU ARE"
63  PRINT "USING A PUTTER. THE PUTT POTENCY NO. REFERS TO THE STRENGTH"
64  PRINT "WITH WHICH THE BALL IS PUTTED. USE NUMBERS GREATER THAN"
65  PRINT "ZERO, INCREASING THE NUMBER FOR GREATER DISTANCE.ON"
66  PRINT "YOU WILL BE ASKED FOR 'PERCENT FULL SWING' ON CLUBS 22-29."
67  PRINT "THIS SHOULD BE A NUMBER FROM 1 TO 99."
68  C=9
69  GOTO 51
70  IF C<1 OR C>29 THEN 76
71  IF C>3 THEN 78
72  IF L[1]<5 OR C=14 OR C=23 THEN 81
73  GOTO 76
74  S1=S1-1
75  W=1
76  GOSUB 234
77  GOTO 48
78  IF C<12 THEN 76
```

```
 79  C=C-6
 80  GOTO 72
 81  S1=S1+1
 82  W=1
 83  IF C>13 THEN 114
 84  IF F/3=INT(F/3) THEN 110
 85  IF C<4 THEN 87
 86  GOTO 88
 87  IF L[1]=2 THEN 96
 88  IF S1>7 THEN 99
 89  D1=INT(((30-H)*2.5+187-((30-H)*.25+15)*C/2)+25*RND(0))
 90  D1=INT(D1*W)
 91  IF T=2 THEN 148
 92  O=(RND(0)/.8)*(2*H+16)*ABS(TAN(D1*.0035))
 93  D2=INT(SQR(O+2+ABS(D-D1)+2))
 94  IF D-D1<0 THEN 101
 95  GOTO 103
 96  PRINT "YOU DUBBED IT."
 97  D1=35
 98  GOTO 92
 99  IF D<200 THEN 161
100  GOTO 89
101  IF D2<20 THEN 103
102  PRINT "TOO MUCH CLUB. YOU ARE PAST HOLE."
103  B=D
104  D=D2
105  IF D2>27 THEN 128
106  IF D2>20 THEN 141
107  IF D2>.5 THEN 143
108  L[1]=9
109  GOTO 182
110  IF (72+((H+1)/.85))/18<S2+Q+(10*(F-1)/18) THEN 85
111  Q=Q+1
112  IF S1/2#INT(S1/2) THEN 124
113  GOTO 96
114  PRINT "PERCENT FULL SWING ";
115  INPUT W
116  W=W/100
117  T7=0
118  GOSUB (W>0 AND W<1)+1 OF 234
119  IF T7 THEN 114
120  IF L[1]=5 THEN 160
121  IF C=14 THEN 88
122  C=C-10
123  GOTO 88
124  IF D<95 THEN 96
125  PRINT "BALL HIT TREE - BOUNCED INTO ROUGH"D-75"YARDS FROM HOLE."
126  D=D-75
127  GOTO 48
128  IF O<30 OR J>0 THEN 146
129  IF T>0 THEN 135
130  IF (S2+1)/15=INT((S2+1)/15) THEN 136
131  PRINT "YOU HOOKED-";
132  L[1]=L[3]
133  IF O>45 THEN 139
134  GOTO 41
135  IF (S2+1)/15=INT((S2+1)/15) THEN 131
136  PRINT "YOU SLICED-";
137  L[1]=L[2]
138  GOTO 133
139  PRINT "BADLY."
140  GOTO 41
141  L[1]=5
142  GOTO 41
143  L[1]=8
144  D2=INT(D2*3)
145  GOTO 169
146  L[1]=1
147  GOTO 41
148  D1=INT(.85*D1)
149  GOTO 92
150  IF L[1]>6 THEN 158
151  PRINT "YOUR SHOT WENT INTO WATER."
152  S1=S1+1
153  PRINT "PENALTY STROKE ASSESSED. HIT FROM PREVIOUS LOCATION."
154  J=J+1
155  L[1]=1
156  D=B
157  GOTO 48
158  PRINT "YOUR SHOT WENT OUT OF BOUNDS."
```

```
159  GOTO 152
160  IF T=3 THEN 163
161  D2=1+(3*INT((80/(40-H))*RND(0)))
162  GOTO 169
163  IF RND(0)>N THEN 167
164  N=N*.2
165  PRINT "SHOT DUBBED, STILL IN TRAP."
166  GOTO 48
167  N=.8
168  GOTO 161
169  PRINT "ON GREEN"D2"FEET FROM PIN. PUT POTENCY NUMBER ";
170  INPUT I
171  T7=0
172  GOSUB 1.E+11*I+1 OF 234
173  IF T7 THEN 169
174  S1=S1+1
175  IF S1+1-P>(H*.072)+2 THEN 182
176  IF K>2 THEN 182
177  K=K+1
178  IF T=4 THEN 188
179  D2=D2-I*(4+2*RND(0))+1.5
180  IF D2<-2 THEN 190
181  IF D2>2 THEN 185
182  PRINT "YOU H*O*L*E*D IT"
183  F=F+1
184  GOTO 24
185  PRINT "PUTT SHORT."
186  D2=INT(D2)
187  GOTO 169
188  D2=D2-I*(4+RND(0))+1
189  GOTO 180
190  PRINT "PASSED BY CUP."
191  D2=-D2
192  GOTO 186
193  READ D,P,L[2],L[3]
194  PRINT "READY TO GO ";
195  INPUT A$
196  IF A$="NO" THEN 208
197  T5=T5+P
198  PRINT "YOU ARE AT TEE OFF HOLE"F", DISTANCE"D"YARDS PAR"P
199  PRINT "ON YOUR RIGHT IS ";
200  X=2
201  GOSUB 217
202  PRINT "ON YOUR LEFT IS ";
203  GOSUB 217
204  GOTO 48
205  DATA 361,4,4,2,389,4,3,3,206,3,4,2,500,5,7,2,408,4,2,4,359,4,6,4
206  DATA 424,4,4,2,388,4,4,4,196,3,7,2,400,4,7,2,560,5,7,2,132,3,2,2
207  DATA 357,4,4,4,294,4,2,4,475,5,2,3,375,4,4,2,180,3,6,2,550,5,6,6
208  PRINT "TOTAL SCORE FOR"F-1"HOLES WAS"S2
209  IF S2-T5 THEN 212
210  PRINT "PAR GAME"
211  END
212  IF S2-T5>0 THEN 215
213  PRINT -(S2-T5)"UNDER PAR"
214  END
215  PRINT S2-T5"OVER PAR"
216  END
217  GOTO L[X] OF 222,224,226,228,230,232
218  PRINT "OUT OF BOUNDS."
219  RETURN
220  PRINT "PGA RULES HANDICAP=0-30"
221  GOTO 8
222  PRINT "FAIRWAY."
223  RETURN
224  PRINT "ROUGH."
225  RETURN
226  PRINT "TREES."
227  RETURN
228  PRINT "ADJACENT FAIRWAY."
229  RETURN
230  PRINT "TRAP."
231  RETURN
232  PRINT "WATER."
233  RETURN
234  PRINT "TO GET A DESCRIPTION OF CLUBS, ETC. TYPE 0 FOR A CLUB NO. WHEN REQUESTED"
235  T7=9
236  RETURN
237  END
```

CURFIT
A404-36038B

TITLE:

PERFORMS LEAST-SQUARES FIT

DESCRIPTION:

This program performs a least squares curve fit to the following functions:

1. Y = A + B(X)
2. Y = A exp (B * X)
3. Y = A (X^B)
4. Y = A + B/X
5. Y = 1/(A + B * X)
6. Y = X/(A + B * X)
7. Y = A + B * Log(X)

INSTRUCTIONS:

Before running the program enter the following data beginning in line 9900:

9900 DATA N

9901 DATA X_1, Y_1, X_2, Y_2...

—
—
—
—

99--DATA.....X_n, Y_n

Where: N = Number of Data Pairs

X_i, Y_i = the ith Data Pair

Where X_i is the independent variable and Y_i is the dependent variable.

The program will print summary data for the curve fits for the six functions and request the user to indicate which function he wishes detailed information about (Input a Ø, 1,2,3,4,5,6 or 7). A zero (Ø) will terminate the program.

SYSTEM SPECIFICATIONS:

2000A and Teletype

SPECIAL CONSIDERATIONS:

If there are more than 200 data pairs, change the dimension of variables X, Y, U, V in statement 9003 to this number.

If data is made up of multiple observations in the dependent variable for each independent variable, use MULTX, 36186, as a calling program and APPend CURFIT.

ACKNOWLEDGEMENTS:

Jerry L. Mulcahy
Raychem Corporation

STORE 17

RUN

```
CURFIT

9900  DATA 7
9901  DATA 8.32,12.78
9902  DATA 8.34,12.53
9903  DATA 8.36,12.08
9904  DATA 8.38,11.57
9905  DATA 8.4,11.19
9906  DATA 8.42,10.91
9907  DATA 8.44,10.73
9999  END
                LEAST SQUARES CURVES FIT

CURVE TYPE        INDEX  OF        A               B
                  DETERMINATION
1. Y=A+(B*X)      .979167          165.023         -18.2981
2. Y=A*EXP(B*X)   .981411          5.64762E+06     -1.56211
3. Y=A*(X↑B)      .937287          5.35430E+12     -12.6316
4. Y=A+(B/X)      .988257         -142.787          1294.44
5. Y=1/(A+B*X)    .985601         -1.03558          .133832
6. Y=X/(A+B*X)    .991327         -9.45113          1.21377
7. Y=A+B*LOG(X)   .935615          326.308         -148

MEAN AND STANDARD DEVIATION OF RAW DATA

                  MEAN            STANDARD DEVIATION
X                 8.38             4.32637E-02
Y                 11.6843          .800022

DETAILS FOR CURVE TYPE?4
   4. Y=A+(B/X) IS A HYPERBOLIC FUNCTION.  THE RESULTS
      OF A LEAST-SQUARES FIT OF ITS LINEAR TRANSFORM
      (SORTED IN ORDER OF ASCENDING VALUES OF X)
      ARE AS FOLLOWS:
X-ACTUAL          Y-ACTUAL          Y-CALC          PCT DIFFER

 8.32              12.78             12.7947         -.1
 8.34              12.53             12.4216                  .8
 8.36              12.08             12.0503                  .2
 8.38              11.57             11.6808         -.9
 8.4               11.19             11.313          -1
 8.42              10.91             10.9469         -.3
 8.44              10.73             10.5826                  1.3

DETAILS FOR CURVE TYPE?0

DONE
```

LISTING

```
CURFIT

8900  REM ***** HP TIME-SHARED BASIC PROGRAM LIBRARY *****
8901  REM
8902  REM          CURFIT: LEAST SQUARES CURVE FIT
8903  REM
8904  REM       36038    REV B       3/72
8905  REM
8906  REM  **** CONTRIBUTED PROGRAM ************************************
9000  REM  *****  CURFIT  *****  MATHEMATICS PROGRAM  *****
9001  REM  *****  VERSION 1  *****  7/31/69  *****
9002  REM  LEAST SQUARES CURVE FIT #1
9003  DIM X[200],Y[200],U[200],V[200],A[7],B[7],C[7],S[6],F[7]
9004  READ N
9005  PRINT
9006  FOR I=1 TO N
9007  READ X[I],Y[I]
9008  NEXT I
9009  PRINT
9010  MAT F=CON
9011  PRINT " ","LEAST SQUARES CURVES FIT"
9012  PRINT
9013  PRINT "CURVE TYPE","  INDEX  OF","   A","   B"
9014  PRINT " ","DETERMINATION"
9015  PRINT
```

```
9016  FOR I=1 TO 7
9017  MAT S=ZER
9018  GOSUB 9135
9019  IF (I-5)*(I-6)=0 THEN 9037
9020  IF (I-2)*(I-3)=0 THEN 9030
9021  FOR J=1 TO N
9022  LET V[J]=Y[J]
9023  GOSUB 9110
9024  NEXT J
9025  Y1=S[3]/N
9026  Y2=SQR((S[5]-(S[3]↑2/N))/(N-1))
9027  IF I=1 THEN 9047
9028  IF I=7 THEN 9052
9029  GOTO 9058
9030  FOR J=1 TO N
9031  IF Y[J] <= 0 THEN 9044
9032  LET V[J]=LOG(Y[J])
9033  GOSUB 9110
9034  NEXT J
9035  IF I=3 THEN 9052
9036  GOTO 9047
9037  FOR J=1 TO N
9038  IF Y[J]=0 THEN 9044
9039  LET V[J]=1/Y[J]
9040  GOSUB 9110
9041  NEXT J
9042  IF I=6 THEN 9058
9043  GOTO 9047
9044  PRINT "CAN'T FIT"
9045  LET F[I]=0
9046  GOTO 9065
9047  FOR J=1 TO N
9048  LET U[J]=X[J]
9049  GOSUB 9113
9050  NEXT J
9051  GOTO 9063
9052  FOR J=1 TO N
9053  IF X[J] <= 0 THEN 9044
9054  LET U[J]=LOG(X[J])
9055  GOSUB 9113
9056  NEXT J
9057  GOTO 9063
9058  FOR J=1 TO N
9059  IF X[J]=0 THEN 9044
9060  LET U[J]=1/X[J]
9061  GOSUB 9113
9062  NEXT J
9063  GOSUB 9182
9064  PRINT C[I],A[I],B[I]
9065  NEXT I
9066  PRINT
9067  PRINT
9068  PRINT "MEAN AND STANDARD DEVIATION OF RAW DATA"
9069  PRINT
9070  PRINT " "," MEAN ","STANDARD DEVIATION"
9071  PRINT "X",X1,X2
9072  PRINT "Y",Y1,Y2
9073  GOSUB 9120
9074  PRINT
9075  PRINT
9076  PRINT
9077  PRINT "DETAILS FOR CURVE TYPE";
9078  INPUT I
9079  IF I=0 THEN 9207
9080  LET K=I
9081  IF F[I]=1 THEN 9085
9082  GOSUB 9135
9083  PRINT " COULD NOT BE FIT."
9084  GOTO 9074
9085  GOSUB 9156
9086  IF (I-1)*(I-5)*(I-6)#0 THEN 9096
9087  FOR J=1 TO N
9088  LET Y=A[I]+B[I]*X[J]
9089  IF I=1 THEN 9093
9090  LET Y=1/Y
9091  IF I=5 THEN 9093
9092  LET Y=X[J]*Y
9093  GOSUB 9197
```

```
9094  NEXT J
9095  GOTO 9074
9096  FOR J=1 TO N
9097  IF I=2 THEN 9106
9098  IF I=3 THEN 9104
9099  IF I=4 THEN 9102
9100  LET Y=A[7]+B[7]*LOG(X[J])
9101  GOTO 9107
9102  LET Y=A[4]+B[4]/X[J]
9103  GOTO 9107
9104  LET Y=A[3]*(X[J]+B[3])
9105  GOTO 9107
9106  LET Y=A[2]*EXP(B[2]*X[J])
9107  GOSUB 9197
9108  NEXT J
9109  GOTO 9074
9110  LET S[5]=S[5]+V[J]+2
9111  LET S[3]=S[3]+V[J]
9112  RETURN
9113  LET S[1]=S[1]+U[J]
9114  LET S[2]=S[2]+U[J]+2
9115  IF I>1 THEN 9118
9116  X1=S[1]/N
9117  X2=SQR((S[2]-(S[1]+2/N))/(N-1))
9118  LET S[4]=S[4]+U[J]*V[J]
9119  RETURN
9120  FOR I=1 TO N-1
9121  LET M=I
9122  FOR J=I+1 TO N
9123  IF X[M] <= X[J] THEN 9125
9124  LET M=J
9125  NEXT J
9126  IF M=I THEN 9133
9127  LET P=X[M]
9128  LET Q=Y[M]
9129  LET X[M]=X[I]
9130  LET Y[M]=Y[I]
9131  LET X[I]=P
9132  LET Y[I]=Q
9133  NEXT I
9134  RETURN
9135  LET K=I
9136  IF K=1 THEN 9154
9137  IF K=2 THEN 9152
9138  IF K=3 THEN 9150
9139  IF K=4 THEN 9148
9140  IF K=5 THEN 9146
9141  IF K=6 THEN 9144
9142  PRINT "7. Y=A+B*LOG(X)";
9143  RETURN
9144  PRINT "6. Y=X/(A+B*X) ";
9145  RETURN
9146  PRINT "5. Y=1/(A+B*X) ";
9147  RETURN
9148  PRINT "4. Y=A+(B/X)",
9149  RETURN
9150  PRINT "3. Y=A*(X+B)",
9151  RETURN
9152  PRINT "2. Y=A*EXP(B*X)";
9153  RETURN
9154  PRINT "1. Y=A+(B*X)",
9155  RETURN
9156  PRINT "   ";
9157  GOSUB 9136
9158  PRINT " IS A";
9159  IF K=1 THEN 9165
9160  IF K=2 THEN 9167
9161  IF K=3 THEN 9169
9162  IF K=7 THEN 9171
9163  PRINT " HYPERBOLIC";
9164  GOTO 9172
9165  PRINT " LINEAR";
9166  GOTO 9172
9167  PRINT "N EXPONENTIAL";
9168  GOTO 9172
9169  PRINT " POWER";
```

```
9170  GOTO 9172
9171  PRINT " LOGARITHMIC";
9172  PRINT " FUNCTION.  THE RESULTS"
9173  IF K=1 THEN 9175
9174  PRINT "      OF A LEAST-SQUARES FIT OF ITS LINEAR TRANSFORM"
9175  PRINT "      (SORTED IN ORDER OF ASCENDING VALUES OF X)"
9176  PRINT "      ARE AS FOLLOWS:"
9177  PRINT
9178  PRINT "X-ACTUAL","Y-ACTUAL"," Y-CALC"," PCT DIFFER"
9179  PRINT
9180  RETURN
9181  PRINT
9182  LET B=(N*S[4]-S[1]*S[3])/(N*S[2]-(S[1]↑2))
9183  LET A=(S[3]-B*S[1])/N
9184  LET S1=S[5]-(S[3]↑2)/N
9185  LET S2=(B↑2)*(S[2]-(S[1]↑2)/N)
9186  LET C[I]=S2/S1
9187  IF (I-1)*(I-4)*(I-5)*(I-7)=0 THEN 9194
9188  IF (I-2)*(I-3)=0 THEN 9192
9189  LET A[6]=B
9190  LET B[6]=A
9191  RETURN
9192  LET A[I]=EXP(A)
9193  GOTO 9195
9194  LET A[I]=A
9195  LET B[I]=B
9196  RETURN
9197  PRINT X[J],Y[J],Y,
9198  LET D=Y[J]-Y
9199  LET D=.1*SGN(D)*INT(1000*ABS(D/Y))
9200  IF D<0 THEN 9205
9201  IF D>0 THEN 9204
9202  PRINT "      0"
9203  RETURN
9204  PRINT "       ";
9205  PRINT D
9206  RETURN
9207  STOP
9900  DATA 7
9901  DATA 8.32,12.78
9902  DATA 8.34,12.53
9903  DATA 8.36,12.08
9904  DATA 8.38,11.57
9905  DATA 8.4,11.19
9906  DATA 8.42,10.91
9907  DATA 8.44,10.73
9999  END
```