HEWLETT PACKARD

Computer Museum

2000 BASIC CONTRIBUTED LIBRARY HANDBOOK

ADDENDUM TO VOLUMES I, II, III, IV

MAY 1975

HEWLETT-PACKARD; CONTRIBUTED SOFTWARE CENTER

5303 Stevens Creek Blvd., Santa Clara, California 95050 Area Code 408 249-7020

36000-92001 **6/74**
Changed 8/76

August 1976

LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the most recent date on which the technical material on any given page was altered. If a page is simple re-arranged due to a technical change on a previous page, it is not listed as a changed page. Within the manual, changes are marked with a vertical bar in the margin.

Program Name	Page Number (s)	Effective Date
	COVER	Aug 1976
	INTRODUCTION	Aug 1976
	NEW PROGRAMS	Aug 1976
	REVISED PROGRAM	Aug 1976
	ALPHABETICAL INDEX	Aug 1976
	CROSS REFERENCE INDEX	Aug 1976
	VOLUME I CONTENTS	Aug 1976
FORM 2K	Pgs. 3-4	Aug 1976
PLOTXY	Pgs. 1-2	Aug 1976
TITLE	Pgs. 1-2	Aug 1976
	VOLUME II CONTENTS	Aug 1976
COFTAB	Pgs. 1-2, 7-10	Aug 1976
	VOLUME III CONTENTS	Aug 1976
INZOUT	Pgs. 1-2	Aug 1976
	VOLUME IV CONTENTS	Aug 1976
PTUD	Pgs. 1-5	Aug 1976
PTUE	Pgs. 1-4	Aug 1976
TRAN1	Pg. 1	Aug 1976
STABIL	Pgs. 1-2	Aug 1976
USECON	Pgs. 1-2	Aug 1976
PMAG	Pgs. 1-4	Aug 1976
MCMAN	Pgs. 1-2	Aug 1976
MUSIC	Pgs. 1-2	Aug 1976
INQUIR	Pg. 1	Aug 1976
TSAP	Pgs. 1-2	Aug 1976
ITEM	Pgs. 1-2	Aug 1976
SRP	Pgs. 1-2	Aug 1976

INTRODUCTION

This addendum to the HP BASIC Contributed Library includes documentation for all new and revised programs since the reprint of June 1974. There are 30 new packages and 18 revisions to existing library programs.

Please replace existing documentation with revised documentation and add the new documentation to each of your volumes.

INDEXES

Two indexes to the complete BASIC Library are also included:

Alphabetical Index
Cross Reference Index

The page describing Revised Programs should be helpful in updating your library software and documentation.

How to Order This Addendum

Contact your local HP sales office to order this addendum:

HP 36000-92001 2000 BASIC Contributed Library Handbook Addendum

How to Order Library Software

All of the programs (New & Revised documented in this addendum and which operate on the 2000 Series System are available on the single magnetic tape, part number HP 36600A-10001 or 36600A-11001, priced at $133. This Mag tape contains all of the software for the entire library, volumes 1-5 and this addendum. The programs for this addendum is stored in account Z906 and C906.

DIRECT MAIL

Customers in the U.S. are encouraged to order Contributed Library software and manuals via Direct Mail by sending check or money order to Hewlett-Packard Co., Mail Order Department, P.O. Drawer #20, Mountain View, Ca. 94043. Direct Mail Orders must include $1.50. A Direct Mail Order Form is included in this addendum. Contact your local HP sales office for prices and assistance in placing orders.

NEW PROGRAMS IN ADDENDUM DATED MAY 1975

NAME	TITLE	VOLUME
ATG	COMPUTER ORIENTED ACCOUNTING - INTERACTIVE VERSION	4
BASP	BASIC ANALOG SIMULATION PROGRAM	4
BINO	BINOMIAL FUNCTION EXPANSION	2
COFTAB	CONVERSATIONAL FREQUENCY AND CROSS TABULATOR	2
EXTADD	40 DIGIT PRECISION STRING ADDITION	2
FDUMP	LISTS FILES, TOTAL RECORDS, INDICATES STRINGS & NUMERICS	1
FORM2K	TEXT FORMATTER	1
INQUIR	SOCIAL SCIENCE INQUIRY PACKAGE	4
LENGTH	CWF SUBPROGRAM	4
LOGICK	BOOLEAN ALGEBRA EVALUATOR	2
MCMAN	CARDIOVASCULAR SIMULATION	4
MUSIC	TWELVE TONE COMPOSITION PROGRAM	4
ORG5	TEACHES NOMENCLATURE OF ALKYL HALIDES AND ALCOHOLS	4
PMAG	STELLAR MAGNITUDES	4
PTUD	PILOT TUTORIAL	4
PTUE	PILOT TUTORIAL	4
PLOTXY	TWO VARIABLE PLOT PROGRAM	1
POLISH	POLISH CONVERSION	4
POWER2	POWERS OF TWO TABLES	2
PYRMID	BUILD A PYRAMID	4
SCOREF	COMPUTES MEAN, STANDARD DEVIATION AND STANDARD SCORES FOR TEST SCORES	2
SIMLAT	SIMULATION OF INVESTMENT RETURNS WITH SENSITIVITY ANALYSIS	3
SIMPLX	SOLVES LINEAR PROGRAM (CONDENSED TABLEAU METHOD)	2
STABIL	FEDERAL GOVERNMENT STABILIZATION POLICY	4
STAT1	HISTOGRAM, STANDARD DEVIATION & PLOT OF A SET OF NUMBERS	2
TITER	SIMULATES TITRATION OF A BASE BY AN ACID	4
TRAN1	GRAPHICS UTILITY PROGRAMS	4
TRIVIA	COMMON NAME QUIZ FOR CHEMISTRY STUDENTS	4
UHCX	COMPLEX NUMBER CALCULATOR	2
UNLETH	UNILETH STATISTICS PACKAGE	2
USECON	TESTING ECONOMIC HYPOTHESES	4

REVISED PROGRAMS IN ADDENDUM DATED MAY 1975

NAME TITLE	VOLUME	REASON FOR REVISION
ACNODE:AC CIRCUIT ANALYSIS PROGRAM	2	*Documentation change only.* *Page 3 "X=REAL PART OF G_m(mhos)"*
ANVAR1:ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN	2	*Documentation change only. The variable A is calculated on line 9012 and does not have to be entered.*
BITEST:BINOMIAL PROPORTION	2	*Corrects one line.* *9410 PRINT "9105 LETG=5";G5+1*
CHEM1 :CAI IN CHEMISTRY	4	*FILES statement corrected for use in A000 ID.*
DECSN :TOP MANAGEMENT DECISION GAME	3	*Documentation only updated to be compatible with 2000F system.*
EDIT2K:TEXT EDITOR FOR THE HP 2000C/2000C'/F	1	*New corrected version of software.*
EXTPRE:40-DIGIT PRECISION MATHEMATICS	2	*Improved, faster version of software.*
FINDIT:INFORMATION RETRIEVAL SYSTEM	1	*1 line corrected in "SEARCH" program.* *1203 IF NOT J(K) THEN 1210*
FRQ :FREQUENCY BETWEEN BOUNDARIES	2	*Corrects 4 lines.* *1175 IF M=M2 THEN 1160* *1535 LET M(1)=M2* *2195 LET D$="XX"* *865 IF D$="XX" THEN 980*
GDPA :EFficient 'CORNER' PORTFOLIOS	3	*Adds Program GDPA2 to software.*
GSIMEQ:SIMULTANEOUS LINEAR EQUATIONS	2	*Line 343 added to program.* *343 PRINT "DO NOT BEGIN AN EQUATION WITH A MINUS SIGN".*
HELLO :TYPES DATE, TIME, AND PORT NUMBER ON TERMINAL	1	*Shortened version of software with faster response time during log-on.*
INZOUT:INPUT/OUTPUT ANALYSIS ON ECONOMIC FLOWS	3	*4 lines corrected in program.* *9432 LET J=I* *9440 LET J=I9* *9445 LET D(J)=D* *9450 LET T(J)=D(J)/D(I9)*
MORGAG:MORTGAGE ANALYSIS	3	*Corrects 1 line.* *630 R1=R*
MULTX :LEAST-SQUARES FIT, MULTIPLE Y'S PER X	2	*Corrects 1 line.* *8012 GOTO 9022*
SNOBOL:SNOBOL3 COMPILER FOR HP 2000C SYSTEM	4	*1 line corrected in program "SNOBOL".* *9020 CHAIN $SNOBER*
STAT16:COMPUTES AN ANALYSIS OF VARIANCE TABLE AND F-RATIOS	2	*Corrects 1 line. Change line 9891 to 9904.*
TITLE :CHARACTER GENERATION	1	*Program 'CHRGEN' added to generate the file "CHAR".*

NUMERICAL INDEX

PROGRAM NUMBER	NAME	TITLE	SYSTEM	SOFTWARE DATE	DOCUMENT DATE
36001A	LOGRAM:	LOG-ON TAPE ANALYZER	F	12/70	6/74
36003B	MACRO :	A TEXT AND FILE PROCESSING SYSTEM	F	10/73	6/74
36005B	FORMAT:	ALLOWS SPECIAL FORMATTING OF DATA PRINTOUT	E	2/73	6/74
36006A	FILMAN:	FILE MANAGER	F	12/70	6/74
36007A	FILIN :	KEYBOARD FILE LOADING PROGRAM	E	12/70	6/74
36008C	FILDUM:	PAPER TAPE FILE DUMP	E	2/73	6/74
36009D	FILIST:	LISTS FILE CONTENTS BY RECORD NUMBER	E	6/74	6/74
36010C	FILOAD:	LOADS A FILE FROM THE TELETYPE	E	2/73	6/74
36011A	FILREA:	REENTERS THE DATA TAPE DUMPED BY FILDUM	E	12/70	6/74
36012B	FLCOPY:	COPIES ONE FILE INTO ANOTHER	E	3/72	6/74
36017A	FNCTS :	COMPUTES TRIG FUNCTIONS FOR COMPLEX ARGUMENTS	E	12/70	6/74
36019A	BESSEL:	CALCULATES BESSEL FUNCTION OF FIRST KIND	E	12/70	6/74
36022B	ROMINT:	INTEGRATES A FUNCTION (ROMBERG METHOD)	E	6/73	6/74
36023B	POLFIT:	FITS LEAST-SQUARES POLYNOMIALS	F	3/72	6/74
36024A	ROOTER:	FINDS THE ROOTS OF POLYNOMIALS	E	12/70	6/74
36025A	CDETER:	COMPUTES VALUE OF COMPLEX DETERMINANT	E	12/70	6/74
36027A	CROUT1:	SOLVES SIMULTANEOUS LINEAR EQUATIONS	E	12/70	6/74
36028A	CTRFFT:	COMPLEX TO REAL FAST FOURIER TRANSFORM	F	12/70	6/74
36029B	RTCFFT:	REAL TO COMPLEX FAST FOURIER TRANSFORM	F	6/74	6/74
36030A	GFFT :	GENERAL FAST FOURIER TRANSFORM	F	12/70	6/74
36032A	DE-1OR:	1ST ORDER DIFFERENTIAL EQUATION	E	12/70	6/74
36033A	DE-2OR:	2ND ORDER DIFFERENTIAL EQUATION	E	12/70	6/74
36034A	SPHERE:	SOLVES SPHERICAL TRIANGLES	E	12/70	6/74
36037A	FACTOR:	FINDS PRIME FACTORS OF POSITIVE INTEGERS	E	12/70	6/74
36038D	CURFIT:	PERFORMS LEAST SQUARES FIT	E	6/74	6/74
36039B	ANVAR1:	ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN	E	9/71	3/75
36040B	ANVAR2:	ANALYSIS OF VARIANCE (LATIN SQUARE DESIGN)	E	9/71	6/74
36041A	BINOPO:	PROBABILITY DISTRIBUTION COMPARISONS	E	12/70	6/74
36042A	CHISQ :	COMPUTES PROBABILITY OF CHI-SQUARE VALUES	E	12/70	6/74
36043B	CHISQS:	CHI-SQUARE STATISTICS FOR M*N CONTINGENCY TABLES	E	3/72	6/74
36045A	GEOMEN:	STATISTICS OF GEOMETRIC DISTRIBUTION	E	12/70	6/74
36052A	STAT2 :	MANN-WHITNEY 2 SAMPLE RANK TEST	E	12/70	6/74
36053A	STAT3 :	SPEARMAN RANK CORRELATION COEFFICIENTS	E	12/70	6/74
36054B	REGCOR:	REGRESSION/CORRELATION	E	2/73	6/74
36055B	HISTOG:	A HISTOGRAM FORMED FROM A SET OF NUMBERS	E	3/72	6/74
36056A	ANALAD:	CIRCUIT ANALYSIS	F	12/70	6/74
36057A	ACNODE:	AC CIRCUIT ANALYSIS PROGRAM	F	12/70	3/75
36058A	HTXFR :	TWO DIMENSIONAL HEAT TRANSFER	F	12/70	6/74
36059A	DEBYE :	COMPUTES DEBYE OR EINSTEIN FUNCTION	E	12/70	6/74
36060A	LPFLTR:	DESIGNS LOW-PASS FILTERS	E	12/70	6/74
36062A	MICRO :	MICROWAVE PARAMETERS CONVERSION	E	12/70	6/74
36064A	MIXSPR:	MIXER SPURIOUS RESPONSE PROGRAM	E	12/70	6/74
36065C	DECSN :	TOP MANAGEMENT DECISION GAME	E	2/73	3/75
36067A	DYNPRO:	DYNAMIC PROGRAMMING MODEL	F	12/70	6/74
36068C	LINPRO:	LINEAR PROGRAMMING MODEL	F	6/74	6/74
36069A	LNTRND:	LINEAR TREND FORECASTING	E	12/70	6/74
36072A	SALSIM:	SALARY SCHEDULE COST SIMULATOR	F	12/70	6/74
36073A	BUDGET:	DEPARTMENTAL MANAGER'S BUDGETING PROGRAM	F	12/70	6/74
36074A	ANNUIT:	ANNUITY ANALYSIS	E	12/70	6/74
36075A	BALSHT:	PROFORMA INCOME STATEMENT AND BALANCE SHEET	F	12/70	6/74
36076B	BNDPRC:	BOND PRICE ANALYSIS	E	3/72	6/74
36077A	BNDSWH:	BOND SWITCH ANALYSIS	F	12/70	6/74
36078B	BNDYLD:	BOND YIELD ANALYSIS	E	3/72	6/74
36079A	BNKRSV:	BANK RESERVE CALCULATIONS	E	12/70	6/74
36080A	CAPINV:	CAPITAL INVESTMENT ANALYSIS	E	12/70	6/74
36082A	DEPCOM:	DEPRECIATION METHOD COMPARISON	E	12/70	6/74
36083A	EQUITY:	COST OF EQUITY CAPITAL	E	12/70	6/74
36084A	EXDRSK:	EXTENDED RISK ANALYSIS	E	12/70	6/74
36085B	EXSMOO:	EXPONENTIAL SMOOTHING ON PRICE DATA	E	12/70	6/74
36086A	GNPSUM:	GROSS NATIONAL PRODUCT SUMMARY	E	12/70	6/74
36087A	INACNT:	NATIONAL INCOME & PRODUCT ACCOUNTS	E	12/70	6/74
36088B	IN/OUT:	INPUT/OUTPUT ANALYSIS ON ECONOMIC FLOWS	E	3/75	3/75
36089B	INSTMT:	INCOME STATEMENT	E	2/73	6/74
36090A	LENOER:	SIMPLE LOAN ANALYSIS	E	12/70	6/74
36091B	LESSEE:	LEASE ANALYSIS AS DETERMINED BY THE LESSEE	E	6/74	6/74
36092B	MARKOV:	SECURITIES PORTFOLIO USING MARKOWITZ MODEL	E	10/73	6/74
36093A	MKBUY :	MAKE-BUY DECISION ANALYSIS	E	12/70	6/74
36094C	MORGAG:	MORTGAGE ANALYSIS	E	3/75	6/74
36095A	SALES :	SALES COMMISSION REPORT	E	12/70	6/74
36096A	STKINC:	STOCK MERGER INCENTIVE PROGRAM	E	12/70	6/74
36098A	STKRTN:	STOCK RETURNS REPORT	E	12/70	6/74
36099A	STKSMO:	EXPONENTIAL SMOOTHING AS A STOCK GUIDE	E	12/70	6/74
36100A	STKVAL:	STOCK VALUE & EVALUATION REPORT	E	12/70	6/74
36101A	TRUINT:	TRUE ANNUAL INTEREST RATE ANALYSIS	E	12/70	6/74
36102A	GOLF :	A GOLF GAME	E	12/70	6/74
36103A	BLJACK:	GAME OF TWENTY-ONE	E	12/70	6/74
36104B	PLOT :	PLOTS A GIVEN FUNCTION ON THE TELETYPE	E	10/73	6/74
36105B	BATNUM:	MAKE UP YOUR OWN 'NIM' GAME, THEN PLAY AGAINST THE COMPUTER	E	12/74	6/74
36106B	QUBIC5:	THREE-D TIC-TAC-TOE...COMPUTER HAS 5 LEVELS OF STRATEGIES	F	12/74	6/74
36107A	SLOT :	SIMULATES A SLOT MACHINE (USE IN NEVADA ONLY)	E	12/70	6/74
36109A	BEMDES:	RECOMMENDS CORRECT STEEL BEAM USE	E	12/70	6/74
36110A	CANNON:	HP 7200A GAME DEMO FOR TIME-SHARED BASIC	F	12/70	6/74
36111B	EXPEND:	BUDGET EXPENDITURES VS. TARGETS MONITOR	F	10/73	6/74
36112A	FPLOT :	FUNCTION PLOT	E	12/70	6/74
36119A	CXEXP :	VECTOR EXPONENTIATION	E	7/71	6/74
36120A	FC :	ANALYSIS OF LOG TAPE	E	3/72	6/74
36121A	ORBIT :	INTEGRATES EQUATIONS OF MOTION	E	8/71	6/74
36122A	SORT :	FILE SERIAL STRING SORT	F	3/72	6/74
36114C	TITLE :	CHARACTER GENERATION	F	3/75	3/75
36115A	GRAPHS:	DEMO PLOT PROGRAM FOR HP 7200 PLOTTER	E	12/70	6/74
36118B	CXARTH:	VECTOR ARITHMETIC	E	6/74	6/74
36125C	HELLO :	TYPES DATE, TIME, AND PORT NUMBER ON TERMINAL	F	3/75	6/74
36128A	TENS :	DRILL IN MULTIPLICATION OF NUMBERS WITH TRAILING ZEROS	F	7/71	6/74
36129A	DRILL :	BASIC ARITHMETIC DRILL	F	7/71	6/74
36130A	TWQUES:	MATHEMATICAL GAME OF TWENTY QUESTIONS	E	7/71	6/74
36131A	CALCOM:	CALCULATOR PROGRAM WITH OPTIONAL PLOTTER OUTPUT	F	7/71	6/74
36133A	WKSHT :	GENERATES MATH WORKSHEETS	E	7/71	6/74
36136A	SCORES:	COMPUTES MEAN, STANDARD DEVIATION AND STANDARD SCORES FOR TEST SCORES	E	7/71	6/74
36137A	KR20 :	ITEM ANALYSIS AND KUDER-RICHARDSON FORMULA 20 RELIABILITY	E	7/71	6/74
36138B	LIFE :	JOHN CONWAY'S CELLULAR BOARD GAME	F	12/74	6/74
36139A	TUTOR :	TUTOR SERIES BASIC LANGUAGE PROGRAMMING COURSE, TUT01 - TUT25	E	7/71	6/74
36142B	CSHFL :	CASH FLOW ANALYSIS	E	2/73	6/74
36143C	XREF :	BASIC LANGUAGE PROGRAM CROSS-REFERENCE GENERATOR	F	2/73	6/74
36144B	EXTPRE:	40-DIGIT PRECISION MATHEMATICS	E	3/75	3/75
36145A	STGSRT:	SORTS STRINGS FROM FILES	F	7/71	6/74
36148A	EXPNTL:	CAI IN SIMPLE EXPONENTIAL FUNCTIONS OF TIME	E	7/71	6/74
36149A	SOLVER:	SOLVES COMPLEX SIMULTANEOUS EQUATIONS	F	7/71	6/74
36152A	AC-1 :	COMPUTER AIDED PRACTICE IN EE AC ANALYSIS	E	7/71	6/74
36153A	COMPLX:	CAI IN ALGEBRA OF COMPLEX NUMBERS	E	7/71	6/74
36154A	INDMTR:	ANALYSIS OF A BALANCED POLYPHASE INDUCTION MOTOR	E	7/71	6/74
36155A	POLAR :	PLOTS SINGLE VARIABLE IN POLAR FORM	E	7/71	6/74
36156A	DVDRS :	CAI IN VOLTAGE AND CURRENT DIVIDERS	E	7/71	6/74
36157A	UNITS :	CAI IN INTERPRETATION OF EE UNITS	E	7/71	6/74
36159A	DIFFEQ:	CAI IN SOLUTION OF LINEAR FIRST-ORDER DIFFERENTIAL EQUATIONS	E	7/71	6/74
36164B	IATA.I:	CALCULATE AIR FREIGHT RATES	F	3/72	6/74
36165A	FGRAPH:	SIMULTANEOUS FUNCTION GRAPHER	E	3/72	6/74
36167A	JACOBI:	EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX	E	7/71	6/74
36170A	T-TEST:	TEST OF HYPOTHESES USING STUDENTS T DISTRIBUTION	E	7/71	6/74
36171B	CPATH :	CRITICAL PATH EVALUATION	F	3/72	6/74
36172A	ANVAR3:	ANALYSIS OF VARIANCE FOR A TWO VARIABLES OF CLASSIFICATION DESIGN	E	9/71	6/74
36173A	ANVAR4:	TWO-WAY ANALYSIS OF VARIANCE FOR A TWO-WAY EXPERIMENT	E	9/71	6/74
36174A	TRCK.I:	CALCULATE TRUCK FREIGHT RATES	F	8/71	6/74
36175A	CARLOS:	COMPUTER-ASSISTED REVIEW LESSONS ON SYNTAX FOR SPANISH II	F	10/72	6/74
36176A	STGINT:	STRING-INTEGER CONVERSIONS	E	3/72	6/74
36177A	REP :	DATA CENTER INVENTORY REPORT GENERATOR	F	3/72	6/74
36178A	MULREG:	MULTIPLE REGRESSION/CORRELATION	F	9/71	6/74
36179A	DROIPB:	DISCOUNTED RETURN ON INVESTMENT AND PAYBACK	E	3/72	6/74
36180A	SUNSET:	SUNRISE-SUNSET PREDICTOR	E	3/72	6/74
36181A	FTBL :	FOOTBALL	E	3/72	6/74
36182A	SAT :	TRIGONOMETRIC SOLUTIONS OF TRIANGLES	E	3/72	6/74
36184A	SPCTRA:	OPTICAL ABSORPTION SPECTRA SIMULATION, 2-SPECIES EQUILIBRIUM MIXTURES	E	3/72	6/74
36186B	MULTX :	LEAST-SQUARES FIT, MULTIPLE Y'S PER X	E	3/75	6/74
36187B	WHEELS:	AUTO PURCHASE AND MAINTENANCE SIMULATION	F	10/73	6/74
36188A	POLY :	POLYNOMIAL APPROXIMATION	E	3/72	6/74
36191C	FRQ :	FREQUENCY BETWEEN BOUNDRIES	F	3/75	6/74
36194A	LEASIN:	LEASE INCOME	E	3/72	6/74
36196A	SOLVIT:	SIMULTANEOUS LINEAR EQUATIONS USING GAUSSIAN REDUCTION	E	3/72	6/74
36197A	JULIAN:	JULIAN CALENDAR FOR THE CURRENT YEAR	E	3/72	6/74
36199A	MRKSIM:	MARKETING SIMULATION	F	2/73	6/74
36202A	ATTEND:	ABSENTEE LISTING	F	10/72	6/74
36204B	TIDEX :	SYMBOLIC FILE EDITOR	F	10/73	6/74
36205A	CADAI1:	COMPUTER ASSISTED ARITHMETIC DRILL	E	3/72	6/74
36208B	STOP :	COMPUTER LITERACY - PSEUDO-MACHINE LANGUAGE	E	10/73	6/74
36209B	SETUPC:	MANAGEMENT SIMULATION GAME FOR THE 2000C	F	10/73	6/74
36210B	CTC1 :	CTC MANUFACTURING PARTS CONTROL	F	2/73	6/74
36211B	CTC2 :	CTC INVENTORY CONTROL FOR FINISHED PRODUCTS	F	2/73	6/74
36212B	CTC3 :	CTC PROJECTION PROGRAMS	F	6/73	6/74
36213B	CTC4 :	CTC PAYROLL PROGRAMS	F	6/73	6/74
36214B	CTC5 :	CTC ACCOUNTS RECEIVABLE	F	2/73	6/74
36218A	HPMLIT:	LIST/DUMP HP ASSEMBLER FILES	F	3/72	6/74
36219A	BRAIN :	BRAIN SIMULATOR PROGRAM	E	10/72	6/74
36220A	CHARS :	ASCII CHARACTER SET	F	10/72	6/74
36221B	XTRACT:	MANUAL/TAPE FILE LOADER AND DUMP PROGRAMS	F	6/74	6/74
36222A	TALK :	TIME SHARING SYSTEM COMMUNICATION	F	10/72	6/74
36226A	LOAN :	LOAN AMORTIZATION	F	10/72	6/74
36228B	PLOTWD:	WORD PLOTTER	E	10/73	6/74
36230A	TRANSP:	TRANSPORTATION PROBLEM	F	10/72	6/74
36231A	ADDRES:	ADDRESS LABELS	E	10/72	6/74
36232A	IRV :	FILE SORT ROUTINE	F	2/73	6/74
36233A	LABOR :	LABOR/MANAGEMENT BARGAINING	F	10/72	6/74

PROGRAM NUMBER	NAME	TITLE	SYSTEM	SOFTWARE DATE	DOCUMENT DATE
36235A	HISS :	SAMPLE STATISTICS AND HISTOGRAM FORMED FROM A SET OF NUMBERS	F	2/73	6/74
36237A	TRIFAC:	FACTORING QUADRATIC TRINOMIALS	E	10/72	6/74
36238A	SIPRAC:	PRACTICE WITH SIGNED NUMBERS	F	10/72	6/74
36239A	POLSUB:	POLYNOMIAL SUBTRACTION	E	10/72	6/74
36240A	BISQAR:	SQUARING BINOMIALS	E	10/72	6/74
36241B	IATA.C:	CALCULATES BREAKPOINT OF IATA CONTAINERS	F	6/74	6/74
36243B	STTRI :	STAR TREK	F	10/73	6/74
36244A	KENO :	$25,000.00 KENO GAME	F	2/73	6/74
36246A	POLFTE:	FITS LEAST-SQUARES POLYNOMIALS	E	2/73	6/74
36247A	FILRPT:	REPORTS FILE CONTENTS AND STRUCTURE	F	6/73	6/74
36249A	PSQUAR:	PATTERN SQUARES FOR HP 7200A PLOTTER	F	6/74	6/74
36250D	FINDIT:	INFORMATION RETRIEVAL SYSTEM	F	3/75	6/74
36251B	LOGIC :	LOGIC EXAMINATION PROGRAMS	E	10/73	6/74
36252A	DBLFIT:	LEAST SQUARES FIT TO POINTS WITH UNCERTAINTIES IN BOTH VARIABLES	E	10/73	6/74
36253A	EXTEND:	INFINITE PRECISION MATH UTILITY PROGRAM	F	2/73	6/74
36256B	ASCII*:	CREATES AN ASCII FILE CONTAINING ALL 256 ASCII CHARACTERS	F	6/73	6/74
36257A	-ASCII:	ASCII CODE GENERATOR	F	2/73	6/74
36258A	KEYSIG:	GIVES MAJOR SCALES	E	2/73	6/74
36259A	MUSIC5:	FINDS DOMINANT SEVENTHS	E	2/73	6/74
36260A	IONIC :	DRILL ON FORMULAS AND CHARGES OF IONS	F	2/73	6/74
36261A	ISOMER:	DRILL ON NAMING ALKANES	F	2/73	6/74
36262A	CXSYSS:	SOLVING SYSTEMS OF COMPLEX LINEAR EQUATIONS	E	2/73	6/74
36263A	DETER4:	DETERMINANTS, CHARACTERISTIC POLYNOMIALS AND INVERSES OF MATRICES	F	2/73	6/74
36264A	RNDORD:	PLACING INTEGERS IN RANDOM ORDER	E	2/73	6/74
36265A	GAME :	CLASSIC MATRIX OF GAME THEORY	F	2/73	6/74
36266A	IONIC1:	DRILL ON FORMULAS OF IONIC COMPOUNDS	F	2/73	6/74
36267A	ELMENT:	DRILL ON SYMBOLS FOR CHEMICAL ELEMENTS	F	2/73	6/74
36271A	ANOVA3:	THREE FACTORIAL ANALYSIS OF VARIANCE	F	6/73	6/74
36272A	FILIS :	FILE LISTING PROGRAM	F	2/73	6/74
36276A	MUSIC2:	TRIAD SOLVING PROGRAM	E	2/73	6/74
36277A	DIMIS :	X-Y AXIS SEGMENT PROGRAM	F	2/73	6/74
36278A	SYSSOL:	SOLVING SYSTEMS OF LINEAR EQUATIONS	F	6/73	6/74
36282A	COLREG:	COLLEGE REGISTRATION DEMO	F	6/73	6/74
36284A	MESSAG:	INTERTERMINAL COMMUNICATOR	F	6/73	6/74
36287A	DATA :	DUMPS FILE TO DATA STATEMENTS	F	6/73	6/74
36288A	CALNDR:	PRINTS A CALENDAR	F	6/73	6/74
36293A	ACTFIL:	ACTIVE FILTER DESIGN	F	6/73	6/74
36294A	ANCOV :	ANALYSIS OF COVARIANCE	E	6/73	6/74
36295B	CHEM :	SELF-CORRECTING CHEMISTRY TEST	E	10/73	6/74
36296B	ALFTOV:	ALPHA TO VARIABLE CONVERSION	E	10/73	6/74
36297B	TIMER :	TIME OF THE DAY	E	10/73	6/74
36298B	DATER :	DATE AND DAY OF THE WEEK	E	10/73	6/74
36299A	PRINT :	GENERATES LARGE LETTERS	F	6/73	6/74
36300A	DROS :	GENETIC CHARACTERISTICS	E	12/71	6/74
36301A	EVOLU :	NATURAL SELECTION EXPERIMENT	E	12/71	6/74
36302A	GAMGN :	PROCESS OF GAMETOGENESIS	E	12/71	6/74
36303A	NZYMC :	ENZYMATIC REACTION RATES	E	12/71	6/74
36304A	NZYM2 :	ENZYME REACTION RATE	E	12/71	6/74
36305A	PHOSYN:	PHOTOSYNTHESIS EXPERIMENT	E	12/71	6/74
36306A	DECAY1:	RADIOACTIVE DECAY GAME	E	12/71	6/74
36307A	DECAY2:	NUCLEAR DECAY	E	12/71	6/74
36308A	EQUILI:	EQUILIBRIUM SYSTEMS	E	12/71	6/74
36309A	KINET :	KINETIC REACTION	E	12/71	6/74
36310A	MASSD :	MASS DEFECT	E	12/71	6/74
36311A	PRCNT :	PERCENT COMPOSITION	E	12/71	6/74
36312A	BFIELD:	MAGNETIC FIELD PICTURE	E	12/71	6/74
36313A	BOHR :	ENERGY LEVEL DIAGRAM	E	12/71	6/74
36314A	CALORI:	CALORIMETRY EXPERIMENT	E	12/71	6/74
36315A	KINERV:	REVIEW OF KINEMATICS	E	12/71	6/74
36316A	NEWTN2:	NEWTON'S 2ND LAW	E	12/71	6/74
36317A	PHOTEL:	PHOTOELECTRIC EFFECT	E	12/71	6/74
36318A	PHOTON:	ENERGY LEVEL PROBLEM	E	12/71	6/74
36319A	REFLCT:	LEAST TIME PRINCIPLE AND LIGHT.	E	12/71	6/74
36320A	SNELL :	SNELL'S LAW	E	12/71	6/74
36321A	SPACE :	SPACECRAFT ORBITS	E	12/71	6/74
36322A	VFIELD:	POTENTIAL FIELD PICTURE	E	12/71	6/74
36323A	VLOCTY:	INSTANTANEOUS VELOCITY	E	12/71	6/74
36324B	WAVES :	SUM OF TWO WAVES	E	3/72	6/74
36325A	CLOUDS:	CLOUD FORMATION	E	12/71	6/74
36326A	WATER1:	WATER BUDGET PROBLEM CHECK	E	12/71	6/74
36327A	WATER2:	WATER BUDGET	E	12/71	6/74
36328A	BALANC:	TRADE AND PAYMENT BALANCES	E	12/71	6/74
36329B	BANK :	SOLVES FINANCIAL PROBLEMS	E	10/73	6/74
36330A	CONSMP:	DEPRESSION/EQUILIBRIUM	E	12/71	6/74
36331A	STOCK :	STOCK MARKET SIMULATION	E	12/71	6/74
36333A	CRVLEN:	COMPUTES LENGTH OF ANY CURVE	E	12/71	6/74
36335A	QUADT :	NATURE OF GRAPH OF A 2ND ORDER EQUATION IN TWO VARIABLES	E	12/71	6/74
36336A	RATIO :	SOLVES PROPORTIONS	E	12/71	6/74
36337A	SETS :	UNION AND INTERSECTION OF SETS	E	12/71	6/74
36338A	SURFAC:	AREA OF SURFACE OF REVOLUTION	E	12/71	6/74
36339A	VOLSOL:	VOLUME OF SOLID OF REVOLUTION	E	12/71	6/74
36340B	AVERG1:	AVERAGES AND CURVES GRADES	E	6/74	6/74
36341A	FREQ :	BAR GRAPHS OF GRADES	E	12/71	6/74
36342A	GRADE :	TEST GRADE FOR NUMBER OF QUESTIONS MISSED	E	12/71	6/74
36343A	ITEM1 :	NUMBER FREQUENCIES	E	12/71	6/74
36344A	STAT :	STATISTICAL ANALYSIS OF LAB DATA	E	12/71	6/74
36350A	PLANK :	A PHOTOELECTRIC SIMULATION	E	12/71	6/74
36351B	SLITS :	YOUNG'S DOUBLE SLIT EXPERIMENT	E	10/73	6/74
36360A	BAGELS:	THREE-DIGIT NUMBER GUESSING	F	12/71	6/74
36501A	GANOVA:	ANALYSIS OF VARIANCE (2-WAY)	F	10/72	6/74
36502A	GSPMG :	STANFORD PORTFOLIO MANAGEMENT GAME	F	10/72	6/74
36503A	GCHLIN:	RATING INVESTMENT FUNDS	F	10/72	6/74
36504A	GCPATH:	CRITICAL PATH ANALYSIS	F	10/72	6/74
36505A	GCPMI :	CRITICAL PATH ANALYSIS	F	10/72	6/74
36507A	GDAPI :	ABNORMAL PERFORMANCE INDEX	F	10/72	6/74
36508B	GDPA :	EFFICIENT 'CORNER' PORTFOLIOS	F	3/75	6/74
36510A	GFNRAT:	FINANCIAL RATIOS	F	10/72	6/74
36512A	GINTLP:	LINEAR PROGRAMMING-VARIABLES RESTRICTED TO VALUES OF ONE OR ZERO	F	10/72	6/74
36513A	GIRRPV:	INVESTMENT RETURN (CASH FLOW)	F	10/72	6/74
36514A	GKASSF:	WARRANT PRICE CALCULATION	F	10/72	6/74
36515A	GKCOST:	PRICE/EARNINGS RATIO CALCULATION	F	10/72	6/74
36516A	GLP :	LINEAR PROGRAMMING	F	10/72	6/74
36517A	GLPSA1:	LINEAR PROGRAMMING TWO-PHASE SIMPLEX METHOD	F	10/72	6/74
36518A	GMARKS:	GSB GRADING PROGRAM	F	10/72	6/74
36519A	GMCRO1:	FISCAL POLICY GAME	F	10/72	6/74
36521A	GMCROS:	ECONOMIC POLICY GAME	F	10/72	6/74
36529A	GNETFL:	NETWORK FLOW	F	10/72	6/74
36530A	GNMRVB:	SECURITIES PORTFOLIO ANALYSIS AND DETERMINATION	F	10/72	6/74
36541A	GRANK :	RANKING STATISTICS	F	10/72	6/74
36542A	GRGPLT:	SIMPLE REGRESSION AND PLOT	F	10/72	6/74
36543A	GRISKA:	RISK ANALYSIS IN CAPITAL INVESTMENT	F	10/72	6/74
36545A	GSTKVL:	STOCK VALUATION	F	10/72	6/74
36547B	GSIMEQ:	SIMULTANEOUS LINEAR EQUATIONS	F	3/75	3/75
36548A	GTAPID:	PAPER TAPE TITLER	F	10/72	6/74
36549A	GTASPD:	SUBJECTIVE PROBABILITY DISTRIBUTION	F	10/72	6/74
36550A	GVOTE :	COMMITTEE CHOICE ANALYSIS	F	10/72	6/74
36551A	GWBULL:	SUBJECTIVE PROBABILITY - RANDOM VALUES	F	10/72	6/74
36552A	GSSS :	SMALL SYSTEMS SIMULATOR	F	10/72	6/74
36553A	GTHOR :	SECURITIES EPS GROWTH	F	10/72	6/74
36554A	GVPDQT:	PLOTTING DATA	F	10/72	6/74
36555B	VCHART:	INVESTMENT DECISIONS USING TEKTRONIX 4010	F	10/72	6/74
36556B	VREGPL:	PLOTTING X AND Y VARIABLES USING TEKTRONIX 4010	F	10/72	6/74
36557A	VRRC :	INVESTMENT STRATEGY ANALYSIS	F	10/72	6/74
36558B	VSUB :	DISPLAY ROUTINE USING TEKTRONIX 4010	F	6/74	6/74
36559B	VTTT :	TIC-TAC-TOE ON THE TEKTRONIX 4010	F	10/72	6/74
36560A	UCHARS:	CREATES FILE 'VCHAR'	F	6/74	6/74
36601B	DRAG :	SIMULATES A DRAG RACE	E	10/73	6/74
36602B	TRUTH :	TRUTH TABLES FOR BOOLEAN EXPRESSIONS	E	10/73	6/74
36603B	ALERA :	PROPAGATION OF ERROR	E	10/73	6/74
36604A	CHARGE:	CHARGE ACCOUNT SIMULATION	F	6/73	6/74
36605A	BASKET:	BASKETBALL STATISTICS	F	10/73	6/74
36606A	FISHER:	FISHER'S EXACT PROBABILITY TEST	F	6/73	6/74
36607A	STAT19:	KRUSKAL-WALLIS ONE WAY ANALYSIS OF VARIANCE	F	6/73	6/74
36608A	STAT20:	FRIEDMAN TWO-WAY ANALYSIS OF VARIANCE	E	10/73	6/74
36609A	MAXFLO:	MAXIMUM FLOW IN A CAPACITATED NETWORK	F	10/73	6/74
36610A	SHORTR:	SHORTEST ROUTE PROOBLEM	F	10/73	6/74
36611A	MEMBR :	DIFFUSION EXPERIMENT	E	6/73	6/74
36612A	FORMIF:	F AND I FORMAT	E	10/73	6/74
36613A	ATWT :	CALCULATES ATOMIC WEIGHT	E	6/73	6/74
36614A	AVOGA :	AVOGADRA'S NUMBER	E	6/73	6/74
36615A	EMPIR :	CALCULATES EMPIRICAL FORMULAS	E	6/73	6/74
36616A	MOLAR :	ACID-BASE TITRATION	E	6/73	6/74
36617A	PHPOH :	PH, POH, PCT, DISSOCIATION	E	6/73	6/74
36618A	STOICH:	MASS VOLUME	E	6/73	6/74
36619A	MARKET:	TWO COMPANIES IN ONE-PRODUCT COMPETITION	E	10/73	6/74
36620A	CVAREA:	AREA UNDER CURVE	E	6/73	6/74
36621A	CHARG :	MILLIKAN'S OIL-DROP EXPERIMENT	E	10/73	6/74
36622A	LIMSIN:	LIMIT OF (SIN X)/X	E	6/73	6/74
36623A	PI :	CALCULATES PI	E	6/73	6/74
36624A	LOCKEY:	LOCK & KEY MODEL OF ENZYME ACTION	E	10/73	6/74
36625A	ROOTS2:	QUADRATIC EQUATION SOLVER	E	6/73	6/74
36626A	SLOPE :	FINDS DERIVATIVES	E	6/73	6/74
36627A	SQRT :	FINDS SQUARE ROOT	E	6/73	6/74
36628A	STATAL:	ARITHMETIC MEAN	E	6/73	6/74
36629A	EFIELD:	ELECTRIC FIELD STRENGTH	E	6/73	6/74
36630A	LENSES:	SOLVES LENS PROBLEMS	E	6/73	6/74
36631A	PRJTL :	PROJECTILE MOTION	E	6/73	6/74
36632A	CIRFLW:	CIRCULAR FLOW MODEL	E	6/73	6/74
36633A	CRVFT :	LEAST-SQUARES CURVEFITTING	E	10/73	6/74
36634A	SYSDAT:	SYSTEM DATE UTILITY	F	10/73	6/74
36635A	METRIC:	CONVERTS ENGLISH TO METRIC	F	6/73	6/74
36636A	GPAC :	COLLEGE GRADE PACKAGE	E	10/73	6/74
36637C	PILOTE:	PILOT 73 AUTHOR LANGUAGE FOR HP 2000E	E	3/75	3/75
36638A	CTC6 :	CTC ACCOUNTS PAYABLE	F	6/73	6/74
36639B	POLUT :	WATER POLLUTION SIMULATION	E	10/73	6/74
36640B	POLSYS:	SIMULATION OF CITY COUNCIL	E	10/73	6/74
36641B	STERL :	FLY POPULATION CONTROL	E	10/73	6/74
36642B	GENE1 :	GENETICS SIMULATION	E	10/73	6/74
36644A	LODUMP:	FILE LOAD/DUMP	F	10/73	6/74
36645B	FILES :	FILE MANIPULATION - CREATES, SORTS, UPDATES, COPIES, CHANGES FORMAT	F	6/74	6/74
36646A	ORGCHE:	DRILL ON ORGANIC COMPOUND NOMENCLATURE	F	10/73	6/74
36647A	SLAB :	SYSTEM LIBRARY ABSTRACTS	E	10/73	6/74
36648A	FMS :	FILE MANAGEMENT SYSTEM	E	10/73	6/74
36649A	POLICY:	STUDENTS FORMULATE NATIONAL POLICY	E	10/73	6/74
36650A	ROULET:	ROULETTE	F	10/73	6/74
36652A	NEWTON:	INTERPOLATION OF NON-LINEAR FUNCTIONS BY NEWTON'S FORMULA	E	6/74	6/74
36653A	BIRDAY:	BIRTHDAY PROBABILITY	F	10/73	6/74
36654A	T-CPL :	THERMOCOUPLE TABLE PACKAGE	E	6/74	6/74
36655A	M/M/1 :	QUEUEING SYSTEM	E	10/73	6/74
36656A	M/M/S :	QUEUEING SYSTEM	E	10/73	6/74
36657A	CLOCKS:	GUESSING TIME GIVEN METRIC CLUE WORDS	E	10/73	6/74
36659A	PLOT33:	KEYBOARD ENTRY MULTIPLE FUNCTION PLOTTER	F	10/73	6/74
36660A	WORD :	A WORD GAME	F	10/73	6/74
36661A	MLREG :	MULTIPLE REGRESSION PROGRAM	E	6/74	6/74
36662A	DERTIV:	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36663A	DERSIN:	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36664A	FUNDTH:	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36665A	GRAPH1:	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36666A	GRAPH2:	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74
36667A	INTEGR:	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36668A	AREA :	COMPUTER-AUGMENTED CALCULUS TOPICS	E	10/73	6/74
36669A	REFRAC:	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74

PROGRAM NUMBER	NAME	TITLE	SYSTEM	SOFTWARE DATE	DOCUMENT DATE
36670A	LANDIN	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74
36671A	MKS1	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74
36672A	REFLEC	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74
36673A	MKS2	COMPUTER-AUGMENTED PHYSICS TOPICS	E	10/73	6/74
36674A	DESCAL	DESK CALCULATOR SUBROUTINE	E	10/73	6/74
36683A	CHASE	PURSUIT OF A TARGET	E	10/73	6/74
36684A	LANDER	ROCKET LANDING VEHICLE	E	10/73	6/74
36685A	CONVRT	TEMPERATURE SCALE CONVERSION	E	10/73	6/74
36686A	GRNPRX	GRAN PRIX RACING GAME	E	10/73	6/74
36688A	EVPI	COMPUTES THE EXPECTED VALUE OF PERFECT INFORMATION	E	10/73	6/74
36689A	CORREL	CORRELATION COEFFICIENT	E	10/73	6/74
36691A	BICONF	CONFIDENCE LIMITS	E	10/73	6/74
36692B	BITEST	BINOMIAL PROPORTION	E	3/75	6/74
36693A	CONLM2	COMPUTES CONFIDENCE LIMITS FOR DIFFERENCE BETWEEN TWO POPULATION MEANS	E	10/73	6/74
36694A	CONLM1	COMPUTES CONFIDENCE LIMITS FOR AN UNKNOWN POPULATION MEAN	E	10/73	6/74
36696A	ROOTNR	LOCATES ROOT OF A FUNCTION WHOSE DERIVATIVE IS KNOWN	E	10/73	6/74
36697A	ROOTNL	FINDS ROOTS OR FIXED POINTS OF A NON-LINEAR FUNCTION	E	10/73	6/74
36698A	INTGR	DEFINITE INTEGRAL BY MEANS OF 3-POINT GAUSSIAN INTEGRATION FORMULA	E	10/73	6/74
36699A	INTGRS	COMPUTES THE AREA UNDER A CURVE	E	10/73	6/74
36700A	LINES	CONVERSATIONAL COMPUTER GENERATED CALCULUS QUIZ	E	10/73	6/74
36701A	MARKOV	COMPUTES FOR AN ERGODIC MARKOV CHAIN	F	10/73	6/74
36702A	PARABO	EQUATION OF PARABOLA PASSING THROUGH 3 GIVEN POINTS	E	10/73	6/74
36703A	POLYGN	COMPUTES THE AREA ENCLOSED IN ANY POLYGON	E	10/73	6/74
36704A	QUADRA	ANALYZES A QUADRATIC EQUATION	E	10/73	6/74
36705A	ATDM	DERIVES THE ELECTRONIC CONFIGURATION OF ANY ELEMENT	E	10/73	6/74
36708A	SAVING	COMPOUND INTEREST	E	10/73	6/74
36709A	MCOST	COMPARE AND EVALUATE UP TO 1000 MORTGAGE PAYMENT PLANS SIMULTANEOUSLY	E	10/73	6/74
36710A	LINQUP	SOLVES EITHER A LINEAR OR QUADRATIC PROGRAMMING PROBLEM	E	10/73	6/74
36711A	FINFLO	CALCULATES PRESENT VALUE - STREAM OF CASH FLOWS	E	10/73	6/74
36712A	OC-DC	OCTAL-TO-DECIMAL CONVERTER	E	10/73	6/74
36713A	BNKSIM	SIMULATES ONE YEAR'S DEPOSIT AND WITHDRAWAL ACTIVITIES OF SMALL BANK	E	10/73	6/74
36716A	EINDIS	EINSTEIN DISTORTIONS	E	6/74	6/74
36718A	PROB	COMPUTES BINOMIAL, POISSON AND HYPERGEOMETRIC PROBABILITIES	E	10/73	6/74
36719A	SEVPRO	CHI-SQUARE TEST	E	10/73	6/74
36720A	FVALUE	EXACT PROBABILITY OF AN F-RATIO WITH DEGREES OF FREEDOM (M,N)	E	10/73	6/74
36721A	TVALUE	COMPUTES THE EXACT PROBABILITY OF A T-VALUE WITH A TWO-TAILED TEST	E	10/73	6/74
36722A	TESTUD	TEST UNKNOWN POPULATION MEAN	E	10/73	6/74
36724A	STAT06	CALCULATES SIGN TEST CONFIDENCE INTERVAL	E	10/73	6/74
36725A	STAT07	CALCULATES THE CONFIDENCE LIMITS FOR A SET OF DATA	E	10/73	6/74
36727A	STAT18	COMPUTES ANALYSIS OF VARIANCE TABLE	E	10/73	6/74
36728A	STAT17	ANALYSIS OF VARIANCE FOR A BALANCED INCOMPLETE BLOCK DESIGN	E	10/73	6/74
36729B	STAT16	COMPUTES AN ANALYSIS OF VARIANCE TABLE AND F-RATIOS	E	3/75	6/74
36730A	STAT14	ANALYSIS OF VARIANCE AND F-RATIOS (RANDOMIZED COMPLETE BLOCK DESIGN)	E	10/73	6/74
36732A	STAT08	COMPARES TWO GROUPS OF DATA USING THE MEDIAN TEST	E	10/73	6/74
36733A	WAVFN	COMPUTES AND PLOTS THE RADIAL PART OF HYDROGEN-LIKE WAVE FUNCTIONS	E	10/73	6/74
36735A	GOMOKU	9 BY 9 BOARD GAME	F	10/73	6/74
36736A	SPSORT	SPEED SORT - GENERAL PURPOSE FILE SORT	F	6/74	6/74
36737A	PI2	INFORMATION SYSTEM	F	2/74	6/74
36738A	REGRES	STEP-WISE REGRESSION	F	10/73	6/74
36743A	TM	TURING MACHINE SIMULATOR	E	10/73	6/74
36746A	GRAZE	GRASSLANDS ECOLOGY SIMULATION	F	10/73	6/74
36747A	DC-OC	DECIMAL-TO-OCTAL CONVERTER	E	10/73	6/74
36748A	MANDSD	CALCULATES BASIC STATISTICS FOR GROUPED AND/OR UNGROUPED DATA	E	10/73	6/74
36749A	EDITOR	FILE MANIPULATION - CREATES, EDITS, LISTS, SORTS, EMULATES G.E. MK II.	F	6/74	6/74
36750A	FORCST	WEATHER FORECASTING PROGRAM	E	6/74	6/74
36754A	TAXMAN	FACTORS AND PRIME NUMBERS ARE THE KEYS TO THIS ONE	E	12/74	6/74
36755C	IDA	INTERACTIVE DATA ANALYSIS	F	3/75	3/75
36756B	PILOTF	PILOT 73 AUTHOR LANGUAGE FOR THE HP 2000F	F	10/73	6/74
36757A	CHARSE	ASCII CHARACTER SET FOR HP 2000E	E	10/73	6/74
36758B	P-TUA	PILOT TUTORIAL	E	10/73	6/74
36759A	P-HYP	SCIENTIFIC METHOD AND HYPOTHESIS	E	10/73	6/74
36760A	CLIMAT	CLIMATE TEACHING PROGRAM	F	6/74	6/74
36764A	TEASER	BRAIN TEASER PUZZLE	E	10/73	6/74
36765A	STPAL	STOP MACHINE AND ASSEMBLY LANGUAGE SIMULATOR	F	10/73	6/74
36768A	QUADGT	FOUR-DIGIT NUMBER GUESSING GAME	F	10/73	6/74
36770A	INDEXR	INDEXING PROGRAM	E	2/74	6/74
36771A	KSWEEP	FREQUENCY PLOT OF POLES AND ZEROES IN A COMPLEX PLANE	F	2/74	6/74
36773A	KING	IF YOU SURVIVE YOUR EIGHT YEAR TERM...WE CONGRATULATE YOU	E	12/74	6/74
36775A	P-TUB	PILOT TUTORIAL	E	10/73	6/74
36776A	TRAP	TRAP MY NUMBER. THEN MAKE THE TRAP SMALLER AND SMALLER AND SMALLER	E	12/74	6/74
36777A	NUMBER	A SIMPLE NUMBER GUESSING GAME	E	12/74	6/74
36779A	P-KEPL	KEPLER'S 3RD LAW	E	2/74	6/74
36782A	BUTTON	BUTTON, BUTTON, WHO'S GOT THE BUTTON?	E	12/74	6/74
36783A	ABAGEL	TRY TO GUESS THE COMPUTER'S 3-LETTER WORD	E	12/74	6/74
36784A	GENFIL	DESIGNS PASSIVE FILTERS	F	2/74	6/74
36785A	DRAGON	SOLAR ECLIPSE SIMULATION	E	2/74	6/74
36786B	HAZEL	HAZELTINE 2000 USER SUBROUTINES	E	6/74	6/74
36787A	CAVES1	FIND YOUR WAY OUT OF THE CAVES. 3 LEVELS OF DIFFICULTY	E	12/74	6/74
36788A	HANGMN	GUESS THE SECRET WORD, LETTER BY LETTER	F	12/74	6/74
36790A	CASE1	CASES IN COMPUTER AND MODEL ASSISTED MARKETING: PLANNING	F	2/74	6/74
36792A	MADLIB	COMPLETE A ZANY STORY BY FILLING IN THE BLANKS..BEFORE YOU SEE THE STORY	F	12/74	6/74
36793A	PSRC	POWER SERIES REGRESSION CURVE WITH X-AXIS OFFSET	F	6/74	6/74
36795A	STARS	GUESS THE COSMIC NUMBER. WIN SOME STARS FOR YOURSELF	E	12/74	6/74
36796A	BASE	BASE CONVERTER	E	6/74	6/74
36797A	MALAR	MALARIA ERADICATION PROGRAM	E	2/74	6/74
36798A	ELECT	CAMPAIGN STRATEGY AND ELECTIONS	F	6/74	6/74
36801A	TIMDIF	FIRST DIFFERENCES, PERCENT CHANGES, PERCENT DIFFERENCE	F	6/74	6/74
36802A	USPOP	HUMAN POPULATION PROJECTION	E	6/74	6/74
36806B	HPASMB	HP ASSEMBLER PACKAGE	F	3/75	6/74
36807A	7AIRE	QUESTIONNAIRE ANALYSIS	F	6/74	6/74
36825A	CAPDCF	CAPITAL INVESTMENT ANALYSIS (DISCOUNTED CASH FLOW METHOD)	F	6/74	6/74
36826A	ORG1	TEACHES STRAIGHT ALKANE NOMENCLATURE AND NAMING OF BRANCHED ALKANES	E	6/74	6/74
36832A	SGFMS	STUDENT GRADE FILE MANAGEMENT SYSTEM	F	6/74	6/74
36833A	LETTER	THERE ARE 26 LETTERS IN THE ALPHABET, BUT ONLY ONE OF THEM IS **IT**	E	12/74	6/74
36834A	23MTCH	23 MATCHES..TAKE TURNS REMOVING THEM..WHOEVER TAKES THE LAST ONE LOSES	E	12/74	6/74
36835A	NIM	THERE ARE PILES OF STARS..WHOEVER TAKES THE LAST STAR WINS	E	12/74	6/74
36837A	CHOMP	THE BOARD IS A BIG COOKIE, BUT ONE CORNER SQUARE IS POISON	E	12/74	6/74
36838B	EDIT2K	TEXT EDITOR FOR THE HP 2000C/2000C'/F	F	3/75	6/74
36840A	PLOTS	ASCII CHARACTER PLOTTER FOR 7200 PLOTTER	F	6/74	6/74
36841B	CASE2	CASES IN COMPUTER AND MODEL ASSISTED MARKETING: DATA ANALYSIS	F	3/75	6/74
36843A	SAP	SURVEY ANALYSIS PROGRAM	E	2/74	6/74
36844A	POP	POPULATION GROWTH	E	2/74	6/74
36845B	COBOL	HP COBOL SIMULATOR	F	2/74	6/74
36847A	BASCAL	BASE CALCULATOR	E	6/74	6/74
36848A	ABASE	GUESS MY NUMBER...IN BASE 2 OR 5 OR 8 OR ...	E	12/74	6/74
36849B	SNOBOL	SNOBOL3 COMPILER FOR HP 2000C SYSTEM	F	3/75	6/74
36851A	ALGIE	BASIC ALGEBRA DRILL	F	6/74	6/74
36854A	ZOT	THE BOARD IS MADE OF STARS, AND WHOMEVER ZOTS THE LAST ONE WINS	E	12/74	6/74
36855A	HURKLE	THE HURKLE IS A HAPPY BEAST - FIND HIM IF YOU CAN!	E	12/74	6/74
36856A	MUGWMP	CIRCLES CROSSING CIRCLES WILL TELL YOU WHERE MUGWUMP HIDES	E	12/74	6/74
36857A	SNARK	CATCH HIM WITH A WELL PLACED CIRCLE	E	12/74	6/74
36858A	AMAZE	GET A RANDOM MAZE TAILORED TO THE SIZE YOU CHOOSE	E	12/74	6/74
36859A	SUNSGN	YOUR NAME AND SUNSIGN ARE USED TO CREATE A UNIQUE ARTISTIC PATTERN	E	12/74	6/74
36860A	CROSS2	CROSS TABULATION AND CHI-SQUARE	F	6/74	6/74
36861A	MANDAL	CREATE YOUR OWN MANDALAS BY CHOOSING DESIGN PARAMETERS	E	12/74	6/74
36862A	DANGLE	USE THE COMPUTER TO MAKE A DANGLING STRING OF STARS..FOR THE VERY YOUNG	E	12/74	6/74
36863A	PMSD	POOLED MEANS AND STANDARD DEVIATIONS	E	6/74	6/74
36864A	FREQ1	FAST FREQUENCY DISTRIBUTIONS	F	6/74	6/74
36867A	FINDAD	CONVERTS A FILE TO A FINDIT FILE	F	6/74	6/74
36869A	CIVIL2	A SIMULATION WHICH RECREATES 14 CIVIL WAR BATTLES	F	12/74	6/74
36870A	ANOVA	FACTORIAL ANALYSIS OF VARIANCE (FIVE-WAY, FOR ANY BALANCED DESIGN)	F	6/74	6/74
36871A	ANVA1	ONE-WAY ANALYSIS OF VARIANCE USING SANPLE MEANS AND STD. DEVIATIONS	E	6/74	6/74
36872A	CAVES2	LIKE CAVES1, BUT **YOU** SET UP THE CAVES FOR A FRIEND TO SOLVE	E	12/74	6/74
36873A	WUMPUS	HUNT THE WUMPUS IN ITS WORLD OF CAVES AND SUPERBATS	E	12/74	6/74
36874A	HAMRBI	AN ECONOMIC SIMULATION TAKING PLACE IN THE ANCIENT CITY-STATE OF SUMER	E	12/74	6/74
36875A	BIOSIN	CHARTS YOUR LIFE'S PHYSICAL, SENSITIVE AND COGNITIVE CYCLES	F	12/74	6/74
36876B	IATATA	ANALYSIS OF IATA AND ATA UNITIZATION PROGRAM	F	3/75	3/75
36877A	P-TUC	PILOT TUTORIAL	F	6/74	6/74
36878B	CHEM1	CAI IN CHEMISTRY	F	3/75	6/74
36879A	BOLA	NUCLEAR PHYSICS GAME	F	6/74	6/74
36880A	VOCAB	CAI IN WORD USAGE	E	6/74	6/74
36881A	ROVER	GUIDE ROVER ROBOT ACROSS A GRID, BUT WATCH OUT FOR THE STOP SIGNS!	E	12/74	6/74
36882A	TRADER	STAR TRADER IS AN INTERSTELLAR GAME OF MERCANTILE SKILL	F	12/74	6/74
36883A	CRASH	LAND A SPACE MODULE ON THE PLANET OF YOUR CHOICE	E	12/74	6/74
36884A	LUNAR	YOU MUST LAND ON THE MOON MANUALLY BECAUSE THE COMPUTERS WENT KAPUT	E	12/74	6/74
36885A	ZEROS	TWO PLAYERS BUILD BRIDGES UNTIL ONE PLAYER CANNOT MOVE	F	12/74	6/74
36886A	REVERS	UNSCRAMBLE A LIST OF NUMBERS WITH YOUR INGENUITY AND LOTS OF PATIENCE	E	12/74	6/74
36888-10044	PHYTOP	INTERACTIVE METHODS FOR SELECTED TOPICS IN PHYSICS & MATH	F	3/75	3/75
36888-18001	INQUIR	SOCIAL SCIENCE INQUIRY PACKAGE	E	3/75	3/75
36888-18002	MCMAN	CARDIOVASCULAR SIMULATION	F	3/75	3/75

PROGRAM NUMBER	NAME	TITLE	SYSTEM	SOFTWARE DATE	DOCUMENT DATE
36888-18003	STATI :	HISTOGRAM, STANDARD DEVIATION & PLOT OF A SET OF NUMBERS	F	3/75	3/75
36888-18004	INDEX :	XREF INDEX OF LIBRARY	F	3/75	3/75
36888-18005	UHCX :	COMPLEX NUMBER CALCULATOR	F	3/75	3/75
36888-18006	TITER :	SIMULATES TITRATION OF A BASE BY AN ACID	E	3/75	3/75
36888-18007	ITEM :	SCORES MULTIPLE CHOICE TESTS	E	3/75	3/75
36888-18008	SRP :	STUDENT RESPONSE PRINT	F	3/75	3/75
36888-18009	POWER2:	POWERS OF TWO TABLES	E	3/75	3/75
36888-18010	ORG3 :	HYDROCARBON CLASSES OF ALKENES AND ALKYNES	E	3/75	3/75
36888-18011	ORG5 :	TEACHES NOMENCLATURE OF ALKYL HALIDES AND ALCOHOLS	E	3/75	3/75
36888-18012	TRIVIA:	COMMON NAME QUIZ FOR CHEMISTRY STUDENTS	F	3/75	3/75
36888-18013	PYRMID:	BUILD A PYRAMID	E	3/75	3/75
36888-18014	POLISH:	POLISH CONVERSION	E	3/75	3/75
36888-18015	LOGICK:	BOOLEAN ALGEBRA EVALUATOR	E	3/75	3/75
36888-18016	P-TUD :	PILOT TUTORIAL	E	3/75	3/75
36888-18017	P-TUE :	PILOT TUTORIAL	E	3/75	3/75
36888-18018	P-MAG :	STELLAR MAGNITUDES	E	3/75	3/75
36888-18019	SIMLAT:	SIMULATION OF INVESTMENT RETURNS WITH SENSITIVITY ANALYSIS	E	3/75	3/75
36888-18020	COFTAB:	CONVERSATIONAL FREQUENCY AND CROSS TABULATOR	F	3/75	3/75
36888-18021	UNLETH:	UNILETH STATISTICS PACKAGE	F	3/75	3/75

PROGRAM NUMBER	NAME	TITLE	SYSTEM	SOFTWARE DATE	DOCUMENT DATE
36888-18022	BASP :	BASIC ANALOG SIMULATION PROGRAM	F	3/75	3/75
36888-18023	STABIL:	FEDERAL GOVERNMENT STABILIZATION PO	E	3/75	3/75
36888-18024	LENGTH:	CWF SUBPROGRAM	F	3/75	3/75
36888-18025	CLEARB:	CWF SUBPROGRAM	F	3/75	3/75
36888-18026	CLEARF:	CWF SUBPROGRAM	F	3/75	3/75
36888-18027	STDATA:	CWF SUBPROGRAM	F	3/75	3/75
36888-18028	MUSIC :	TWELVE TONE COMPOSITION PROGRAM	E	3/75	3/75
36888-18029	BINO :	BINOMIAL FUNCTION EXPANSION	E	3/75	3/75
36888-18030	SIMPLX:	SOLVES LINEAR PROGRAM (CONDENSED TABLEAU METHOD)	F	3/75	3/75
36888-18031	USECON:	TESTING ECONOMIC HYPOTHESES	E	3/75	3/75
36888-18032	ATG :	COMPUTER ORIENTED ACCOUNTING - INTERACTIVE VERSION	E	3/75	3/75
36888-18033	TSAP :	TIME SERIES ANALYSIS GROWTH & DEVELOPMENT OF U.S. 1790-1860	F	3/75	3/75
36888-18034	PLOTXY:	TWO VARIABLE PLOT PROGRAM	E	3/75	3/75
36888-18035	SCOREF:	COMPUTES MEAN, STANDARD DEVIATION AND STANDARD SCORES FOR TEST SCORES	F	3/75	3/75
36888-18036	FORM2K:	TEXT FORMATTER	F	3/75	3/75
36888-18037	FDUMP :	LISTS FILES, TOTAL RECORDS, INDICATES STRINGS & NUMERICS	F	3/75	3/75
36888-18038	TRAN1 :	GRAPHICS UTILITY PROGRAMS	F	3/75	3/75
36888-18039	TSBILL:	TIMESHARE BILLING PROGRAM	F	3/75	3/75
36888-18040	EXTADD:	40 DIGIT PRECISION STRING ADDITION	E	3/75	3/75
36888-18041	TESTGD:	MULTIPLE CHOICE TEST GRADER	E	3/75	3/75
36888-18042	HP7260:	SUBROUTINE TO INPUT CARDS	E	3/75	3/75

HEWLETT hp PACKARD

CUSTOMER SERVICE CENTER
Direct Mail
Parts and Supplies Order Form

SHIP TO:

NAME ________________________________

COMPANY ________________________________ CUSTOMER REFERENCE # ____________

STREET ________________________________ TAXABLE*? ____________

CITY ____________ STATE ____________ ZIP CODE ____________

Please call the nearest HP Sales Office listed on the back of this form if you need assistance in placing your order.

Item No.	Check Digit	Part No.	Qty.	Description	List Price Each	Extended Total

Special Instructions

Sub-total	
Your State & Local Sales Taxes*	
Handling Charge	1 50
TOTAL	

*Tax is verified by computer according to your ZIP CODE. If no sales tax is added, your state exemption number must be provided: # ____________.
If not, your order may have to be returned.

Check or Money Order, made payable to Hewlett-Packard Company, must accompany order.

When completed, please mail this form with payment to:

HEWLETT-PACKARD COMPANY
Mail Order Department
P.O. Drawer #20
Mountain View, CA 94043

Phone: (415) 968-9200

Most orders are shipped within 24 hours of receipt. Shipments to California, Oregon and Washington will be made via UPS. Others shipments will be sent Air Parcel Post, with the exception that shipments over 25 pounds will be made via truck. No Direct Mail Order can be shipped outside the U.S.

HEWLETT-PACKARD
SALES & SERVICE OFFICES

If you need assistance in placing an order, please contact your nearest HP Sales Office. Inquiries regarding status of a Direct Mail Order should be sent to Mountain View, California.

ALABAMA
- **Huntsville** 35802
  Tel: (205) 881-4591

ARIZONA
- **Phoenix** 85034
  Tel: (602) 244-1361
- **Tucson** 85706
  Tel: (602) 889-4661

CALIFORNIA
- **Fullerton** 92631
  Tel: (714) 870-1000
- **North Hollywood** 91604
  Tel: (213) 877-1282
- **Los Angeles** 90045
  Tel: (213) 649-2511
- **Sacramento** 95825
  Tel: (916) 482-1463
- **San Diego** 92123
  Tel: (714) 279-3200
- **Santa Clara** 95050
  Tel: (408) 249-7000

COLORADO
- **Englewood** 80110
  Tel: (303) 771-3455

CONNECTICUT
- **New Haven** 06525
  Tel: (203) 389-6551

FLORIDA
- **Ft. Lauderdale** 33307
  Tel: (305) 731-2020
- **Orlando** 32809
  Tel: (305) 859-2900

GEORGIA
- **Atlanta** 30328
  Tel: (404) 434-4000

HAWAII
- **Honolulu** 96814
  Tel: (808) 955-4455

ILLINOIS
- **Skokie** 60076
  Tel: (312) 677-0400

INDIANA
- **Indianapolis** 46205
  Tel: (317) 546-4891

IOWA
- **Iowa City** 52240
  Tel: (319) 338-9466

LOUISIANA
- **Kenner** 70062
  Tel: (504) 721-6201

MARYLAND
- **Baltimore** 21207
  Tel: (301) 944-5400
- **Rockville** 20850
  Tel: (301) 948-6370

MASSACHUSETTS
- **Lexington** 02173
  Tel: (617) 861-8960

MICHIGAN
- **Farmington** 48024
  Tel: (313) 476-6400

MINNESOTA
- **St. Paul** 55114
  Tel: (612) 645-9461

MISSOURI
- **Kansas City** 64137
  Tel: (816) 763-8000
- **Maryland Heights** 63043
  Tel: (314) 567-1455

NEW JERSEY
- **Paramus** 07652
  Tel: (201) 265-5000

NEW MEXICO
- **Albuquerque** 87110
  Tel: (505) 265-3713
- **Las Cruces** 88001
  Tel: (505) 526-2485

NEW YORK
- **Albany** 12205
  Tel: (518) 458-1550
- **New York City**
  Manhattan, Bronx
  Contact Paramus, NJ Office
  Tel: (201) 265-5000
  Brooklyn, Queens, Richmond
  Contact Woodbury, NY Office
  Tel: (516) 921-0300
- **Poughkeepsie** 12601
  Tel: (914) 454-7330
- **Rochester** 14623
  Tel: (716) 473-9500
- **Syracuse** 13211
  Tel: (315) 455-2486
- **Woodbury** 11797
  Tel: (516) 921-0300

NORTH CAROLINA
- **High Point** 27262
  Tel: (919) 885-8101

OHIO
- **Cleveland** 44130
  Tel: (216) 243-7300
- **Columbus** 43229
  Tel: (614) 436-1041
- **Dayton** 45449
  Tel: (513) 859-8202

OKLAHOMA
- **Oklahoma City** 73132
  Tel: (405) 721-0200

OREGON
- **Tualatin** 97062
  Tel: (503) 620-3350

PENNSYLVANIA
- **King of Prussia** 19406
  Tel: (215) 265-7000
- **Pittsburgh** 15238
  Tel: (412) 782-0400

TENNESSEE
- **Memphis** 38104
  Tel: (901) 274-7472

TEXAS
- **Houston** 77027
  Tel: (713) 781-6000
- **Richardson** 75080
  Tel: (214) 231-6101
- **San Antonio** 78226
  Tel: (512) 434-8241

UTAH
- **Salt Lake City** 84115
  Tel: (801) 487-0715

VIRGINIA
- **Richmond** 23228
  Tel: (804) 285-3431

WASHINGTON
- **Bellevue** 98004
  Tel: (206) 454-3971

WISCONSIN
- **Milwaukee** 53227
  Tel: (414) 541-0550

VOLUME I
CONTENTS

100 DATA HANDLING

VOLUME I
CONTENTS

100 DATA HANDLING Continued

200 TESTING, DEBUGGING AND PROGRAMMING AIDS

CSORT
36888-18043

TITLE: CSORT: PROGRAM TO READ AND SORT CARDS

DESCRIPTION: This program will input data via cards or teletype and then print the information back out in alphabetical order.

INSTRUCTIONS: A file WORK of length 48 must be opened before this program can be run. The program will type a message and then state "NOW PUSH 'READY' ON THE CARD READER". At this point up to 144-72 column pieces of information can be entered via the card reader or by typing them in. The last piece of information must contain END in the first three columns.

Equipment required: TSB/2000E and Teletype (Card reader optional).

SPECIAL CONSIDERATIONS: If one piece of information has END in the first three columns, the program will assume that it is the END card and proceed to the sorting.

ACKNOWLEDGEMENTS: Lary R. Smith
Livonia Public Schools

TITLE: FDUMP: LISTS FILES, TOTAL RECORDS, INDICATES STRINGS AND NUMERICS

DESCRIPTION: This is a simple and useful file listing program. The total number of records used in the file is printed out, and strings are enclosed in < >.

INSTRUCTIONS: Enter name of your file and select the records you wish to list. If your file is very large, it may take a minute to count all of the records.

ACKNOWLEDGEMENTS:

RUN

```
RUN
FDUMP
FILE NAME OR END?TSTAH
NUMBER OF RECORDS IN YOUR FILE =  15ØØ
START RECORD,END RECORD OR Ø,Ø?1,2Ø
STOP AT FIRST EOF?N

RECORD  1
 99Ø  E O R

RECORD  2
<Ø1234-ØØØØØA> 99Ø  <36Ø75-ØØØØØA> 7Ø   <36137-ØØØØØA> 542
<36214-ØØØØØB> 218  <36294-ØØØØØA> 34   <36351-ØØØØØB> 8Ø8
<36611-ØØØØØA> 614  <36664-ØØØØØA> 372  <36733-ØØØØØA> 972
<36832-ØØØØØA> 8ØØ  <-ASCII> 4    <BNDPRC> 1Ø2  <CROUT1> 2Ø2  <EVOLU>
 3Ø2  <GINTLP> 4Ø2  <IN/OUT> 5Ø2  <MARKET> 6Ø2  <PLOTS> 7Ø2  <SHORTR>
 8Ø2  <TALK> 9Ø2  E O R

RECORD  3
< >E O R

RECORD  4
<36257-ØØØØØA><F1Ø4>< ><-ASCII><ASCII CODE GENERATOR><AØ5Z99Z99Z99>< >
< >< >< ><73Ø2> 534   6    < > Ø    < >E O R

RECORD  5
< >E O R

RECORD  6
<36834-ØØØØØA><X9Ø3>< ><23MTCH>
<23 MATCHES..TAKE TURNS REMOVING THEM..WHOEVER TAKES THE LAST ONE LOSES>
<Z99Z99Z99Z99>< >< >< ><7412> 652   8    < > Ø    < >E O R

RECORD  7
< >E O R

RECORD  8
<368Ø7-ØØØØØA><F1Ø2  ><     ><?AIRE   ><QUESTIONNAIRE ANALYSIS>
<4Ø8Z99Z99Z99                        ><                               >< >
<    ><74Ø6        > 132   99Ø  < > Ø    < >E O R

RECORD  9
< >E O R

RECORD  1Ø
<36783-ØØØØØA><X9Ø3>< ><ABAGEL>
<TRY TO GUESS THE COMPUTER'S 3-LETTER WORD><Z99Z99Z99Z99>< >< >< >< >
<7412> 394   12   < > Ø    < >E O R

RECORD  11
< >E O R

RECORD  12
<36848-ØØØØØA><X9Ø3>< ><ABASE>
<GUESS MY NUMBER...IN BASE 2 OR 5 OR 8 OR ...><Z99Z99Z99Z99>< >< >< >< >
<7412> 818   14   < > Ø    < >E O R

RECORD  13
< >E O R

RECORD  14
<36152-ØØØØØA><E82Ø>< ><AC-1><COMPUTER AIDED PRACTICE IN EE AC ANALYSIS>
<513PØ9Z99Z99>< >< >< ><71Ø7> 184   16   < > Ø    < >E O R

RECORD  15
< >E O R

RECORD  16
<36Ø57-ØØØØØA><F513>< ><ACNODE><AC CIRCUIT ANALYSIS PROGRAM>
<82ØZ99Z99Z99>< >< >< ><7Ø12> 49Ø   18   < > Ø    < >E O R
```

```
RECORD  17
< >E O R

RECORD  18
<36293-00000A><F513>< ><ACTFIL><ACTIVE FILTER DESIGN><Z99Z99Z99Z99>< >
< >< >< ><7306> 34     20   < > 0     < >E O R

RECORD  19
< >E O R

RECORD  20
<36231-00000A><E102>< ><ADDRES><ADDRESS LABELS><L05Z99Z99Z99>< >< >< >
< ><7210> 522    22    < > 0     < >E O R
START RECORD,END RECORD OR 0,0?0,0
FILE NAME OR END?END

DONE
```

TITLE: FORM2K: TEXT FORMATTER

DESCRIPTION:

1.1 PURPOSE

THE PURPOSE OF THE FORMATTER IS TO PROVIDE A MEANS OF STRUCTURING THE PRINTING OF THE CONTENTS OF A 2000 TSB SYSTEM DATA FILE FOR A MORE READABLE RESULT. THE FORMATTER ACCEPTS, AS INPUT, ELEMENTS FROM A BASIC 2000 DATA FILE (SUCH AS THOSE PRODUCED BY EDIT2K). THE TEXTUAL CONTENT OF THE FILE IS PRINTED IN A FORMAT CONTROLLED BY FORM2K COMMANDS ENTERED DURING INITIALIZATION AND FOUND IN THE FILE.

ELEMENTS ARE READ IN A SEQUENTIAL MANNER BEGINNING WITH THE FIRST ELEMENT IN THE FILE. THREE TYPES OF ELEMENTS ARE RECOGNIZED: NUMBERS, TEXT STRINGS AND COMMAND STRINGS. NUMBERS ARE IGNORED. TEXT STRINGS ARE DISTINGUISHED FROM COMMAND STRINGS BY A CONTROL CHARACTER (INITIALLY PERIOD [.] BUT CHANGABLE BY COMMAND) APPEARING AS THE FIRST CHARACTER OF THE STRING.

THE OUTPUT FORMAT OF THE TEXT IS CONTROLLED BY COMMANDS PLACED IN THE FILE. THE COMMANDS MAY NOT BE IMBEDDED IN TEXT STRINGS BUT, RATHER, MUST APPEAR AS SEPERATE STRINGS. TWO OR MORE COMMANDS MAY APPEAR IN THE SAME COMMAND STRING IN WHICH CASE THEY MUST BE SEPERATED BY SEMICOLONS[;].

INSTRUCTIONS:

1.2 FILE CAPABILITY

THE FORMATTER HAS A TOTAL FILE CAPABILITY OF FIVE FILES. THE PRIMARY SOURCE OF INPUT IS REFERRED TO AS A TEXT FILE (SEE THE 'TEXT' COMMAND). DURING INITIALIZATION, THE USER MUST DECLARE AT LEAST ONE TEXT FILE AS DESCRIBED IN SECTION 1.2. DURING THE FORMATTING PROCESS, THE USER MAY 'NEST' FILES TO A MAXIMUM LEVEL OF FIVE BY ISSUING ADDITIONAL 'TEXT' COMMANDS WITHIN THE TEXT FILES. THIS WILL CAUSE THE FORMATTER TO SUSPEND ACCEPTING INPUT FROM THE CURRENT TEXT FILE AND COMMENCE TAKING INPUT FROM THE NEW FILE. WHEN THE END-OF-FILE OF THE NEW FILE IS REACHED, THE FORMATTER WILL RESUME ACCEPTING INPUT FROM THE ORIGINAL FILE. FORMATTER PROCESSING NORMALLY TERMINATES WHEN THE END-OF-FILE OF THE ORIGINALLY DECLARED TEXT FILE IS REACHED. ABNORMAL TERMINATION OCCURS WHEN (1) A HARD ERROR IS ENCOUNTERED, OR (2) THE 'BREAK' KEY IS PRESSED.

TEXT FILE NESTING MAY ALSO OCCUR IN A SERIAL FASHION. IN THIS MANNER, THE TEXT FILE FIRST DECLARED WOULD CONSIST OF NOTHING BUT 'TEXT' COMMANDS. SINCE ONLY TWO FILE POSITIONS WOULD BE REQUIRED AT ANY ONE TIME (ASSUMING THE NESTED FILES DID NOT, IN TURN, CAUSING NESTING), AN UNLIMITED NUMBER OF FILES COULD

ACKNOWLEDGEMENTS: Don Coleman
Hewlett-Packard/Data Systems

BE PROCESSED BY THE FORMATTER.

IN ADDITION TO TEXT FILE DECLARATION, THE USER ALSO HAS THE ABILITY TO OPEN FROM ONE TO THREE UTILITY FILES (SEE THE 'OPEN' COMMAND) INCLUDING:

(1) A COMMAND DEFINITION FILE WHICH HOLDS THOSE FORMATTER COMMANDS DEFINED BY THE USER AS DISCUSSED UNDER THE 'DEFINE' COMMAND. THE FILE MAY BE MAINTAINED AND EXPANDED AS A LIBRARY FROM ONE RUN TO THE NEXT.
(2) A NEED FILE TO SAVE LINES AS REQUIRED BY THE 'NEED' COMMAND. THE NEED FILE IS USED TO TEMPORARILY HOLD TABLES, FIGURES, OR ANY OTHER TEXT WHICH THE USER HAS INDICATED CANNOT BE SPLIT BETWEEN TWO PAGES UNTIL AN END-OF-PAGE IS REACHED. FOR A THOROUGH DESCRIPTION OF HOW THE FILE IS USED, SEE THE 'NEED' COMMAND.
(3) A HOLD FILE TO BE USED FOR CONTROLLED AUXILIARY INPUT. THE HOLD FILE MAY CONTAIN BOTH TEXT RECORDS AND FORMATTER COMMANDS. REFER TO THE DEFINITION OF THE 'HOLD' COMMAND.

THE BALANCE OF UNUSED FILES MAY BE USED FOR TEXT FILE NESTING AS PREVIOUSLY DISCUSSED.

1.3 COMMAND DESCRIPTION SYNTAX

THE FORMATTER COMMANDS ARE DESCRIBED IN SECTION 3. WHERE APPROPRIATE, CLASSES OF PARAMETERS ARE SHOWN ON SEPERATE LINES - AS IN THE 'HEADING' COMMAND. TWO OR MORE PARAMETERS OF THE SAME COMMAND MUST BE SEPERATED BY COMMAS. THE PARAMETERS ARE DEFINED IN TERMS OF THE SYNTAX DESCRIPTION ELEMENTS AND SYMBOLS LISTED IN TABLE 1-1.

1.4 FORMATTER FUNCTIONS

ALTHOUGH THE FUNCTIONS OF THE FORMATTER MAY BE DESCRIBED IN MANY DIFFERENT WAYS, THE FOLLOWING CLASSIFICATIONS WERE SELECTED WITH THE HOPES OF CLARIFYING THE USAGE OF THE FORMATTER COMMANDS.

A. PAGE DEFINITION

SIX COMMANDS ARE PROVIDED FOR THE DEFINITION OF THE SIZE OF THE PRINT PAGE, A HEADING IDENTIFIER AND A PAGE INDICATOR. THOSE COMMANDS ARE:

1. BOTTOMSPACE
2. HEADING
3. MARGIN
4. PAGING

```
***********************************************************************
*                                                                     *
*  1.  KEYWORDS - PARAMETERS NOT ENCLOSED IN <>                       *
*                                                                     *
*  2.  []         - ANY PARAMETERS ENCLOSED IN BRACKETS ARE           *
*                   OPTIONAL.                                         *
*                                                                     *
*  3.  ()         - WHEN TWO OR MORE PARAMETERS APPEAR                *
*                   WITHIN PARENTHESES, AT LEAST ONE OF               *
*                   THE PARAMETERS MUST BE GIVEN.                     *
*                                                                     *
*  4.  /          - 'OR'                                              *
*                                                                     *
*  5.  <CHAR>     - ANY CHARACTER EXCEPT "," OR ";"                   *
*                                                                     *
*  6.  <NUM>      - ANY NON-NEGATIVE UNSIGNED INTEGER                 *
*                                                                     *
*  7.  <STRING> - <DELIMITER><CHARSTRING><DELIMITER>                  *
*                                                                     *
*  8.  <CHARSTRING> - <CHAR>[<CHARSTRING>]                            *
*                                                                     *
*  9.  <DELIMITER> - <CHAR> NOT APPEARING IN <CHARSTRING>             *
*                                                                     *
***********************************************************************
```

TABLE 1-1
COMMAND DEFINITION NOTATION

5. PAPER
6. TOPSPACE

B. TEXT INPUT CONTROL

TEXT INPUT TO THE FORMATTER CAN BE PROVIDED FROM THREE DIFFERENT SOURCES: A TEXT FILE, AN AUXILIARY HOLD FILE OR THE KEYBOARD. THE COMMANDS AVAILABLE ARE:

1. TEXT
2. HOLD
3. ENTER

C. PAGE FORMAT CONTROL

THE FOLLOWING COMMANDS ALLOW THE USER TO CONTROL THE CONTENT ON EACH PAGE AND THE SPACING BETWEEN THE LINES.

1. NEED
2. NEWPAGE
3. ODDPAGE
4. SKIP
5. SPACING

D. LINE FORMAT CONTROL

LINE FORMAT CONTROL COMMANDS ALLOW THE USER CONTROL OVER HOW THE OUTPUT TEXT WILL APPEAR.

1. ADJUST
2. BLANK
3. BREAK
4. CENTER
5. FILL
6. INDENT
7. SUPPRESS
8. TAB
9. UNDENT

E. MISCELLANEOUS

THE FOLLOWING COMMANDS PROVIDE ADDITIONAL FACILITIES TO THE USER.

1. CHECK
2. CONTROL
3. DEFINE
4. FLAG
5. OPEN
6. PAUSE
7. REPEAT

2.0 OPERATING PROCEDURES

2.1 FORMATTER INITIALIZATION

A USER LOGGED ONTO A 2000C HIGH SPEED OR 2000F TIME SHARE SYSTEM MAY OBTAIN A COPY OF THE FORMATTER BY ENTERING THE COMMAND:

```
GET-FORM2K
```

PRIOR TO EXECUTING THE PROGRAM, ANY DESIRED UTILITY FILES (COMMAND FILE OR NEED FILE) SHOULD BE CREATED VIA THE TSB SYSTEM 'OPEN' COMMAND IF THEY DO NOT ALREADY EXIST. EXECUTION IS THEN INITIATED BY ENTERING THE COMMAND:

```
RUN
```

THE PROGRAM RESPONDS WITH THE CONTROL CHARACTER AS A PROMPT. IF THE USER WISHES TO MAKE USE OF ANY UTILITY FILES, THEY SHOULD BE OPENED WITH THE FORMATTER 'OPEN' COMMAND AT THIS TIME. FOR A DESCRIPTION OF HOW TO SPECIFY A UTILITY FILE, SEE THE 'OPEN' COMMAND.

THE USER MAY THEN ENTER ANY DESIRED INITIALIZATION COMMANDS NECESSARY TO TAILOR THE FORMATTER TO THE IMMEDIATE TASK. REFER TO APPENDIX A FOR THE FORMATTER PARAMETER DEFAULT VALUES AND WHICH COMMANDS CAN BE EXECUTED DURING INITIALIZATION. NOTE THAT AT LEAST ONE 'TEXT' COMMAND MUST BE PROVIDED. EACH TIME A STRING OF FORMATTER COMMANDS IS EXECUTED, THE PROGRAM RETURNS TO THE USER WITH THE PROMPT CHARACTER TO REQUEST MORE INPUT.

IF THE INITIAL VALUES (SEE APPENDIX A) ARE SATISFACTORY, THE USER NEED ONLY ENTER THE TEXT FILE NAME IN THE FORM:

```
TEXT <FNAME>
```

A CARRIAGE RETURN IN RESPONSE TO THE CONTROL CHARACTER PROMPT WILL TERMINATE INITIALIZATION AND START THE FORMATTING PROCESS.

2.2 PERFORMANCE INPROVEMENT SUGGESTIONS

INPUT COMMANDS AND PARAMETERS MAY BE EITHER UPPER OR LOWER CASE FOR THE CONVIENCE OF THE USER. THE INITIAL COMMAND SCAN, HOWEVER, WILL BE FOR UPPER CASE CHARACTERS. IF A COMMAND IS NOT IDENTIFIED, THE PROCESS IS REPEATED FOR THE LOWER CASE. PERFORMANCE CAN THUS BE IMPROVED BY USING UPPER CASE LETTERS EXCLUSIVELY.

EACH LINE OF CHARACTERS TO BE PRINTED IS FIRST SCANNED FOR ANY 'BLANK' CHARACTERS UNLESS THE 'BLANK' CHARACTER IS A SPACE. IMPROVED PERFORMANCE CAN BE OBTAINED BY SETTING THE 'BLANK' CHARACTER TO A SPACE UNLESS IT IS BEING USED.

2.3 RESTRICTIONS

ALTHOUGH THE FORMATTER ALLOWS INPUT COMMANDS AND PARAMETERS IN EITHER UPPER OR LOWER CASE, BOTH MODES CANNOT BE MIXED WITHIN A COMMAND.

3.0 COMMAND DEFINITION

THE COMMAND SYNTAX IS DEFINED ON THE FOLLOWING PAGES. ALTHOUGH THE COMMANDS ARE SPELLED OUT, ONLY THE FIRST THREE LETTERS OF THE COMMAND ARE RECOGNIZED BY THE FORMATTER.

THE PARAMETERS MUST BE SEPERATED FROM THE COMMAND BY AT LEAST ONE SPACE. IF TWO OR MORE PARAMETERS ARE GIVEN, THEY MUST BE SEPERATED BY COMMAS.

3.1 ADJUST (ON/OFF)

'ADJUST' PROVIDES FOR THE JUSTIFICATION OF TEXT TO THE RIGHT MARGIN BY ADDING MORE BLANKS WHERE IMBEDDED BLANKS APPEAR.

WHEN THE 'ADJUST' FEATURE IS TURNED 'ON', THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. FOLLOWING LINES OF TEXT CONTAINING AT LEAST ONE BLANK BETWEEN THE LEFT MARGIN AND THE LAST NON-BLANK CHARACTER WILL BE EXTENDED TO THE RIGHT MARGIN BY ADDING BLANKS WHERE IMBEDDED BLANKS APPEAR. PROCESSING OCCURS FROM RIGHT TO LEFT WITH EACH BLANK BEING REPLACED BY TWO UNTIL THE LEFT MARGIN IS REACHED. IF THE LEFT MARGIN IS REACHED BEFORE THE LINE HAS BEEN FULLY EXPANDED, THE PROCESS IS REPEATED. IF A LINE CANNOT BE ADJUSTED DUE TO THE ABSENCE OF BLANKS IN THE OUTPUT LINE, AN ERROR MESSAGE WILL BE PRINTED.

IF THE USER DOES NOT WISH A SPACE TO BE EXPANDED DURING 'ADJUST', A SPECIAL CHARACTER MAY BE USED INSTEAD OF A BLANK. THE CHARACTER IS PRINTED AS A BLANK. SEE THE 'BLANK' COMMAND DESCRIPTION.

3.2 BLANK <CHAR>

THE 'BLANK' CHARACTER IS USED TO PREVENT THE ADDITION OF BLANKS AT USER SELECTED POSITIONS IN TEXT WHICH IS BEING ADJUSTED. THE 'BLANK' COMMAND MAY BE USED TO CHANGE THE CHARACTER FROM IT'S ORIGINAL VALUE OF % TO ANY CHARACTER EXCEPT COMMA [,] OR SEMICOLON [;].

```
EXAMPLE:
    .BLANK #
    THIS#IS#AN#EXAMPLE
WILL APPEAR AS:
    THIS IS AN EXAMPLE
```

3.3 BOTTOMSPACE ([A=<NUM>]/ [B=<NUM>])

ON EACH PRINTED PAGE OF TEXT, A BOTTOM-OF-PAGE FORMAT CAN BE DEFINED. THE 'BOTTOMSPACE' COMMAND ALLOWS THE USER TO DEFINE THE NUMBER OF LINES SURROUNDING THE PAGING STRING. THE FORMAT CONSISTS OF:

(1) 'A' NUMBER OF BLANK LINES BETWEEN THE LAST PRINTED TEXT LINE OF THE PAGE AND THE PAGE STRING.
(2) A PAGE STRING (SEE 'PAGING').
(3) 'B' NUMBER OF BLANK LINES BETWEEN THE PAGE STRING AND THE END OF THE PAGE.

THE 'A' PARAMETER DETERMINES THE NUMBER OF BLANK LINES WHICH WILL PRECEDE THE PAGE STRING. THE 'B' PARAMETER DETERMINES THE NUMBER OF BLANK LINES WHICH WILL BE PRINTED BETWEEN THE PAGE STRING AND THE END OF THE PAGE. THE INITIAL VALUES ARE A=3, B=3.

THE 'A' AND 'B' PARAMETERS ARE RELATED TO THE LENGTH OF THE PAGE (SEE 'PAPER' COMMAND) AND THE TOP-OF-PAGE FORMAT (SEE THE 'TOPSPACE' COMMAND). THE PAGE LENGTH MINUS THE 'A' AND 'B' PARAMETERS OF BOTH THE TOP AND BOTTOM-OF-PAGE FORMAT CANNOT BE LESS THAN 2. IF SO DEFINED, A LIMIT ERROR WILL RESULT.

3.4 BREAK

THE 'BREAK' COMMAND WILL CAUSE THE CURRENT LINE OF FORMATTED TEXT TO BE PRINTED AND SUBSEQUENT TEXT TO BE STARTED ON A NEW LINE.

3.5 CENTER [<NUM>]

THE 'CENTER' COMMAND WILL CAUSE THE NEXT <NUM> INPUT TEXT LINES (COMMAND STRINGS ARE NOT COUNTED) TO BE CENTERED BETWEEN THE LEFT AND RIGHT MARGINS (SEE 'MARGIN'). IF <NUM> IS 0 OR NOT GIVEN, 1 IS THE DEFAULT VALUE USED. ALL OTHER FORMATTER COMMANDS ARE IGNORED WHEN 'CENTER' IS BEING EXECUTED.

EXAMPLE:

```
.CENTER 3
```

WILL CAUSE THE NEXT THREE INPUT TEXT LINES TO BE CENTERED BETWEEN THE MARGINS.

3.6 CHECK [LIST]

A HARD ERROR WILL TERMINATE FORMATTER PROCESSING IMMEDIATELY. THE 'CHECK' COMMAND PROVIDES THE MEANS OF DETERMINING IF ANY HARD ERRORS WILL BE PRODUCED BY FORMATTER COMMANDS IN A FILE OR SET OF FILES.

WHEN THE 'CHECK' COMMAND HAS BEEN ISSUED, ONLY FORMATTER COMMANDS IN THE TEXT FILE(S) ARE SCANNED FOR VALIDITY. WHEN A FILE IS ENTERED, THE FILE NAME IS PRINTED TO INDICATE FROM WHICH FILE THE COMMANDS ARE BEING READ. EACH COMMAND IS EXECUTED AS IN NORMAL PROCESSING WITH THE EXCEPTION OF THE 'REPEAT' COMMAND. IF A HARD ERROR IS DETECTED, THE OFFENSIVE COMMAND AND AN APPROPRIATE MESSAGE WILL BE PRINTED TO IDENTIFY THE PROBLEM. PROCESSING WILL THEN CONTINUE UNTIL THE END-OF-FILE OF THE TEXT FILE FIRST DECLARED IS REACHED.

THE [LIST] OPTION PROVIDES FOR THE PRINTOUT OF EACH COMMAND STRING. FOR EACH COMMAND STRING READ FROM THE FILE, THE OUTPUT, CONSISTING OF THE STRING PRECEDED BY ITS' RELATIVE STRING POSITION WITHIN THE FILE (NUMBERS ARE IGNORED), IS INDENTED BY TWO SPACES FOR EACH NESTED LEVEL OF THE FILE AND PRINTED ON THE LIST DEVICE.

3.7 CONTROL <CHAR>

A COMMAND STRING IS IDENTIFIED BY A 'CONTROL' CHARACTER APPEARING AS THE FIRST CHARACTER OF THE STRING. THE 'CONTROL' COMMAND MAY BE USED TO CHANGE THE CHARACTER FROM ITS ORIGINAL VALUE OF PERIOD [.] TO ANY OTHER CHARACTER.

EXAMPLE:

```
    .CONTROL ?
    ?SKIP 3
```

3.8 DEFINE <CNAME>=<STRING>

IT MAY BE THE CASE THAT CERTAIN COMBINATIONS OF FORMATTER COMMANDS APPEAR FREQUENTLY IN A FILE TO BE FORMATTED. IT WOULD BE SIMPLER, AND PERHAPS MORE MEANINGFUL, TO REPLACE THE SET OF COMMANDS WITH A SINGLE COMMAND. THE 'DEFINE' COMMAND PROVIDES THIS CAPABILITY BY ALLOWING THE USER TO DEFINE HIS OWN COMMANDS.

IF A USER EXPECTS TO DEFINE HIS OWN COMMANDS, A 'COMMAND DEFINITION FILE' MUST EXIST AND HAVE BEEN OPENED (SEE THE 'OPEN' COMMAND) BEFORE DECLARING THE FIRST TEXT FILE. THE FILE MAY HAVE JUST BEEN CREATED VIA A 2000 TSB COMMAND OR MAY HAVE BEEN USED DURING PREVIOUS FORMATTING SESSIONS. EACH TIME A 'DEFINE' COMMAND IS ENCOUNTERED, THE COMMAND DEFINITION FILE IS SCANNED TO PREVENT DUPLICATE COMMAND ENTRIES. IF FOUND TO BE UNIQUE, IT IS INSERTED IN THE FILE FOR FUTURE USE.

USER DEFINED COMMANDS ARE SUBJECT TO THE FOLLOWING RESTRICTIONS:

1. THE FIRST THREE LETTERS OF THE COMMAND NAME MUST NOT CORRESPOND TO THE FIRST THREE LETTERS OF ANY STANDARD FORMATTER COMMAND NAMES.
2. THE COMMAND STRING IS LIMITED BY THE END OF THE STANDARD 72 CHARACTER LINE.
3. THE COMMAND STRING MAY INCLUDE ONLY ONE USER DEFINED COMMAND. IF INCLUDED, IT MUST BE THE LAST COMMAND OF THE STRING.

EXAMPLE: CONSIDER THE CASE IN WHICH A FILE IS COMPOSED OF A SET OF SEGMENTS, EACH TO BE PRINTED ON A SEPERATE PAGE. EACH SEGMENT IS PRECEDED BY A MULTILINE HEADING WHICH IS NOT TO BE EITHER FILLED OR ADJUSTED. THE HEADING IS TO BE SEPERATED FROM THE BODY OF THE SEGMENT BY A DOUBLESPACE, THEN THE BODY OF THE SEGMENT IF TO BE BOTH FILLED AND ADJUSTED.

THE USER MAY DEFINE THE COMMANDS:

```
.DEFINE SEGMENT="NEW;FILL OFF;ADJUST OFF"
.DEFINE BODY="SKIP 2;FILL ON;ADJUST ON"
```

THE DESIRED OUTPUT MAY BE ACHIEVED BY PRECEDING EACH SEGMENT WITH THE COMMAND:

```
SEGMENT
```

AND INSERTING THE COMMAND:

```
BODY
```

AFTER EACH SEGMENT HEADING.

3.9 ENTER [<NUM>]

IT MAY BE THE CASE THAT INFORMATION MUST BE ADDED TO THE OUTPUT THAT IS NOT AVAILABLE WHEN THE FILE IS BUILT, SUCH AS A NAME OR A DATE; OR IT MAY BE THAT OUTPUT CONTROL MUST VARY FROM ONE PRINTING TO THE NEXT, SUCH AS SWITCHING BETWEEN SINGLE AND DOUBLE SPACING.

THE 'ENTER' COMMAND ALLOWS THE USER TO INSERT EITHER TEXT OR COMMANDS FROM THE TERMINAL. THE OPTIONAL <NUM> PARAMETER IS USED TO INDICATE THE NUMBER OF LINES EXPECTED TO BE ENTERED. IF A NUMBER IS NOT PROVIDED, THE DEFAULT VALUE OF 1 IS USED.

EXAMPLE:

```
.ENTER 3
```

WILL ALLOW THE USER TO INSERT THREE STRINGS FROM THE KEYBOARD.

3.10 FILL (ON/OFF)

THE 'FILL' COMMAND IS ONE OF THE PRIMARY COMMANDS OF THE FORMATTER. IT IS USED TO PLACE THE MAXIMUM NUMBER OF COMPLETE WORDS ON EACH PRINTED LINE.

WHEN FILL IS TURNED ON, THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. FOLLOWING LINES OF TEXT ARE THEN FORMATTED BY PLACING THE MAXIMUM NUMBER OF WORDS, WITH RESPECT TO THE LEFT AND RIGHT MARGIN, ON EACH PRINTED LINE. IF A LINE CANNOT BE FILLED BECAUSE OF THE ABSENCE OF BLANKS IN THE INPUT LINE, THE OUTPUT LINE WILL BE TERMINATED WITH A HYPHEN AND AN ERROR MESSAGE WILL BE PRINTED. THE FILL OPERATION WILL REMAIN IN EFFECT UNTIL IT IS TURNED OFF.

3.11 FLAG (ON/OFF/
<CHAR>/
<NUM>)

THE 'FLAG' COMMAND PROVIDES A MEANS OF PLACING AN IDENTIFICATION MARKER ON SELECTED TEXT LINES.

THE 'FLAG' COMMAND PROVIDES FOR THE MARKING OF PRINTED LINES BY PLACING A FLAG CHARACTER THREE SPACES TO THE RIGHT OF THE RIGHT MARGIN. THE DEFAULT FLAG CHARACTER IS THE EXCLAMATION MARK[!]. THIS CHARACTER MAY BE CHANGED VIA THE 'FLAG' COMMAND TO ANY OTHER CHARACTER EXCEPT THE COMMA[,] OR SEMICOLON[;].

LINES MAY BE FLAGGED BY ONE OF TWO METHODS. THE FLAG COMMAND MAY BE TURNED 'ON' IN WHICH CASE OUTPUT LINES WILL BE FLAGGED UNTIL A 'FLAG OFF' COMMAND IS ENTERED. ALTERNATIVELY, THE COMMAND 'FLAG <NUM>' MAY BE ENTERED IN WHICH CASE THE NEXT <NUM> INPUT LINES WILL BE FLAGGED.

NOTE: THE 'ON/OFF' OR '<NUM>' PARAMETERS PERTAIN TO TEXT LINES ENTERED INTO THE FORMATTER. MORE OR FEWER OUTPUT LINES THAN INDICATED MAY BE FLAGGED AS A RESULT OF THE 'FILL' COMMAND.

EXAMPLE:

THE COMMAND:

```
.FLAG 3,#
```

WILL RE-DEFINE THE FLAG CHARACTER TO BE A # AND WILL CAUSE IT TO BE PLACED IN THE RIGHT MARGIN OF THE FORMATTED RESULT OF THE NEXT 3 LINES OF TEXT READ FROM THE FILE. IF, AS A RESULT OF FILL, THE NEXT 3 INPUT LINES RESULT IN 2 OUTPUT LINES, ONLY 2 FLAG CHARACTERS WILL BE PRINTED.

3.12 HEADING (ON/OFF/
CENTER/MARGIN/FACING/OPPOSED/
<STRING>)

EACH PAGE OF FORMATTED TEXT MAY BE IDENTIFIED BY A HEADING STRING. THE 'HEADING' COMMAND PROVIDES FOR THE FORMAT AND CONTENT OF THE PAGE HEADING.

IF THE HEADING STRING DOES NOT CONTAIN ANY PRINTABLE CHARACTERS, OR IF THE HEADING OPTION IS 'OFF', NO HEADING STRING WILL APPEAR. A BLANK LINE, HOWEVER, WILL BE PRINTED.

THE HEADING PRINT FEATURE MAY BE TURNED 'ON' OF 'OFF' WITHIN THE TEXT FILE. ONCE TURNED 'ON', THE HEADING STRING WILL BE PRINTED AT THE TOP OF EACH PAGE UNTIL TURNED 'OFF' OR THE INPUT TEXT FILE IS EXHAUSTED.

THE PRINT LOCATION OF THE HEADING STRING WITH REGARD TO THE MARGINS IS CONTROLLED BY ONE OF FOUR PARAMETERS:

- CENTER - THE HEADING STRING WILL BE CENTERED BETWEEN THE MARGINS.
- MARGIN - THE HEADING STRING WILL BE RIGHT JUSTIFIED.
- FACING - THE HEADING STRING WILL BE RIGHT JUSTIFIED ON ODD NUMBERED PAGES AND LEFT JUSTIFIED ON EVEN NUMBERED PAGES.
- OPPOSED - THE HEADING STRING WILL BE JUSTIFIED TO THE MARGIN OPPOSITE THE PAGE STRING (SEE 'PAGING').

THE HEADING STRING MAY BE CHANGED AT ANY TIME DURING PROCESSING BY THE EXECUTION OF THE COMMAND 'HEADING <STRING>' WHERE <STRING> IS TO BE THE NEW HEADING STRING.

EXAMPLE: THE HEADING ON THIS PAGE IS A RESULT OF THE COMMAND:

```
.HEADING ON,MARGIN,"FORM2K FORMATTER"
```

3.13 HOLD [<NUM>]

THE 'HOLD' FILE ALLOWS THE USER TO INSERT EITHER TEXT OR COMMANDS FROM AN AUXILIARY INPUT FILE. THE USER SPECIFIES THE NUMBER <NUM> OF INPUT LINES TO BE TAKEN FROM THE HOLD FILE. IF A NUMBER IS NOT GIVEN, THE DEFAULT VALUE OF 1 IS USED.

NOTE: IF A HOLD FILE IS TO BE USED, IT MUST BE OPENED (SEE 'OPEN' COMMAND) BEFORE THE FIRST TEXT FILE IS DECLARED.

EXAMPLE:

```
.HOLD 3
```

WILL CAUSE THE FORMATTER TO READ THE NEXT 3 STRINGS FROM THE HOLD FILE.

3.14 INDENT <NUM> [FOR <NUM>]

THE 'INDENT' COMMAND ALLOWS THE USER TO INDENT TEXT LINES WITHOUT CHANGING THE LEFT MARGIN.

THE 'BREAK' COMMAND IS EXECTED TO PRINT ANY UNPRINTED FORMATTED TEXT. FOLLOWING TEXT WILL THEN BE INDENTED <NUM> SPACES FROM THE LEFT MARGIN. THE 'FOR' PARAMETER DETERMINES THE NUMBER OF INPUT LINES TO BE FORMATTED WITH THE INDENTATION. IF NOT GIVEN, THE 'FOR' DEFAULT VALUE IS 1.

NOTE: IF 'FILL' IS ON, MORE OR LESS THAN 'FOR <NUM>' OUTPUT LINES MAY BE INDENTED.

3.15 MARGIN (LEFT=<NUM>/ RIGHT=<NUM>)

'MARGIN' PROVIDES FOR THE SETTING OF LEFT AND RIGHT MARGINS WITHIN THE PAGE WIDTH TO LIMIT THE FORMATTED OUTPUT.

THE LEFT MARGIN DEFINES THE NUMBER OF SPACES TO PRECEDE THE PRINTED LINE (MODIFIABLE BY 'INDENT' AND 'UNDENT'). THE RIGHT MARGIN INDICATES THE NUMBER OF SPACES TO BE LEFT BETWEEN THE LAST PRINTABLE CHARACTER OF TEXT AND THE RIGHT SIDE OF THE PAGE. IF LINES ARE TO BE FLAGGED, THE RIGHT MARGIN SHOULD HAVE A VALUE OF AT LEAST THREE.

EXAMPLE:

```
.MAR LEFT=5,RIGHT=5
```

WILL LEAVE A 5 CHARACTER MARGIN ON EACH SIDE OF THE PAGE.

NOTE THAT THE LEFT AND RIGHT MARGINS ARE RELATED TO THE LINE WIDTH (SEE THE 'PAPER' COMMAND). THE SUM OF THE TWO MARGINS MUST BE GREATER THAT THE LINE WIDTH.

2.16 NEED <NUM>
[,KEEP <NUM>]

THE 'NEED' COMMAND IS USED TO GUARANTEE THAT A SECTION OF TEXT WILL NOT BE SPLIT BY A PAGE BOUNDARY. CONSIDER, FOR EXAMPLE, A TEXT FILE INCLUDING 20 LINES OF TEXT WHICH CONSTITUTES A TABLE OR ILLUSTRATION. THE USER MAY NOT WISH PART OF THE TABLE PRINTED ON ONE PAGE AND THE REMAINDER PRINTED ON THE FOLLOWING PAGE. SUCH A CONDITION CAN BE AVOIDED BY IMMEDIATELY PRECEDING THE 20 LINES OF TEXT WITH THE COMMAND ".NEED 20". IF, WHEN THE 20 LINES ARE TO BE PRINTED, 20 LINES DO NOT REMAIN ON THE PAGE, THE PRINTING WILL BE DEFERRED UNTIL THE TOP OF THE NEXT PAGE IS REACHED.

WHEN DECLARING THE NUMBER OF LINES NEEDED, ALLOWANCE SHOULD BE MADE FOR LINES INSERTED BY THE FORMATTER SUCH AS IN THE CASE OF A 'SKIP' COMMAND OR IF DOUBLE SPACING IS IN EFFECT. IF THE TEXT IS BEING 'FILLED', THE 'NEED' REQUIREMENTS WILL BE MORE DIFFICULT TO DETERMINE.

THE 'NEED' COMMAND MAY BE USED IN CONJUNCTION WITH THE 'NEED FILE'. THE RESULT OF COMMAND EXECUTION WILL BE DETERMINED BY THE EXISTENCE OR ABSENCE OF THE NEED FILE.

'NEED' FILE NOT GIVEN:

A CHECK IS MADE TO DETERMINE IF <NUM> LINES REMAIN ON THE CURRENT PAGE. IF SO, THE FORMATTER CONTINUES. IF NOT, A PAGE EJECT IS ISSUED BEFORE CONTINUING. THE 'KEEP' PARAMETER, IF GIVEN, IS IGNORED.

'NEED' FILE GIVEN:

THE 'NEED' FILE IS USED TO REDUCE, IF NOT ELIMINATE, THE NUMBER OF BROKEN PAGES IN THE TEXT OUTPUT AS A RESULT OF THE 'NEED' COMMAND. IF A 'NEED' COMMAND IS ENTERED BUT THE NECESSARY LINES ARE NOT AVAILABLE ON THE CURRENT PAGE, THE 'NEED' TEXT IS SAVED IN THE 'NEED' FILE AND THE FORMATTER CONTINUES PROCESSING AFTER THE NEXT <NUM> (OR 'KEEP' <NUM> IF GIVEN) LINES. WHEN THE NEXT PAGE EJECT OCCURS, THE CONTENT OF THE 'NEED' FILE WILL BE PRINTED AT THE BEGINNING OF THE NEW PAGE. THUS INFORMATION WHICH FOLLOWS A TABLE IN THE TEXT FILE MAY PRECEDE THE TABLE IN THE OUTPUT.

IF THE 'NEED' FILE ALREADY CONTAINS TEXT AS A RESULT OF A PREVIOUS 'NEED' COMMAND, A PAGE EJECT IS GIVEN AND THE 'NEED' FILE IS EMPTIED. THE CURRENT 'NEED' COMMAND IS THEN RE-EXECUTED TO DETERMINE WHETHER TO PRINT THE TEXT OR PUT IT IN THE NEED FILE.

THE 'KEEP' PARAMETER DETERMINES THE NUMBER OF LINES TO TRANSFER TO THE 'NEED' FILE. IT MAY BE GREATER OR LESS

THAN <NUM> IF FORMATTER COMMANDS ARE TO BE SAVED WITH THE TEXT. IF NOT GIVEN, THE 'KEEP' DEFAULT IS <NUM>.

NOTE: WHEN TEXT IS TRANSFERRED TO THE 'NEED' FILE, THE CURRENT STATUS OF THE FORMATTER PARAMETERS IS ALSO SAVED SO THAT THE 'NEED' TEXT WILL BE PRINTED UNDER THE CONDITIONS IN FORCE AT THE TIME OF ITS ENTRY INTO THE FILE. WHEN THE 'NEED' FILE IS PRINTED, FORMATTER STATUS AT THAT TIME IS SAVED, THEN RESTORED AFTER THE 'NEED' FILE IS EMPTIED. THUS ANY PARAMETER ALTERATIONS WHICH OCCURRED DURING THE 'NEED' FILE PRINT WILL NOT BE RETAINED.

3.17 NEWPAGE [<NUM>]

THE 'NEWPAGE' COMMAND ALLOWS THE USER TO FORCE TEXT TO THE BEGINNING OF A NEW PAGE.

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. A PAGE EJECT WILL THEN BE ISSUED.

THE USER MAY RESET THE PAGE NUMBER TO ANY NON-NEGATIVE VALUE BY PROVIDING <NUM> IN THE COMMAND. IF THE PAGE NUMBER IS SET TO 0, ONLY THE PAGE STRING WILL BE PRINTED. THE PAGE NUMBER WILL REMAIN AT 0 UNTIL RESET TO A POSITIVE NUMBER BY ANOTHER NEWPAGE COMMAND.

3.18 ODDPAGE

THE 'ODDPAGE' COMMAND ALLOWS THE USER TO FORCE TEXT TO THE BEGINNING OF THE NEXT ODD NUMBERED PAGE. ONE EXAMPLE OF A USE WOULD BE TO BEGIN A SEPERATE SECTION OF TEXT ON AN ODD NUMBERED PAGE AS IN CHAPTERS OF A BOOK.

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. A PAGE EJECT IS THEN ISSUED. IF THE NEW PAGE NUMBER IS EVEN, A SECOND PAGE EJECT IS ISSUED.

3.19 OPEN (HOLD FILE = <FILE NAME>/
NEED FILE = <FILE NAME>/
COMMAND FILE = <FILE NAME>)

THE 'OPEN' COMMAND ALLOWS THE USER TO DEFINE ONE OR MORE UTILITY FILES TO BE USED DURING PROCESSING. THESE FILES MUST BE OPENED DURING INITIALIZATION AND PRIOR TO THE DEFINITION OF THE FIRST 'TEXT' FILE. REFER TO SECTION 1.2 FOR A DESCRIPTION OF HOW THE FILES ARE USED.

3.20 PAGING (ON/OFF/
CENTER/MARGIN/FACING/
TOP/BOTTOM/
<STRING>)

EACH PAGE OF FORMATTED TEXT MAY BE IDENTIFIED BY A PAGE INDICATOR COMPRISED OF A PAGE STRING AND A PAGE NUMBER. THE 'PAGING' COMMAND PROVIDES FOR THE MODIFICATION AND CONTROL OF THE FORMAT AND CONTENT OF THIS INDICATOR.

THE PAGE INDICATOR PRINT FEATURE MAY BE TURNED 'ON' OR 'OFF' WITHIN THE TEXT FILE. ONCE TURNED ON, THE PAGE INDICATOR WILL BE CONSTRUCTED BY PREFIXING THE PAGE STRING TO THE PAGE NUMBER AND PRINTED WHERE DIRECTED (BY OTHER 'PAGING PARAMETERS) ON EACH PAGE UNTIL TURNED OFF. THE PAGE NUMBER IS AUTOMATICALLY INCREMENTED EACH TIME A PAGE EJECT OCCURS.

THE PRINT LOCATION OF THE PAGE INDICATOR WITH REGARD TO THE PAGE IS CONTROLLED BY ONE OF TWO PARAMETERS:

TOP - THE PAGE INDICATOR WILL BE PRINTED ON THE SAME LINE AS THE HEADING INDICATOR.
BOTTOM - THE,PAGE INDICATOR WILL BE PRINTED AT THE BOTTOM OF EACH PAGE.

THE PRINT LOCATION OF THE PAGE INDICATOR WITH REGARD TO THE MARGINS IS CONTROLLED BY ONE OF THREE PARAMETERS:

- CENTER - THE PAGE INDICATOR WILL BE CENTERED BETWEEN THE MARGINS
- MARGIN - THE PAGE INDICATOR WILL BE RIGHT JUSTIFIED.
- FACING - THE PAGE INDICATOR WILL BE RIGHT JUSTIFIED ON ODD NUMBERED PAGES AND LEFT JUSTIFIED ON EVEN NUMBERED PAGES.

THE PAGE STRING MAY BE ALTERED BY THE USER AT ANY TIME DURING THE PROCESSING. IF THE LENGTH OF THE PAGE INDICATOR (PAGE STRING AND PAGE NUMBER) EXCEEDS THE LINE WIDTH (SEE 'MARGIN'), TRUNCATION WILL OCCUR.

IF THE USER WISHES ONLY THE PAGE STRING TO BE PRINTED (WITH NO NUMBER), THE PAGE NUMBER MUST BE SET TO ZERO (SEE NEWPAGE).

IF THE USER WISHES ONLY THE PAGE NUMBER TO BE PRINTED (WITH NO STRING), THE STRING VALUE SHOULD BE SET TO NULL OR BLANKS.

3.21 PAPER (LENGTH=<NUM>/ WIDTH=<NUM>)

THE 'PAPER' COMMAND ALLOWS THE USER TO DEFINE THE LIMITS OF THE PRINT PAGE SIZE.

THE PAGE SIZE IS DEFINED BY THE 'LENGTH' AND 'WIDTH' PARAMETERS. 'LENGTH' DETERMINES THE MAXIMUM NUMBER OF LINES ON A PAGE INCLUDING THE 'TOPSPACE' AND 'BOTTOMSPACE' REQUIREMENTS. 'WIDTH' DETERMINES THE MAXIMUM NUMBER OF CHARACTERS WHICH CAN BE PRINTED ON A LINE. THE INITIAL VALUES ARE 'LENGTH' = 66 LINES, 'WIDTH' = 72 CHARACTERS.

NOTE THAT THE PAPER LENGTH IS RELATED TO THE TOP AND BOTTOM-OF-PAGE FORMAT. IF THE PAPER LENGTH MINUS THE 'A' AND 'B' PARAMETERS OF THE 'TOPSPACE' AND 'BOTTOMSPACE' COMMANDS IS LESS THAN 2, A LIMIT ERROR WILL RESULT.

NOTE ALSO THAT THE PAPER WIDTH IS RELATED TO THE LEFT AND RIGHT MARGINS. IF THE WIDTH IS SET TO A VALUE LESS THAN OR EQUAL TO THE SUM OF THE TWO MARGINS, A LIMIT ERROR WILL RESULT.

3.22 PAUSE

THE 'PAUSE' COMMAND PROVIDES FOR THE USE OF DISCONTINUOUS FORMS SUCH AS SHEETS OF TYPING PAPER.

ONCE ENTERED, THE 'PAUSE' COMMAND WILL CAUSE THE FORMATTER TO STOP AFTER A PAGE EJECT UNTIL THE USER ENTERS A CARRIAGE RETURN. THIS WILL ALLOW THE USER TO REMOVE THE FINISHED PAGE AND INSERT A NEW ONE.

3.23 REPEAT

THE 'REPEAT' COMMAND ALLOWS FOR REPETITION OF OUTPUT. THE 'BREAK' COMMAND IS EXECUTED TO OUTPUT ANY UNPRINTED TEXT. THE CURRENT TEXT FILE BEING READ WILL THEN BE SET TO THE FIRST RECORD FOR CONTINUED FORMATTING. THE PROCESS MUST BE TERMINATED BY PRESSING THE 'BREAK' KEY.

3.24 SKIP [<NUM>]

THE 'SKIP' COMMAND ALLOWS THE USER TO SKIP <NUM> NUMBER OF LINES

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. IF <NUM> IS SPECIFIED, <NUM> BLANK LINES WILL THEN BE PRINTED. IF NO PARAMETER IS GIVEN, THE DEFAULT IS ONE LINE. IF A NEW PAGE IS REACHED BEFORE ALL BLANK LINES ARE PRINTED, NO MORE BLANK LINES WILL BE PRINTED.

3.25 SPACING <NUM>

THE 'SPACING' COMMAND ALLOWS THE USER TO ALTER THE SPACING BETWEEN OUTPUT LINES.

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. FOLLOWING TEXT LINES WILL THEN BE SPACED ACCORDING TO THE <NUM> VALUE: <NUM>=1 FOR SINGLE SPACING, <NUM>=2 FOR DOUBLE SPACING, <NUM>=3 FOR TRIPLE SPACING, ETC.. THE FORMATTER IS INITIALLY SET FOR SINGLE SPACING.

3.26 SUPPRESS

THE 'SUPPRESS' COMMAND WILL SUPPRESS A LINE FEED ON A LINE TO ALLOW FOR UNDERLINING OR OVERLAYING OF PORTIONS OF A LINE. THE 'BREAK' COMMAND IS EXECUTED TO OUTPUT ANY UNPRINTED TEXT. THE NEXT LINE WILL THEN BE PRINTED WITHOUT A LINE FEED.

3.27 TAB (ON/OFF/
<CHAR>/
<NUM>[,<NUM>,<NUM>,<NUM>,<NUM>]]

THE 'TAB' COMMAND PROVIDES FOR THE TABULATION OF TEXT UNDER THE CONTROL OF USER DEFINED TAB POSITIONS AND THE TABULATOR CHARACTER.

WHEN THE 'TAB' FEATURE IS TURNED 'ON', THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. IN THE FOLLOWING LINES OF TEXT, EACH TIME A TAB CHARACTER IS FOUND, SPACES WILL BE INSERTED, REPLACING THE TAB CHARACTER, TO THE NEXT TAB POSITION. IF A NEXT TAB POSITION DOES NOT EXIST, NO TABULATION WILL OCCUR. THE TABULATOR WILL REMAIN ACTIVE UNTIL A 'TAB OFF' COMMAND IS RECEIVED.

THE 'TAB' COMMAND ALLOWS THE USER TO SET FROM 1 TO 5 TAB POSITIONS. THE <NUM> PARAMETERS REPRESENT THE DESIRED TAB POSITIONS RELATIVE TO THE BEGINNING OF THE LINE. IF MORE THAN FIVE POSITIONS ARE DECLARED IN ONE OR MORE 'TAB' COMMANDS, THE RIGHT-MOST TAB POSITION WILL BE LOST.

THE TAB CHARACTER, INITIALLY #, CAN BE CHANGED TO ANY CHARACTER DESIRED BY THE USER. IF NO PARAMETERS ARE GIVEN, ALL TAB POSITIONS ARE CLEARED.

EXAMPLE:

```
.TAB ON,/,8,12,15
```

TURNS THE TABULATOR ON, SPECIFIES / AS THE TAB CHARACTER TO RECOGNIZE, AND SETS TABS AT COLUMN POSITIONS 8, 12, AND 15 RELATIVE TO THE BEGINNING OF THE LINE (NOT THE MARGIN). NOTE THAT IF THE LEFT MARGIN WAS SET TO 10, THE FIRST TAB POSITION WOULD NEVER BE USED.

3.28 TEXT <FNAME>

THE 'TEXT' COMMAND ALLOWS THE USER TO SPECIFY A SOURCE FILE TO BE FORMATTED.

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. TEXT WILL THEN BE TAKEN FROM THE FILE <FNAME> UNTIL AN END-OF-FILE IS REACHED, OR ANOTHER 'TEXT' COMMAND IS ENCOUNTERED.

FILES MAY BE NESTED THROUGH USE OF THE 'TEXT' COMMAND TO A LEVEL OF 5 FILES LESS THE NUMBER OF UTILITY FILES USED. AN UNLIMITED NUMBER OF FILES CAN BE ACCEPTED IN A SERIAL MANNER BY INITIALLY USING A TEXT FILE CONTAINING ONLY 'TEXT' COMMANDS WHICH DIRECT THE FORMATTER TO OTHER TEXT FILES. REFER TO SECTION 1.2 FOR A MORE THOROUGH DISCUSSION OF FILES.

3.29 TOPSPACE (A=<NUM>/
B=<NUM>)

ON EACH PRINTED PAGE OF TEXT, A TOP-OF-PAGE FORMAT CAN BE DEFINED. THE 'TOPSPACE' COMMAND ALLOWS THE USER TO DEFINE THE NUMBER OF LINES SURROUNDING THE HEADING STRING. THE FORMAT CONSISTS OF:

(1) 'A' NUMBER OF BLANK LINES BETWEEN THE TOP OF THE PAGE AND THE HEADING STRING.
(2) A HEADING STRING (SEE 'HEADING').
(3) 'B' NUMBER OF BLANK LINES BETWEEN THE HEADING STRING AND THE FIRST LINE OF PRINTED TEXT.

THE 'A' PARAMETER DETERMINES THE NUMBER OF BLANK LINES WHICH WILL PRECEDE THE HEADING STRING. THE 'B' PARAMETER DETERMINES THE NUMBER OF BLANK LINES WHICH WILL BE PRINTED BETWEEN THE HEADING STRING AND THE FIRST TEXT LINE. THE INITIAL VALUES ARE A=3, B=3.

THE 'A' AND 'B' PARAMETERS ARE RELATED TO THE LENGTH OF THE PAGE (SEE THE 'PAPER' COMMAND) AND THE BOTTOM-OF-PAGE FORMAT (SEE THE 'BOTTOMSPACE' COMMAND). THE PAGE LENGTH MINUS THE 'A' AND 'B' PARAMETERS OF BOTH THE TOP AND BOTTOM-OF-PAGE FORMAT CANNOT BE LESS THAN 2. IF SO DEFINED, A LIMIT ERROR WILL RESULT.

3.30 UNDENT <NUM>

THE 'UNDENT' COMMAND ALLOWS THE USER TO CONCATENATE TWO LINES, INDENTING THE RESULT <NUM> CHARACTER POSITIONS LESS WITHOUT CHANGING THE LEFT MARGIN.

THE 'BREAK' COMMAND IS EXECUTED TO PRINT ANY UNPRINTED FORMATTED TEXT. THE NEXT TWO LINES ARE THEN CONCATENATED IN SUCH A WAY THAT THE FIRST LINE IS PRINTED <NUM> SPACES BEFORE THE LEFT MARGIN AND THE SECOND LINE IS PRINTED BEGINNING AT THE LEFT MARGIN.

NOTE:IF THE FIRST OF THE TWO LINES IS GREATER THAN <NUM>, IT WILL BE TRUNCATED AT THE LEFT MARGIN.

AS AN EXAMPLE, CONSIDER THE FOLLOWING TEXT INPUT IF THE LEFT MARGIN IS 5.

```
 .UNDENT 4
5.2
SUBPARAGRAPH HEADING
```

THE RESULT WILL BE:

```
5.2 SUBPARAGRAPH HEADING
```

APPENDIX A

COMMAND SUMMARY

	INITIAL VALUE	BREAK	CAN BE INITIALIZED
ADJUST (ON/OFF)	ON	YES	YES
BLANK <CHAR>	%	NO	YES
BOTTOMSPACE ([A=<NUM>]	3	NO	YES
[B=<NUM>])	3	NO	YES
BREAK		YES	NO
CENTER [<NUM>]	0	YES	NO
CHECK [LIST]	OFF	YES	YES
CONTROL <CHAR>	.	NO	YES
DEFINE <CNAME> = <STRING>		NO	YES
ENTER (<NUM>)	0	NO	YES
FILL (ON/OFF)	ON	YES	YES
FLAG (ON/OFF/	OFF	NO	YES
<CHAR>/	!	NO	YES
COUNT=<NUM>)	0	NO	YES
HEADING (ON/OFF/	OFF	NO	YES
CENTER/MARGIN/FACING/OPPOSED/	CENTER	NO	YES
<STRING>)		NO	YES
HOLD [<NUM>]	0	YES	NO
INDENT <NUM>	0	YES	NO
[FOR <NUM>]	0	YES	NO
MARGIN ([LEFT=<NUM>]	5	NO	YES
[RIGHT=<NUM>])	5	NO	YES
NEED <NUM>[,KEEP <NUM>]		YES	NO
NEWPAGE [<NUM>]		YES	NO
ODDPAGE		YES	NO
OPEN (HOLD FILE = <FNAME>/		NO	YES
NEED FILE = <FNAME>/		NO	YES
COMMAND FILE = <FNAME>)		NO	YES
PAGING (ON/OFF/	ON	NO	YES
CENTER/MARGIN/FACING/	MARGIN	NO	YES
TOP/BOTTOM/	TOP	NO	YES
<STRING>)	"PAGE "	NO	YES
PAPER (LENGTH=<NUM>/	66	NO	YES
WIDTH=<NUM>)	72	NO	YES
PAUSE	OFF	NO	YES
REPEAT		YES	NO
SKIP [<NUM>]		YES	NO
SPACING <NUM>	1	YES	YES
SUPPRESS		YES	YES
TAB [ON/OFF/	OFF	YES	YES
<CHAR>/	#	YES	YES
<NUM> [,<NUM>,<NUM>,<NUM>,<NUM>]]		YES	YES
TEXT <FNAME>		YES	YES
TOPSPACE (A=<NUM>/	3	NO	YES
B=<NUM>)	3	NO	YES
UNDENT <NUM>	0	YES	NO

APPENDIX B

ERROR MESSAGES

THE INTERPRETATION OF FORM2K COMMANDS CAN RESULT IN ONE OF TWO TYPES OF ERROR MESSAGES : SOFT AND HARD. EACH WILL AFFECT THE PRINTED OUTPUT IN A DIFFERENT MANNER.

B-1 SOFT ERRORS

SOFT ERROR MESSAGES ARE WARNINGS TO THE USER THAT THE EXPECTED RESULT WAS NOT ACHIEVED. A RESULT, HOWEVER, WILL STILL BE PRINTED AND PROCESSING WILL CONTINUE. SOFT ERROR MESSAGES INCLUDE:

```
*** POSITION PRECEDES COLUMN POSITION 1
*** 'NEED' EXCEEDS PAGE SIZE
*** RIGHT MARGIN EXCEEDED
*** LINE WIDTH EXCEEDED
*** LINE CANNOT BE FILLED
*** LINE CANNOT BE ADJUSTED
```

B-2 HARD ERRORS

HARD ERROR MESSAGES TERMINATE FORMATTER PROCESSING, THUS, MUST BE CORRECTED BEFORE A COMPLETE PRINTOUT CAN BE OBTAINED. THE 'CHECK' COMMAND CAN BE USED FOR THIS PURPOSE. HARD ERROR MESSAGES INCLUDE:

```
*** INVALID COMMAND
*** INVALID PARAMETER
*** DUPLICATE COMMAND NAME
*** FILE NESTING CAPACITY EXCEEDED
*** COMMAND FILE NOT OPEN
*** FILE OVERFLOW
*** LIMITS EXCEEDED
*** FILE ERROR : CODE =
```

HP7260
36888-18042

TITLE: SUBROUTINE TO INPUT CARDS

DESCRIPTION:

This subroutine is used to input string information from cards using the HP 7260A Optical Mark Reader.

INSTRUCTIONS:

A GOSUB 8000 will cause the subroutine to be executed. Input data consists of Z1 (whether to process condition codes or not) and Z3 (which option is to be executed). Output data consists of Z2 (the condition code) and Z$ (the data on the card). Z$ is defined to a length of 72 in the program and Y$ is defined to a length of 6. The various values for Z1, Z2, and Z3 are as follows:

Z1 whether to process condition codes or not
- 0=no
- 1=yes

Z2 condition codes
- 0=card read successfully
- 1=ready button was pushed
- 2=reader not ready
- 3=card hoppers either full or empty
- 4=pick fail
- 5=select-hoppers full
- 6=select success

Z3 option to be executed
- 0=line-feed and then demand-feed (single pick)
- 1=demand-feed (single pick)
- 2=retransmit with image-off
- 3=image-on demand-feed (single pick)
- 4=stop
- 5=abort
- 6=select

Equipment required is: TSB/2000E system, terminal, and HP 7260A Optical Mark Reader.

SPECIAL CONSIDERATIONS:

Each option produces an "X-on", yet options Z3=4, 5, or 6 do not go on to input data. If this program is used with either the teletype with paper tape reader and the HP 7260A reader, the options 4-6 should NOT be used.

In the image-on demand-feed mode, 80 column cards will be read as 164 characters which is too long for the computer's buffer and an error message will be produced. Only 40 column (or less) cards should be used with option Z3=3.

ACKNOWLEDGEMENTS:

Lary R. Smith
Livonia Public Schools

TITLE: TWO VARIABLE PLOT PROGRAM

PLOTXY
36888-18034

DESCRIPTION:

Plots one arbitrary array against another. Data to be plotted are generated or read by a user written program and passed to PLOTXY through a COM statement. Multiple copies of output may be generated and scales may be changed.

INSTRUCTIONS:

LIMITATIONS: Plots a maximum of 60 pairs of points.

BASIC PROGRAM OPERATION: The program does not generate or read data; the data to be plotted must be provided by a user written BASIC program which must conform to the following conventions:

(1) The first statement in the program should be:

```
10 COM X (61), Y (60)
```

(2) The 61st position in the X array should contain the number of points to be plotted (60 max).

(3) X (1), ..., X (n) should contain the values of the independent variable, and Y (1), ..., Y (n) should contain the values of the dependent variable. The sample problem shows a program to generate values of the standard normal probability distribution

$$Y = \frac{1}{\sqrt{2\pi}} \quad e^{-x^2/2}$$

for values of x from -3 to 2.9 in increments of .1. There are a total of 60 points generated.

Once the data generation program is entered and run, type:

GET - PLOTXY

Note: Do not scratch the data generation program before loading PLOTXY.

When the system responds with a carriage return, type RUN.

The program will plot the X and Y values generated or read by the user's program on a grid that is 40 print positions for X and 60 print positions for Y. Points on the graph are plotted as asterisks (*); if two points coincide, a number 2 is printed instead of an asterisk; if three points coincide, a number 3 is printed; if more than three points coincide, a plus sign (+) is printed. Once the graph is completed, the number of observations and the minimum and maximum values for X and Y are printed. The program will then ask: DO YOU WANT ANOTHER COPY, (1 = YES, 0 = NO)?

SPECIAL CONSIDERATIONS:

Data to be plotted must be passed to PLOTXY through a COM statement of the form:

```
1 CØM X(61), Y(60)
```

The number of points to be plotted must be stored in the 61'st position of the X array.

ACKNOWLEDGEMENTS:

Bill Jarosz
De Paul University

RUN

```
10   COM X[61],Y[60]
20   LET X[61]=60
30   LET Z=-3
40   LET C=1/SQR(2*3.14159)
50   FOR I=1 TO 60
60   LET X[I]=Z
70   LET Y[I]=C*EXP(-Z↑2/2)
80   LET Z=Z+.1
90   NEXT I
95   CHAIN "$PLOTXY"
100   END


RUN
BJ
```

```
I------------------------------------------------------------
I                             *
I                            ****
I                           *     *
I
I                         *        *
I                        *          *
I
I                       *            *
I
I                      *              *
I
I                     *                *
I
I
I                    *                  *
I
I                   *                    *
I
I                  *                      *
I
I
I                 *                        *
I
I                *                          *
I
I               *                            *
I
I              *                              *
I
I             *                                *
I
I            *                                  *
I           *                                    *
I          *                                      *
I
I         *                                        *
I       **                                          **
I      *                                              *
I    **                                                **
I2***                                                    ****
I------------------------------------------------------------
```

```
NO.OBS. 60
MIN. X=-3
MAX. X= 2.9
MIN. Y= 4.43185E-03
MAX. Y= .398943
```

```
DO YOU WANT ANOTHER COPY,(1=YES,Ø=NO)?1

NO OF LINES FOR Y-AXIS,(4Ø FOR SAME SCALE AS FIRST GRAPH)?3Ø

          I------------------------------------------------------------
          I                             *
          I                            ****
          I                           *     *
          I                         *        *
          I                        *          *
          I                       *            *
          I
          I                      *              *
          I
          I                     *                *
          I                    *                  *
          I
          I                   *                    *
          I
          I                  *                      *
          I
          I                 *                        *
          I                *                          *
          I
          I               *                            *
          I              *                              *
          I
          I             *                                *
          I            *                                  *
          I           *                                    *
          I          *                                      *
          I         *                                        *
          I       **                                          **
          I     **                                              **
          I2****                                                  *****
          I------------------------------------------------------------

          NO.OBS. 6Ø
          MIN. X=-3
          MAX. X= 2.9
          MIN. Y= 4.43185E-Ø3
          MAX. Y= .398943

DO YOU WANT ANOTHER COPY,(1=YES,Ø=NO)?Ø

DONE
```

TITLE
36114

TITLE: CHARACTER GENERATION

DESCRIPTION:

This program generates and positions letters, numbers and some symbols for the Model 7200A Graphic Plotter. There are two programs in this package: TITLE and CHRGEN.

INSTRUCTIONS:

CRE-CHAR,22
RUN CHRGEN to create the file "CHAR".
Get and RUN "TITLE".

Program responds with:

Instructions?	Yes or No (Answer "Yes" the first time you RUN)
Graph Size in Major Divisions:	
Width?	Major Divisions
Height?	Major Divisions
Inputs Desired:	1 or 2 or 3 or 4
Size?	0.1 Major Divisions
Location (X,Y)?	Major Divisions from lower left
Angle?	Degrees from horizontal
TITLE:	
More (1234)?	No or 1 or 2 or 3 or 4

ACKNOWLEDGEMENTS: Hewlett-Packard/San Diego

RUN

```
RUN
TITLE

INSTRUCTIONS?YES

YOU WILL BE ASKED FOR INPUTS:

1=SIZE,2=LOCATION,3=ANGLE,4=TITLE(1234 FOR ALL):

ANY COMBINATION OF THE FOUR NUMBERS MAY BE USED,
SUCH AS '24' FOR LOCATION AND TITLE ONLY.

SIZE IS 0.1 MAJOR DIVISION PER LETTER MULTIPLIED BY
THE NUMBER YOU ENTER; ENTERING '20' WOULD PRODUCE LETTERS
OCCUPYING 2X2 MAJOR DIVISIONS. SIZE IS SET AT '5' IF
YOU DO NOT INPUT.

LOCATION IS THE POSITION ON THE GRAPH IN MAJOR DIVISIONS,
MEASURED FROM THE LOWER LEFT OF THE GRAPH; 5,5 WOULD
BE THE CENTER OF A 10X10 DIVISION GRAPH.

ANGLE IS THE ANGLE IN DEGREES FROM HORIZONTAL.

TITLE IS THE TEXT YOU WISH PRINTED, UP TO 72 CHARACTERS.
A CONTROL 'O' PRODUCES A CARRIAGE RETURN AND LINEFEED
ON THE GRAPH. A CONTROL 'N' PRODUCES A LINEFEED ONLY.
A CONTROL 'O' AS THE LAST CHARACTER PRODUCES A CARRIAGE
RETURN, LINEFEED, AND A REQUEST FOR MORE INPUT.

GRAPH SIZE IN MAJOR DIVISIONS:
WIDTH ?15
HEIGHT ?10
INPUTS DESIRED:
1=SIZE,2=LOCATION,3=ANGLE,4=TITLE (1234 FOR ALL):?24
LOCATION (X,Y)?0,8
TITLE:
?ABCDEFGHIJKLMNOPQRSTUVWXYZ
?1234567890
?↑<>?/.,+-=*)(%$#[]
PLTL
MORE (1234)?234
LOCATION (X,Y)?8,3
ANGLE?30
TITLE:
?HP GRAPHICS
PLTL
MORE (1234)?NO

DONE
```

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
1 2 3 4 5 6 7 8 9 0
↑ < > ? / . , + - = *) (% $ # [ ]

HP GRAPHICS

VOLUME II
CONTENTS

300 MATH AND NUMERICAL ANALYSIS

300

VOLUME II
CONTENTS (Continued)

400 PROBABILITY AND STATISTICS

400

VOLUME II
CONTENTS (Continued)

400 PROBABILITY AND STATISTICS (Continued)

VOLUME II
CONTENTS (Continued)

500 SCIENTIFIC AND ENGINEERING APPLICATIONS

500

BINO
36888-18029

TITLE: BINOMIAL FUNCTION EXPANSION

DESCRIPTION: This program expands any binomial of form $(AX + BY)^N$, where N can range from 1 to 15.

INSTRUCTIONS: Instructions are contained in program.

ACKNOWLEDGEMENTS: Clifford E. Cuellar, Jr.
Reichhold Chemicals Inc.

RUN

```
RUN
BINO

PROGRAM COMPUTES COEFFICIENTS FOR POLYNOMIALS OF FORM
(AX+BY)↑N, WHERE A & B ARE +- NUMBERS AND N IS A
POSITIVE INTEGER BETWEEN 1 AND 15
YES   IS CORRECT RESPONSE TO LAST QUESTIONS

INPUT A,B,N?3,2,3
 27    X↑ 3     + 54    X↑ 2     Y↑ 1     + 36    X↑ 1     Y↑ 2     + 8     Y↑
 3
NEXT HIGHER DEGREE?YES
 81    X↑ 4     + 216  X↑ 3     Y↑ 1     + 216  X↑ 2     Y↑ 2     + 96    X↑
 1     Y↑ 3     + 16    Y↑ 4
NEXT HIGHER DEGREE?NO
NEW VALUES?YES
INPUT A,B,N?4,2,2
 16    X↑ 2     + 16    X↑ 1     Y↑ 1     + 4     Y↑ 2
NEXT HIGHER DEGREE?YES
 64    X↑ 3     + 96    X↑ 2     Y↑ 1     + 48    X↑ 1     Y↑ 2     + 8     Y↑
 3
NEXT HIGHER DEGREE?NO
NEW VALUES?NO

DONE
```

EXTADD
36888-18040

TITLE: 40 DIGIT PRECISION STRING ADDITION

DESCRIPTION:

EXTADD is an extraction from EXTPRE, 40 DIGIT PRECISION STRING ARITHMETIC. It has been found from several users of EXTPRE that 90% of the programs written to utilize EXTPRE need only the part which does addition, and that Z8, which returns the number of digits in the result, is not needed. This subroutine has been written to fulfill that need, and runs about 7% to 9% faster than EXTPRE.

INSTRUCTIONS:

Variables used are: A, A1, A2, A3, A4, A5, A6, A9
B, C, D, A(*), B(*), C(*)
D$, Y$, Z$

This subroutine begins at statement number 9000. It is intended to be appended to a user's program. The subroutine performs arithmetic operations on the contents of two strings, Y$ and Z$. The result is returned in Z$. Leading or embedded blanks, a minus sign, commas, and a decimal point may or may not be contained in Y$ and/or Z$ when they are passed to the subroutine.

When the subroutine is called, the variable D must contain a number between 0 and 6 which indicates the largest number of digits to the right of the decimal point which the user desires.

An example of a calling sequence for this subroutine is as follows:

```
211  Y$ = "36243163.123"
212  Z$ = "1234567.89"
213  D = 3
214  Z9 = 2
215  GOSUB 9000
216  PRINT Z$
```

Statement 216 will cause 361188595.233 to be printed on the user's terminal.

SPECIAL CONSIDERATIONS:

A marginal increase in subroutine execution speed may be achieved by removing line 9001 D$="0123456789" and inserting it at the top of the calling program (it need only be done once, instead of for each call, as is now done, if the user does not use it anywhere else in his program.)

It is not necessary to set Z9 to indicate the type of operation as in EXTPRE, since EXTADD does only addition. This instructions are eliminated in both the subroutine and its calling program.

If more than addition is to be done, use EXTPRE.

ACKNOWLEDGEMENTS:

Stephen S. MacKenzie
Hewlett-Packard/Atlanta, Georgia

EXTPRE
36144

TITLE: 40-DIGIT PRECISION MATHEMATICS

DESCRIPTION: This time-shared BASIC subroutine is designed to be appended to a time-shared BASIC program to enable a user to do calculations with up to 40 digits of precision.

INSTRUCTIONS: Variables used are:

A, A1, A2, A3, A4, A5, A6, A9
B, B1, B2, B3, B4, B5
C, C1
D, Z8; Z9
A(*), B(*), C(*)
D$, Y$, Z$

A marginal increase in subroutine execution speed may be achieved by removing line 9001 D$="0123456789" and inserting it at the top of the calling program (it need only be done once, if the user does not use it anywhere else in his program.)

An additional 5% to 7% improvement in speed can be achieved by deleting lines 9032-9036 and changing line 9031 to REM. This should be done only if the variable Z8, which returns the number of digits in the result, is not needed by the user.

INSTRUCTIONS continued on page 2

SPECIAL CONSIDERATIONS: If only addition is to be done, the user should use the subroutine EXTADD.

ACKNOWLEDGEMENTS: David Sanders (Original)
Hewlett-Packard/Cupertino
Stephen MacKenzie (Modification, Rev. B)
Hewlett-Packard/Atlanta, Georgia

GSIMEQ
36547

TITLE: SIMULTANEOUS LINEAR EQUATIONS

DESCRIPTION:

This program allows the user to specify a set of simultaneous linear equations in standard algebraic format. Some of the variables may be exogeneous (i.e., determined outside the system of equations). There must be as many endogeneous variables (i.e., those determined within the system of equations) as there are linear equations.

INSTRUCTIONS:

Each variable must be represented by a simple alphabetic character. As many as 20 variables can be included. All parameters must be specified explicitly. The program solves the system then prints the solution equations.

ACKNOWLEDGEMENTS:

Graduate School of Business
Stanford University

RUN

```
RUN
GSIMEQ

DO YOU WANT INSTRUCTIONS?YES

I WILL ASK YOU FOR EXOGENEOUS VARIABLES AND
ENDOGENEOUS VARIABLES. EACH VARIABLE CONSISTS
OF A SINGLE ALPHABETIC CHARACTER. YOU MAY SEPARATE
VARIABLES WITH COMMAS OR BLANKS -- FOR EXAMPLE:
  EXOGENOUS VARIABLES: G,I
IF THERE ARE NO EXOGENEOUS VARIABLES, ANSWER --
   EXOGENEOUS VARIABLES: NONE
I WILL THEN ASK YOU FOR YOUR EQUATIONS.
YOU MAY USE ANY LINEAR EQUATION WITH CONSTANTS
(NOT VARIABLES) AS PARAMETERS.
MULTIPLICATION MAY BE EXPLICIT (*) OR IMPLICIT.
DO NOT BEGIN AN EQUATION WITH A MINUS SIGN.
DO NOT PLACE A MINUS SIGN IMMEDIATELY AFTER '='.
HERE ARE SOME EXAMPLES --
  C+I+G=Y
  C=.9Y
  I=100-.2*Y
HERE GOES --

EXOGENEOUS VARIABLES: G
ENDOGENEOUS VARIABLES: C,I,Y

I AM GOING TO ASK YOU FOR  3     EQUATIONS

EQUATION: C=.7Y+50
EQUATION: I=.1Y-10
EQUATION: C+I+G=Y

C  =  190.00 +    3.50*G
I  =   10.00 +    0.50*G
Y  =  200.00 +    5.00*G

DONE
```

LOGICK
36888-18015

TITLE: BOOLEAN ALGEBRA EVALUATOR

DESCRIPTION:

This program analyzes Boolean algebraic expressions, checks for proper syntax, and creates a truth table for any legal expression. If it finds an error in the syntax, it exits to an internal error routine that prints an appropriate error message, and returns to statement entry mode.

The program operates by translating the Boolean statement into a psuedo-machine-language program, stored in matrix P, and then running it for every combination of truth-values possible. Once this 'mini-program' is compiled, execution is very fast.

INSTRUCTIONS:

Legal variables are the letters A to Z. Legal operators are:

+ 'OR' inclusive or
* 'AND' and
- 'NOT' not

The program will accept any statement that is a legal combination of variables and operators. The types of errors that will generate error messages are shown in the sample RUN. The program will accept any number of levels of parentheses, and will evaluate the statement in the hierarchal order of those levels. However, within any level, evaluation is strictly from left to right.

Other than the above considerations, the program should be self-explanatory.

SPECIAL CONSIDERATIONS:

Truth table output of '0' and '1' may be changed to 'F' and "T" by changing D$ to "FT" (line 80). The logical operator * + - may be changed by altering the appropriate characters of B$ in line 90.

FOR INSTRUCTIONAL PURPOSES
Suitable Course(s): Intro Logic, Intro Programming, Digital Circuit Design, Boolean Algebra

Student Background Required: Boolean algebra.

The program was written as part of a directed study in Elementary Logic. It was used to quickly check elaborate theorems. The program could be used in conjunction with most any modern text on symbolic logic or Boolean algebra.

ACKNOWLEDGEMENTS: A. B. Jensen
MacMurray College

RUN

```
RUN
LOGICK

THIS PROGRAM WILL PRODUCE A TRUTH TABLE FOR ANY BOOLEAN
ALGEBRA EXPRESSION THAT WILL FIT ON A 72-CHARACTER LINE.
 - = NOT
+ = IOR
* = AND
EVALUATION IS FROM LEFT TO RIGHT.



YOUR LOGICAL STATEMENT IS:
?A&B
ILLEGAL CHARACTER
     '&'
YOUR LOGICAL STATEMENT IS:
?A B
OPERATION ERROR: TWO ADJACENT SYMBOLS
YOUR LOGICAL STATEMENT IS:
?A*+B
OPERATION ERROR: TWO ADJACENT OPERATORS
YOUR LOGICAL STATEMENT IS:
?A-B
OPERATION ERROR: MISPLACED NOT
YOUR LOGICAL STATEMENT IS:
?A*((B+C)*D
UNEQUAL NUMBER OF RIGHT AND LEFT PARENTHESES
YOUR LOGICAL STATEMENT IS:
?A-+B
OPERATION ERROR: MISPLACED NOT
YOUR LOGICAL STATEMENT IS:
?A+A

TRUTH TABLE FOR A+A   :
 A    T/F
----------
 Ø     Ø
 1     1

YOUR LOGICAL STATEMENT IS:
?A+B

TRUTH TABLE FOR A+B   :
 A B    T/F
------------
 Ø Ø     Ø
 Ø 1     1
 1 Ø     1
 1 1     1

YOUR LOGICAL STATEMENT IS:
?A*B

TRUTH TABLE FOR A*B   :
 A B    T/F
------------
 Ø Ø     Ø
 Ø 1     Ø
 1 Ø     Ø
 1 1     1

YOUR LOGICAL STATEMENT IS:
?-(A*B)

TRUTH TABLE FOR -(A*B)   :
 A B    T/F
------------
 Ø Ø     1
 Ø 1     1
 1 Ø     1
 1 1     Ø
```

```
YOUR LOGICAL STATEMENT IS:
?A*(B+C)

TRUTH TABLE FOR A*(B+C)   :
 A B C     T/F
--------------
 Ø Ø Ø      Ø
 Ø Ø 1      Ø
 Ø 1 Ø      Ø
 Ø 1 1      Ø
 1 Ø Ø      Ø
 1 Ø 1      1
 1 1 Ø      1
 1 1 1      1

YOUR LOGICAL STATEMENT IS:
?
DONE
```

POWER2
36888-18009

TITLE: POWERS OF TWO TABLES

DESCRIPTION: This program is intended to be used when a Powers of Two Table is needed (as is often the case for an Assembler programmer). It also demonstrates extended precision integer arithmetic through the use of arrays. The 'RUN' is paginated for 11 inch sheets (a header line is printed every 66 lines).

INSTRUCTIONS: Just GET and RUN program.

SPECIAL CONSIDERATIONS: Array B and String B$ have dimensions corresponding to the accuracy limits. One can have more precision through the manipulation of the dimensions and certain counters. As the program is written it will print up to 2^{162}.

ACKNOWLEDGEMENTS: Mr. Leslie Citrome (Student)
West Hill High School of Montreal

RUN

```
RUN
POWER2
```

POWERS OF TWO TABLE

```
1       2
2       4
3       8
4       16
5       32
6       64
7       128
8       256
9       512
10      102 4
11      204 8
12      409 6
13      819 2
14      163 84
15      327 68
16      655 36
17      131 072
18      262 144
19      524 288
20      104 8576
21      209 7152
22      419 4304
23      838 8608
24      167 7721 6
25      335 5443 2
26      671 0886 4
27      134 2177 28
28      268 4354 56
29      536 8709 12
30      107 3741 824
31      214 7483 648
32      429 4967 296
33      858 9934 592
34      171 7986 9184
35      343 5973 8368
36      687 1947 6736
37      137 4389 5347 2
38      274 8779 0694 4
39      549 7558 1388 8
40      109 9511 6277 76
41      219 9023 2555 52
42      439 8046 5111 04
43      879 6093 0222 08
44      175 9218 6044 416
45      351 8437 2088 832
46      703 6874 4177 664
47      140 7374 8835 5328
48      281 4749 7671 0656
49      562 9499 5342 1312
50      112 5899 9068 4262 4
51      225 1799 8136 8524 8
52      450 3599 6273 7049 6
53      900 7199 2547 4099 2
54      180 1439 8509 4819 84
55      360 2879 7018 9639 68
56      720 5759 4037 9279 36
57      144 1151 8807 5855 872
```

POWERS OF TWO TABLE

58	288 2303 7615 1711 744
59	576 4607 5230 3423 488
60	115 2921 5046 0684 6976
61	230 5843 0092 1369 3952
62	461 1686 0184 2738 7904
63	922 3372 0368 5477 5808
64	184 4674 4073 7095 5161 6
65	368 9348 8147 4191 0323 2
66	737 8697 6294 8382 0646 4
67	147 5739 5258 9676 4129 28
68	295 1479 0517 9352 8258 56
69	590 2958 1035 8705 6517 12
70	118 0591 6207 1741 1303 424
71	236 1183 2414 3482 2606 848
72	472 2366 4828 6964 5213 696
73	944 4732 9657 3929 0427 392
74	188 8946 5931 4785 8085 4784
75	377 7893 1862 9571 6170 9568
76	755 5786 3725 9143 2341 9136
77	151 1157 2745 1828 6468 3827 2
78	302 2314 5490 3657 2936 7654 4
79	604 4629 0980 7314 5873 5308 8
80	120 8925 8196 1462 9174 7061 76
81	241 7851 6392 2925 8349 4123 52
82	483 5703 2784 5851 6698 8247 04
83	967 1406 5569 1703 3397 6494 08
84	193 4281 3113 8340 6679 5298 816
85	386 8562 6227 6681 3359 0597 632
86	773 7125 2455 3362 6718 1195 264
87	154 7425 0491 0672 5343 6239 0528
88	309 4850 0982 1345 0687 2478 1056
89	618 9700 1964 2690 1374 4956 2112
90	123 7940 0392 8538 0274 8991 2422 4
91	247 5880 0785 7076 0549 7982 4844 8
92	495 1760 1571 4152 1099 5964 9689 6
93	990 3520 3142 8304 2199 1929 9379 2
94	198 0704 0628 5660 8439 8385 9875 84
95	396 1408 1257 1321 6879 6771 9751 68
96	792 2816 2514 2643 3759 3543 9503 36
97	158 4563 2502 8528 6751 8708 7900 672
98	316 9126 5005 7057 3503 7417 5801 344
99	633 8253 0011 4114 7007 4835 1602 688
100	126 7650 6002 2822 9401 4967 0320 5376
101	253 5301 2004 5645 8802 9934 0641 0752
102	507 0602 4009 1291 7605 9868 1282 1504
103	101 4120 4801 8258 3521 1973 6256 4300 8
104	202 8240 9603 6516 7042 3947 2512 8601 6
105	405 6481 9207 3033 4084 7894 5025 7203 2
106	811 2963 8414 6066 8169 5789 0051 4406 4
107	162 2592 7682 9213 3633 9157 8010 2881 28
108	324 5185 5365 8426 7267 8315 6020 5762 56
109	649 0371 0731 6853 4535 6631 2041 1525 12
110	129 8074 2146 3370 6907 1326 2408 2305 024
111	259 6148 4292 6741 3814 2652 4816 4610 048
112	519 2296 8585 3482 7628 5304 9632 9220 096
113	103 8459 3717 0696 5525 7060 9926 5844 0192
114	207 6918 7434 1393 1051 4121 9853 1688 0384

POWERS OF TWO TABLE

115	415	3837	4868	2786	2102	8243	9706	3376	0768				
116	830	7674	9736	5572	4205	6487	9412	6752	1536				
117	166	1534	9947	3114	4841	1297	5882	5350	4307	2			
118	332	3069	9894	6228	9682	2595	1765	0700	8614	4			
119	664	6139	9789	2457	9364	5190	3530	1401	7228	8			
120	132	9227	9957	8491	5872	9038	0706	0280	3445	76			
121	265	8455	9915	6983	1745	8076	1412	0560	6891	52			
122	531	6911	9831	3966	3491	6152	2824	1121	3783	04			
123	106	3382	3966	2793	2698	3230	4564	8224	2756	608			
124	212	6764	7932	5586	5396	6460	9129	6448	5513	216			
125	425	3529	5865	1173	0793	2921	8259	2897	1026	432			
126	850	7059	1730	2346	1586	5843	6518	5794	2052	864			
127	170	1411	8346	0469	2317	3168	7303	7158	8410	5728			
128	340	2823	6692	0938	4634	6337	4607	4317	6821	1456			
129	680	5647	3384	1876	9269	2674	9214	8635	3642	2912			
130	136	1129	4676	8375	3853	8534	9842	9727	0728	4582	4		
131	272	2258	9353	6750	7707	7069	9685	9454	1456	9164	8		
132	544	4517	8707	3501	5415	4139	9371	8908	2913	8329	6		
133	108	8903	5741	4700	3083	0827	9874	3781	6582	7665	92		
134	217	7807	1482	9400	6166	1655	9748	7563	3165	5331	84		
135	435	5614	2965	8801	2332	3311	9497	5126	6331	0663	68		
136	871	1228	5931	7602	4664	6623	8995	0253	2662	1327	36		
137	174	2245	7186	3520	4932	9324	7799	0050	6532	4265	472		
138	348	4491	4372	7040	9865	8649	5598	0101	3064	8530	944		
139	696	8982	8745	4081	9731	7299	1196	0202	6129	7061	888		
140	139	3796	5749	0816	3946	3459	8239	2040	5225	9412	3776		
141	278	7593	1498	1632	7892	6919	6478	4081	0451	8824	7552		
142	557	5186	2996	3265	5785	3839	2956	8162	0903	7649	5104		
143	111	5037	2599	2653	1157	0767	8591	3632	4180	7529	9020	8	
144	223	0074	5198	5306	2314	1535	7182	7264	8361	5059	8041	6	
145	446	0149	0397	0612	4628	3071	4365	4529	6723	0119	6083	2	
146	892	0298	0794	1224	9256	6142	8730	9059	3446	0239	2166	4	
147	178	4059	6158	8244	9851	3228	5746	1811	8689	2047	8433	28	
148	356	8119	2317	6489	9702	6457	1492	3623	7378	4095	6866	56	
149	713	6238	4635	2979	9405	2914	2984	7247	4756	8191	3733	12	
150	142	7247	6927	0595	9881	0582	8596	9449	4951	3638	2746	624	
151	285	4495	3854	1191	9762	1165	7193	8898	9902	7276	5493	248	
152	570	8990	7708	2383	9524	2331	4387	7797	9805	4553	0986	496	
153	114	1798	1541	6476	7904	8466	2877	5559	5961	0910	6197	2992	
154	228	3596	3083	2953	5809	6932	5755	1119	1922	1821	2394	5984	
155	456	7192	6166	5907	1619	3865	1510	2238	3844	3642	4789	1968	
156	913	4385	2333	1814	3238	7730	3020	4476	7688	7284	9578	3936	
157	182	6877	0466	6362	8647	7546	0604	0895	3537	7456	9915	6787	2
158	365	3754	0933	2725	7295	5092	1208	1790	7075	4913	9831	3574	4
159	730	7508	1866	5451	4591	0184	2416	3581	4150	9827	9662	7148	8
160	146	1501	6373	3090	2918	2036	8483	2716	2830	1965	5932	5429	76
161	292	3003	2746	6180	5836	4073	6966	5432	5660	3931	1865	0859	52
162	584	6006	5493	2361	1672	8147	3933	0865	1320	7862	3730	1719	04

DONE

SIMPLX
36888-18030

TITLE: SOLVES LINEAR PROGRAMS (CONDENSED TABLEAU METHOD)

DESCRIPTION:

This program solves linear programs and matrix games and finds the best uniform solution for linear equation.

For linear programs, the data consists of

(1) greater than inequalities of the form

$$a_1 x_1 + \ldots + a_n x_n + b \geq 0 ,$$

(2) less than inequalities of the form

$$a_1 x_1 + \ldots + a_n x_n + b \geq 0 ,$$

(3) equalities of the form

$$a_1 x_1 + \ldots + a_n x_n + b = 0 ,$$

and

(4) a linear function of the form

$$c_1 x_1 + \ldots + c_n x_n + d = w$$

INSTRUCTIONS:

After the user has entered the coefficients of the linear constraints and the coefficients of the linear function, the computer finds the solution of the linear program as well as the solution to the dual program.

To avoid cycling problems, the computer randomly chooses pivot spots when two or more coordinates satisfy the rules of the simplex algorithm.

For matrix games, the user enters his matrix and the computer converts the problem into a linear program and solves it. The optimal strategy for the row and columns players are then printed.

The maximum possible size of the input data is a 20 x 20 matrix for the linear programs and a 19 x 19 matrix for the matrix games.

The program will find integer solutions if the constraints are given with integer coefficients.

Instructions continued on following page.

ACKNOWLEDGEMENTS: Donald E. Ramirez
University of Virginia

INSTRUCTIONS continued

PROBLEM: (See first sample RUN)

Maximize $x_1 + x_2 + x_3 + x_4 + 0$

Subject to
$$\begin{cases} 3x_1 + 4x_2 + 5x_3 + 6x_4 - 7 \geq 0 \\ 4x_1 + 5x_2 + 6x_3 + 7x_4 - 8 \geq 0 \\ x_1 + 2x_2 + 3x_3 + 4x_4 - 5 \geq 0 \\ 2x_1 + 3x_2 + 4x_3 + 5x_4 - 6 = 0 \\ x_1, x_2, x_3, x_4, \geq 0 \end{cases}$$

PROBLEM: (See second sample RUN)

Maximize $2x_1 + 5x_2 + x_3 - 1000 \geq 0$

Subject to
$$\begin{cases} x_1 + 2x_2 + 4x_3 - 3000 \geq 0 \\ 3x_1 + x_2 + 6x_3 - 2000 = 0 \\ x_1, x_2, x_3 \geq 0 \\ x_1, x_2, x_3 \ \underline{\text{INTEGRAL}} \end{cases}$$

PROBLEM: (See third sample RUN)

Find the optional strategies for the matrix game

$$\begin{bmatrix} 2 & 4 & -5 & 6 \\ -3 & 4 & 4 & 6 \\ 0 & 1 & 3 & 2 \end{bmatrix}$$

PROBLEM: (See fourth sample RUN)

Find the best uniform solution of

$$\begin{aligned} x_1 + x_2 &= 3 \\ x_1 - x_2 &= 1 \\ x_1 + 2x_2 &= 7 \\ 2x_1 + 4x_2 &= 11.1 \\ 2x_1 + x_2 &= 6.9 \\ 3x_1 + x_2 &= 7.2 \end{aligned}$$

RUN

```
RUN


SIMPLX

THIS PROGRAM SOLVES LINEAR PROGRAMS AND MATRIX GAMES
AND FINDS THE BEST UNIFORM SOLUTION FOR LINEAR EQUATIONS
TYPE LP OR MG OR BS?LP
A PROGRAM TO MAXIMIZE OR MINIMIZE A LINEAR FUNCTION
SUBJECT TO LINEAR CONSTRAINTS
ENTER MAX OR MIN ?MAX
DO YOU WANT TO SEE THE PIVOT STEPS (Y OR N)?N
ENTER NUMBER OF VARIABLES, EQUALITIES (=Ø) ?4,1
ENTER NUMBER OF INEQUALITIES OF THE FORM >=Ø,<=Ø?2,1
ENTER THE SIMPLEX TABLEAU ROW BY ROW - GREATER THAN'S FIRST,
LESS THAN'S NEXT, EQUALITIES NEXT, AND THE LINEAR FUNCTION LAST
?3,4,5,6,-7
??4,5,6,7,-8
??1,2,3,4,-5
??2,3,4,5,-6
??1,1,1,1,Ø
INITIAL TABLEAU (Y OR N)?N
THE MAXIMUM OF THE LINEAR FUNCTION IS  3.

THE SOLUTION OCCURS AT
( 3.          , Ø     , Ø     , Ø     )

THE DUAL SOLUTION OCCURS AT
( Ø     , Ø     , Ø     , Ø     )

DO YOU WANT SOLUTIONS TO BE MORE INTEGRAL (Y OR N)?N


DONE




RUN
SIMPLX

THIS PROGRAM SOLVES LINEAR PROGRAMS AND MATRIX GAMES
AND FINDS THE BEST UNIFORM SOLUTION FOR LINEAR EQUATIONS
TYPE LP OR MG OR BS?LP
A PROGRAM TO MAXIMIZE OR MINIMIZE A LINEAR FUNCTION
SUBJECT TO LINEAR CONSTRAINTS
ENTER MAX OR MIN ?MIN
DO YOU WANT TO SEE THE PIVOT STEPS (Y OR N)?N
ENTER NUMBER OF VARIABLES, EQUALITIES (=Ø) ?3,1
ENTER NUMBER OF INEQUALITIES OF THE FORM >=Ø,<=Ø?1,1
ENTER THE SIMPLEX TABLEAU ROW BY ROW - GREATER THAN'S FIRST,
LESS THAN'S NEXT, EQUALITIES NEXT, AND THE LINEAR FUNCTION LAST
?2,5,1,-1ØØØ
??1,2,4,-3ØØØ
??3,1,6,-2ØØØ
??16,32,23,Ø
INITIAL TABLEAU (Y OR N)?N
THE MINIMUM OF THE LINEAR FUNCTION IS  9666.67

THE SOLUTION OCCURS AT
( 444.444     , Ø     , 111.111     )

THE DUAL SOLUTION OCCURS AT
( 3     , Ø     , Ø     )

DO YOU WANT SOLUTIONS TO BE MORE INTEGRAL (Y OR N)?Y

PROGRAM ASSUMES ALL VARIABLES ARE INTEGER VARIABLES AND ADDS
A CUTTING PLANE ON THE VARIABLE WITH THE LARGEST FRACTIONAL PART.
```

```
THE MINIMUM OF THE LINEAR FUNCTION IS  9694

THE SOLUTION OCCURS AT
( 438.        , 1.99988     , 114.        )

THE DUAL SOLUTION OCCURS AT
( 0     , 0     , 0     , 8.99999     , 35.          )
SOLUTION IS NEARLY INTEGRAL.

DONE



RUN
SIMPLX

THIS PROGRAM SOLVES LINEAR PROGRAMS AND MATRIX GAMES
AND FINDS THE BEST UNIFORM SOLUTION FOR LINEAR EQUATIONS
TYPE LP OR MG OR BS?MG
A PROGRAM TO SOLVE MATRIX GAMES
DO YOU WANT TO SEE THE PIVOT STEPS (Y OR N)?N
ENTER THE NUMBER OF ROWS,COLUMNS?3,4
ENTER THE MATRIX ROW BY ROW
?2,4,-5,6
??-3,4,4,6
??0,1,3,2
MATRIX IS
 2      4     -5      6

-3      4      4      6

 0      1      3      2

THE VALUE OF THE MATRIX GAME IS  .6

THE OPTIMAL STRATEGY FOR THE ROW PLAYER IS
( .3          , 0     , .7           )

THE OPTIMAL STRATEGY FOR THE COLUMN PLAYER IS
( .8          , 0     , .2           , 0     )

DONE



RUN
SIMPLX

THIS PROGRAM SOLVES LINEAR PROGRAMS AND MATRIX GAMES
AND FINDS THE BEST UNIFORM SOLUTION FOR LINEAR EQUATIONS
TYPE LP OR MG OR BS?BS
ENTER NUMBER OF EQUATIONS, VARIABLES?6,2
ENTER THE EQUATIONS IN THE FORM A*X1+B*X2=C
?1,1,3,1,-1,1,1,2,7,2,4,11.1,2,1,6.9,3,1,7.2
MINIMAX DEVIATION IS  1
THE SOLUTION OCCURS AT
( 2     , 2     )

DEVIATIONS ARE
 1     -1     -1      .9          -.9            .8
DONE
```

TITLE: COMPLEX NUMBER CALCULATOR

UHCX
36888-18005

DESCRIPTION:

UHCX is a calculator program which permits the operator to obtain functions of complex numbers and to perform complex arithmetic operations using numbers of the form '$a + bi$'. In the calculator the form of the numbers is described in the more common engineering terminology of 'R + JX' where 'R' is the real portion of the argument, 'X' is the imaginary portion and 'J' = $\sqrt{-1}$. The user may type functions and/or arithmetic operations with parentheses and functions nested as deeply as desired. The answer is outputted both in rectangular form (R + JX) and in polar form with 'Z' equal to the absolute value of the magnitude and the angle is expressed in both radians and degrees. In construction, the calculator program consists of syntax error check routines followed by a reverse Polish conversion program which results in the real and imaginary arguments being placed in push-up stacks and a Polish execution stack being generated for the operators. This is followed by an interpreter which executes from the Polish stack in accordance with the hierarchy of priorities assigned to the operators and functions. Instructions are provided in the program for use including a list of the functions implemented and sample inputs. Blanks are ignored in the input string, however, the input string length is restricted to 72 characters. Only complex arguments are permitted, but either portion of the argument may be made zero.

INSTRUCTIONS:

Highest priority is assigned to the unary operator. Parentheses may be used to modify priority as operations with the parenthesis are evaluated as the second priority. The next priority level is assigned to conversion instructions. 'CONR' (convert from polar to rectangular using radians for the angle argument) uses the general form CONR <magnitude> , < angle > . COND < magnitude >, < angle > also converts from polar to rectangular form, however the argument of the angle must be in degrees. Conversion from rectangular to polar may be accomplished simply by typing the complex numbers such as '3 + J4' or using the conversion command CONP < magnitude >, < angle > .

The next priority of execution is assigned to functions including the vertical arrow ' ' (power/root). These functions are sine (SIN), cosine (COS), tangent (TAN), hyperbolic sine (HSIN), hyperbolic cosine (HCOS), hyperbolic tangent (HTAN), square root (SQR), log base e (LOG), epsilon to the power of a complex number (EXP). Multiplication and division of complex numbers is assigned the next lower priority level with the least priority being assigned to addition and subtraction. All inputs containing operations of equal priority are evaluated from left to right. Implied multiplication is not permitted nor is the use of '-J' to indicate that the imaginary argument is negative, therefore, 2 - J7 is not a permitted input and will result in an error message being outputted. The correct input for that quantity should be 2 + J - 7. The sign of the number must be immediately adjacent to the number itself. In running the program, the user has the option of obtaining instructions by typing 'YES' or 'Y' or refusing them by typing 'NO' or 'N'. The user is next asked to, "Input your expression followed by carriage return:" Sample inputs are attached which indicate how to input various functions.

Continued on following page.

SPECIAL CONSIDERATIONS:

See following page.

ACKNOWLEDGEMENTS:

Professor George C. McKay, Jr.
University of Houston
Electrical-Electronics Technology

INSTRUCTIONS continued

Functions may call additional functions - that is they may be nested. The answer is typed as noted previously in both polar and rectangular form. The notes in script on the attached computer print-outs explain in greater detail the limitations on entering data. Among the syntax errors checked are illegal functions or characters, unequal number of left and right parentheses, illegal format of the complex number such as the use of -J instead of +J, implied multiplication, and use of an arithmetic operator between complex numbers without the second complex number being enclosed in parentheses (this was found to be necessary in order to separate addition or subtraction from a unary operator function).

SPECIAL CONSIDERATIONS

This calculator will perform all the functions of the following programs currently in the contributed library: FNCTS (A 303), SQR (Z) (A 303), CXARTH (A 303), CXEXP (A 303). The writer utilized the work of generating the necessary equations for the functions implemented in the calculator from these programs. In addition, a modified version of the program ALFTOV is used in the calculator as a subroutine. The user of the calculator should be aware of the round off errors which will occur in functions as complex as those implemented. All functions have been checked with a number of sets of data using complex functions in FORTRAN and the results have compared favorably to a minimum of four digits accuracy.

RUN

```
RUN
UHCX

U OF H TECH COMPLEX CALCULATOR; INSTRUCTIONS?YES
FUNCTIONS IMPLEMENTED ARE SIN,COS,TAN,HSIN (HYPERBOLIC SINE)
HCOS, & HTAN (ARGUMENTS IN RADIANS);SQR(Z), '↑' (POWER/ROOT)
LOG (BASE E), EXP (EPSILON TO THE POWER OF R+JX), AND THE
ARITHMETIC OPERATIONS ARE *, /, +, -. SAMPLE INPUTS:
'SIN 2+J-5' OR SIN(2+J-5)..'(2+J5)↑(-3+J-7)'..SIGN MUST
BE NEXT TO THE NUMBER.. 2-J7 IS NOT PERMITTED...
TO CONVERT FROM RECTANGULAR-TO-POLAR FORM TYPE 'CONP'
FOLLOWED BY  ''R' +J 'X'' WHERE 'R' IS THE 'REAL' & 'X' IS
THE IMAGINARY ARGUMENT. FOR EXAMPLE: 'CONP 3+J4' OR '3,4'
WILL RESULT IN Z=5 AT AN ANGLE OF .927295 RADIANS (53.1301
DEGREES). TO CONVERT FROM POLAR-TO-RECTANGULAR FORM:
TYPE 'CONR <MAGNITUDE>, <ANGLE>' I.E. CONR 5, .927295
IF THE ANGLE IS IN RADIANS - OR COND 5, 53.1301 IF THE
ANGLE IS IN DEGREES..
INPUT YOUR EXPRESSION FOLLOWED BY A CARRIAGE RETURN:
SQR 625+J0

ANSWER =  25    +J  0
Z =  25
ANGLE =  0     RADIANS ( 0     DEGREES)
NEXT:
EXP 0 +J 3.14159

ANSWER = -1.          +J  2.24704E-06
Z =  1.
ANGLE = -2.24704E-06    RADIANS (-1.28746E-04    DEGREES)
NEXT:
((((2+J7E1)↑(2+J0)/(1.2345E-2+J1.786))*(2+J3))+(3+J4))

ANSWER = -7948.18     +J  5902.01
Z =  9899.87
ANGLE = -.638723     RADIANS (-36.5962     DEGREES)
NEXT:
SQR (-2+J-67)

ANSWER =  5.70219     +J -5.87494
Z =  8.18718
ANGLE = -.800319     RADIANS (-45.8549     DEGREES)
NEXT:
```

```
HTAN SIN LOG 2.12345E1 +J -.--3.1298E-1

ANSWER =  8.55689E-02   +J -.014578
Z =  8.68019E-02
ANGLE = -.168745     RADIANS (-9.66841    DEGREES)
NEXT:
EXP LOS 2.12345 +J -8.98765
ILLEGAL CHARACTER OR OPERATION
NEXT:
EXP LOG 2.12345 +J -8.98765

ANSWER =  2.12345     +J -8.98765
Z =  9.23509
ANGLE = -1.33879     RADIANS (-76.7069    DEGREES)
NEXT:
COS 3+J4(4+J6)
IMPLIED '*'
NEXT:
COS 3+J4 *(4+J6)

ANSWER =  38.5219     +J  7.71731
Z =  39.2874
ANGLE =  .197731     RADIANS ( 11.3292    DEGREES)
NEXT:
2+J5 SIN 3+J4
IMPLIED '*'
NEXT:
3+J4

ANSWER =  3     +J  4
Z =  5
ANGLE =  .927295     RADIANS ( 53.1301    DEGREES)
NEXT:
CONR 5, .927295

ANSWER =  3.          +J  4
Z =  5.
ANGLE =  .927295     RADIANS ( 53.1301    DEGREES)
NEXT:
SIN COND 5, 53.1301

ANSWER =  3.85363     +J -27.0168
Z =  27.2902
ANGLE = -1.42911     RADIANS (-81.8822    DEGREES)
NEXT:
SIN 3+J4

ANSWER =  3.85374     +J -27.0168
Z =  27.2903
ANGLE = -1.42911     RADIANS (-81.882     DEGREES)
NEXT:
2-J3
ILLEGAL CHARACTER OR OPERATION
NEXT:
2+J3

ANSWER =  2     +J  3
Z =  8.24621
ANGLE =  1.32582     RADIANS ( 75.9638    DEGREES)
NEXT:
4+J4 ↑5+J0
ILLEGAL CHARACTER OR OPERATION
NEXT:
4+J4↑(5+J0)

ANSWER = -4096.       +J -4096.
Z =  5792.62
ANGLE =  .785398     RADIANS ( 45.        DEGREES)
NEXT:
((2+J5)+(3+J8)
UNEQUAL # OF '(' & ')'
NEXT:

DONE
```

TITLE: ANALYSIS OF VARIANCE FOR A RANDOMIZED ONE-WAY DESIGN

DESCRIPTION: This program computes an analysis of variance table for a completely randomized one-way design.

INSTRUCTIONS: Enter data beginning in line 9900 in the following manner: M, the number of different treatments; then the N's, where N_j is the number of observations in the jth treatment; and lastly the observations themselves by first entering the observations of treatment 1, then the observations of treatment 2, and so on. For example:

9900 DATA M

9901 DATA $N_1, N_2, \ldots N_m$

9902 DATA $P(1), P(2), \ldots P(N_1)$

9903 DATA $Q(1), Q(2), \ldots Q(n_1)$

9910 DATA $Z(1), Z(2), \ldots Z(N_m)$

where:

M = the number of different treatments $\leq$ 20

N_k = the number of observations in the kth treatment $\leq$ 50

P_k = the value of the kth observation of treatment one

Q_k = the value of the kth observation of treatment two

Z_k = the value of the kth observation of the mth treatment

SPECIAL CONSIDERATIONS: The maximum number of different treatments is 20 and the maximum number of observations per treatment is 50. These restrictions can be changed by altering the DIM statement.

C,E,F,M,R,U,V,W

N,S,T,X are array names

I,J are used for internal looping

ACKNOWLEDGEMENTS:

Jerry L. Mulcahy
Raychem Corporation

```
RUN
9900  DATA 5
9901  DATA 2,6,11,4,2
9902  DATA 83,85
9903  DATA 84,85,86,86,87,86
9904  DATA 87,87,87,88,88,88,88,88,85,88,90
9905  DATA 89,90,90,91
9906  DATA 90,92
9999  END

RUN
ANVAR1

ANALYSIS OF VARIANCE TABLE

GRAND TOTAL= 2188     NO. OBS.= 25    MEAN= 87.52


SOURCE              SS              DF              MS

TREATMENTS         94.375           4              23.5937
ERROR              25.875           20             1.29375

TOTAL              120.25           24


F = 18.2367     ON 4    AND 20    DEGREES OF FREEDOM.
PROBABILITY OF F>= 18.2367     WITH 4     AND 20    D.F. IS 0

DONE
```

CONTRIBUTED PROGRAM **BASIC**

COFTAB
36888-18020

TITLE: CONVERSATIONAL FREQUENCY AND CROSS TABULATOR

DESCRIPTION:

COFTAB (COnversational Frequency and cross TABulator) is a Time-Shared BASIC program which, under the direction of commands entered from a remote teletype terminal, recodes data and outputs frequency counts or n-dimensional cross tabulations using data that the user has stored on files in the time-shared system.

INSTRUCTIONS:

1.0 GENERAL OVERVIEW

COFTAB is a multi-segment program and only those segments which the user needs reside in the user memory space at any time. The segments are:

1. COFTAB - the main routine which initializes the system and prints the current date, time and program version; calls COFTA1.
2. COFTA1 - the heart of the system, it accepts commands from the keyboard or command file, checks the commands for syntax and sets parameters for the various command routines; transfers control to one of COFTA2, COFTA3, COFTA5, COFTA8, or COFTA9.
3. COFTA2 - the utility processor (i.e., handles listing, punching, editing, etc. of the different files); transfers control back to COFTA1.
4. COFTA3 - does the frequency counts on the data; transfers control to COFTA4.

INSTRUCTIONS: continued on following page.

SPECIAL CONSIDERATIONS:

This program was written for an HP 2000B. If you are using an HP 2000 Series System, create the files limiting the word/record size to 64 words/record; e.g., CRE-VARBLE,128,64.

Do not abort (with the BREAK key) any EDIT or RECODE command. These commands alter the variable, command, and data files and an abort may cause file destruction.

It is recommended that users not attempt to collapse multiple column variables into single column variables since user errors here may quickly destroy the data. For example; if one were recoding the two column variable AGE into the values 1, 2, and 3, it would be best to recode into the two column values 01, 02, and 03.

Several user errors are not detected by the program logic and cause system program halts. In these cases, restart processing by getting and running the main program.

When several users are sharing one account for COFTAB usage or one user has several sets of data, paper tapes of the variable and command files can be punched out on paper tape using the PUNCH command. These may be reloaded into the files by mounting them in the paper tape reader and pushing the reader start switch.

ACKNOWLEDGEMENTS:

J. G. Allan,
University of Lethbridge

INSTRUCTIONS: continued

5. COFTA4 - prints out the tables generated by COFTA3; transfers control to COFTA1.

6. COFTA5 - does the cross tabulations on the data; transfers control to COFTA6 to print out the tables.

7. COFTA6 - prints out the tables generated by COFTA5; transfers control to COFTA7.

8. COFTA7 - calculates the statistics; transfers control back to COFTA1.

9. COFTA8 - recodes the data; transfers control back to COFTA1.

10. COFTA9 - handles the initial input of data; transfers control back to COFTA1.

The program has two modes of operation:

1. keyboard mode
2. programmable mode

In the keyboard mode, commands are singly entered on the teletype and are executed immediately following the input of a carriage return.

In the programmable mode, the commands are stored in a command file and executed following the input of a RUN command. The same commands are used in either mode.

The user must familiarize himself with the following files:

1. variable file - containing the definitions of the variables (i.e., labels and column numbers).

2. command file - containing commands which are to be used as input to the program when a RUN command is entered on the keyboard.

3. data file - containing card images of the data to be tabulated.

2.0 Files Used In The Program (The casual user need not concern himself with this section.)

A total of twelve files may be used in the program, three of which are special files, the remaining nine are available for data. If the user is familiar with the TSB system, he may wish to change the names of the files, remembering of course that the first three are special purpose files. The file declarations appear in the programs as follows:

1. COFTA1 - statement 2000 - 3 special files.

2. COFTA2 - statement 2000 - 3 special files.
   statement 3000 - data files.

3. COFTA3 - statement 2000 - 3 special files.
   statement 3000 - data files.

4. COFTA4 - statement 2000 - 3 special files.

5. COFTA5 - statement 2000 - 3 special files.
   statement 3000 - data files.

6. COFTA6 - statement 2000 - 3 special files.

7. COFTA8 - statement 2000 - 3 special files.
   statement 3000 - data files.

8. COFTA9 - statement 2000 - 3 special files.
   statement 3000 - data files.

In addition to changing the FILES statement, the user must do the following:

1. The second and third of the three special files must be <u>at least</u> as large (in sectors) as
   MAX(INT((length of file#1)/2),MAX(length of data files))

2. The user must also change statement numbers 5050,5030 of COFTA1 and COFTA2 respectively to
   F7 = length of special file #1

The standard file names and their usage are:

1. VARBLE - divided into two parts, the first half is used for the variable definitions, and the last half is used for the commands, which the user has saved, which can be executed by the RUN command.

INSTRUCTIONS: continued

60 responses for any variable. The operator can still obtain frequency counts of continuous variables such as age by doing the following:

```
GET-COFTA3
1001 COM F(1,300)*
PUR-COFTA3
SAVE
GET-COFTA4
1001 COM F(1,300)
PUR-COFTA4
SAVE
```

*300 is the maximum value allowed in this change (i.e., only a maximum of 300 different responses are allowed per variable). This could also have been F(2,150) allowing a maximum of 150 different responses for 2 variables, etc.

The user can now get COFTAB and run it again and do frequency counts on his continuous variables.

Examples
Shown in sample RUN.

5.7 XTAB command

Syntax
XTAB label_1[(constraints)], label_2[(constraints)],...

where:
label_i - labels of variables to be cross tabulated.

[(constraints)] - same as those for COUNT.

Description
This command specifies the variables to be used in the cross tabulations. The number of labels in the command specify the dimension of the table, thus allowing n-dimensional tables.

Examples
Shown in sample RUN.

5.8 RECODE command

Syntax
RECODE label_1 (new value_{11}=old value range_{11}, old value range_{12},...; new value_{12}=old value range $_{21}$,...), label_2 (new value_{21}=old value range_{11},...; new value_{22}=old value range_{21},...)...

where:
label_i - label of variable to be recoded.

new value_{ij} - new value to be assigned to the variable specified by the label_i field for each case when a response falls into a value range specified by the old value range_{ij}.
NOTE: Leading and trailing blanks must be enclosed by apostrophes.

old value range_{jk} - criteria for recoding the data. If a response of label_i falls into this range, the new response for the case is assigned the value of new value_{ij}. These ranges have the same form as the constraints in 5.6 and 5.7 syntax.

Description
This command allows the user to recode data for specified variables. It is useful when ranges such as age and income level are to be grouped into certain categories such as high, low, medium or under twenty, over twenty.

CAUTION: This is an irreversible process (i.e., once data is recoded, the original data cannot be returned unless it is input back into the system as described in Data Preparation, 4.0). When recoding multi-column variables into single column variables the card image is accordingly "shrunk" that many columns and the column fields of the variable file must be readjusted accordingly by use of the EDIT command (see 5.5).

Examples
REC AGE (1=-20;2=21-50;3=51-)
The two column variable AGE is recoded into a one column variable by the following criteria:

INSTRUCTIONS: continued

1 if an observation is less than 20,
2 if an observation is between 21 and 50,
3 if an observation is greater than 50.

REC VAR7(1=1,2,5;2=3,7-9)
The one column variable VAR7 is recoded into a one column variable by the following criteria:
1 if a response is a 1 or 2 or 5, and
2 if a response is a 3 or between 7 and 9.

REC VAR7(1=1,2,5;2=3,7-9),AGE(1=-20;2=21-50;3=51-)
This is equivalent to entering each of the first two examples.

5.9 OBSERVATIONS command

Syntax
OBSERVATIONS

Description
This command allows the user to input data into the data files. The data follows immediately after after the command has been entered by one of the methods described in Data Preparation, 4.0.

5.10 APPEND command

Syntax
APPEND

Description
This command allows the user to add more data to an existing data file. The data follows immediately after the command has been entered by one of the methods described in Data Preparation, 4.0.

5.11 RUN command

Syntax
RUN [st. range$_1$, st. range$_2$,...]

where:
[st. range$_1$, st. range$_2$,...] - optional field specifying the order, and which statements of the command file are to be executed. If this field is omitted, the entire command file is executed sequentially.

Description
This command initiates the programmable mode of COFTAB and allows the user to execute some or all of the commands in the command file.

Examples
RUN
Every command stored in the command file is executed sequentially.

RUN 10, 20
Only statements 10 and 20 of the command file are executed.

RUN 10,20-30,70-
Statements 10, 20 through 30 and from 70 to the end of the command file are executed.

5.12 STOP command

Syntax
STOP

Description
This command stops execution of the COFTAB program.

6.0 AN EXAMPLE CASE STUDY

The sample RUN following illustrates this case study.

Description
This study deals with a researcher who circulates fifty questionnaires to a sample of the community. Each questionnaire has ten questions to be answered:

1. Sex of the respondent
VALUES: Blank - missing
1 - male
2 - female

INSTRUCTIONS: continued

2. Age
   VALUES: Blank - missing
   01-99 - actual age of respondent

3. Marital status
   VALUES: Blank - missing
   1 - single
   2 - married
   3 - other

4. Education
   VALUES: Blank - missing
   1 - university
   2 - no university

5-10. Questions
   VALUES: Blank - missing
   1 - yes
   2 - no
   3 - maybe

The data files F1 to F10 and the work files VARBLE,WORK1, and WORK2 must exist in the user account. Unless COFTAB is changed as indicated in section 2.0, VARBLE, WORK1, and WORK2 must be 128 sectors long.

The CLEAR command removes any information which may be remaining in the files from the previous user.

With the small amount of data shown in the sample RUN most of the data files will be unused. Create files F1 to F10 so that the total number of sectors opened will accommodate the data stored as sequential strings. All unused files must be opened to at least 1 sector.

DEFINING THE VARIABLES FOR PROCESSING

The following names are assigned:

Variable (Question #)	Label	Column Field
1.	SEX	1
2.	AGE	2-3
3.	MSTAT	4
4.	ED	5
5.	VAR5	6
6.	VAR6	7
7.	VAR7	8
8.	VAR8	9
9.	VAR9	10
10.	VAR10	11

These variables are defined by using the VARIABLE command.

CROSS TABULATIONS

The following cross tabulations are used in the sample RUN:

1. Sex versus question 6 (including missing values).

2. Sex versus question 6 (excluding missing values).

3. Sex versus marital status (exluding missing values).

4. Males versus question 7 versus question 8 (excluding missing values).

5. Males versus education versus question 7 versus question 9 (including missing values for question 7 and question 9).

This is done by entering the commands into the command file and executing them with a RUN command.

```
CRE-F1,20
CRE-F2,20
CRE-F32-,20
CRE-F4,20
CRE-F5,20
CRE-F6,20
CRE-F7,20
CRE-F8,20
CRE-F9,20
CRE-F10,20
CRE-VARBLE,128
CRE-WORK1,128
CRE-WORK2,128

GET-COFTAB
RUN
COFTAB
```

```
***** COFTAB *****
VERSION - 06/06/73

15:59   WEDNESDAY   DECEMBER 19   1973


**?CLE VAR
**?OBS
13921231232
276211221 3
1893213232
19131112232
127111322 2
112 1121313
12912 3 132
10721313221
1313 333331
13931222222
 5211323213
2333 312322
19112122113
22831233233
2833122 113
137323113 1
237 2231211
11631231231
28812212331
1551 211233
2042 123321
 502213311
22511332 1
 971 1132 1
26022313133
111122321 1
21412233322
 3432332331
149321  32
186122223 3
20912222313
251 2223333
12712311131
11121211 12
13312312 21
27722313211
1902133 212
16722313131
22111111  1
2781 231111
12511233131
```

INSTRUCTIONS: continued

60 responses for any variable. The operator can still obtain frequency counts of continuous variables such as age by doing the following:

```
GET-COFTA3
1001 COM F(1,300)*
KILL-COFTA3
SAVE
GET-COFTA4
1001 COM F(1,300)
KILL-COFTA4
SAVE
```

*300 is the maximum value allowed in this change (i.e., only a maximum of 300 different responses are allowed per variable). This could also have been F(2,150) allowing a maximum of 150 different responses for 2 variables, etc.

The user can now get COFTAB and run it again and do frequency counts on his continuous variables.

Examples
Shown in sample RUN.

5.7 XTAB command

Syntax
XTAB $label_1$[(constraints)], $label_2$[(constraints)],...

where:
$label_i$ - labels of variables to be cross tabulated.

[(constraints)] - same as those for COUNT.

Description
This command specifies the variables to be used in the cross tabulations. The number of labels in the command specify the dimension of the table, thus allowing n-dimensional tables.

Examples
Shown in sample RUN.

5.8 RECODE command

Syntax
RECODE $label_1$ (new $value_{11}$=old value $range_{11}$, old value $range_{12}$,...; new $value_{12}$=old value range $_{21}$,...), $label_2$ (new $value_{21}$=old value $range_{11}$,...; new $value_{22}$=old value $range_{21}$,...)...

where:
$label_i$ - label of variable to be recoded.

new $value_{ij}$ - new value to be assigned to the variable specified by the $label_i$ field for each case when a response falls into a value range specified by the old value $range_{ij}$.
NOTE: Leading and trailing blanks must be enclosed by apostrophes.

old value $range_{jk}$ - criteria for recoding the data. If a response of $label_i$ falls into this range, the new response for the case is assigned the value of new $value_{ij}$. These ranges have the same form as the constraints in 5.6 and 5.7 syntax.

Description
This command allows the user to recode data for specified variables. It is useful when ranges such as age and income level are to be grouped into certain categories such as high, low, medium or under twenty, over twenty.

CAUTION: This is an irreversible process (i.e., once data is recoded, the original data cannot be returned unless it is input back into the system as described in Data Preparation, 4.0). When recoding multi-column variables into single column variables the card image is accordingly "shrunk" that many columns and the column fields of the variable file must be readjusted accordingly by use of the EDIT command (see 5.5).

Examples
REC AGE (1=-20;2=21-50;3=51-)
The two column variable AGE is recoded into a one column variable by the following criteria:

INSTRUCTIONS: continued

1 if an observation is less than 20,
2 if an observation is between 21 and 50,
3 if an observation is greater than 50.

REC VAR7(1=1,2,5;2=3,7-9)
The one column variable VAR7 is recoded into a one column variable by the following criteria:
1 if a response is a 1 or 2 or 5, and
2 if a response is a 3 or between 7 and 9.

REC VAR7(1=1,2,5;2=3,7-9),AGE(1=-20;2=21-50;3=51-)
This is equivalent to entering each of the first two examples.

5.9 OBSERVATIONS command

Syntax
OBSERVATIONS

Description
This command allows the user to input data into the data files. The data follows immediately after after the command has been entered by one of the methods described in Data Preparation, 4.0.

5.10 APPEND command

Syntax
APPEND

Description
This command allows the user to add more data to an existing data file. The data follows immediately after the command has been entered by one of the methods described in Data Preparation, 4.0.

5.11 RUN command

Syntax
RUN [st. range$_1$, st. range$_2$,...]

where:
[st. range$_1$, st. range$_2$,...] - optional field specifying the order, and which statements of the command file are to be executed. If this field is omitted, the entire command file is executed sequentially.

Description
This command initiates the programmable mode of COFTAB and allows the user to execute some or all of the commands in the command file.

Examples
RUN
Every command stored in the command file is executed sequentially.

RUN 10, 20
Only statements 10 and 20 of the command file are executed.

RUN 10,20-30,70-
Statements 10, 20 through 30 and from 70 to the end of the command file are executed.

5.12 STOP command

Syntax
STOP

Description
This command stops execution of the COFTAB program.

6.0 AN EXAMPLE CASE STUDY

The sample RUN following illustrates this case study.

Description
This study deals with a researcher who circulates fifty questionnaires to a sample of the community. Each questionnaire has ten questions to be answered:

1. Sex of the respondent
VALUES: Blank - missing
1 - male
2 - female

INSTRUCTIONS: continued

2. Age
   VALUES: Blank - missing
   01-99 - actual age of respondent

3. Marital status
   VALUES: Blank - missing
   1 - single
   2 - married
   3 - other

4. Education
   VALUES: Blank - missing
   1 - university
   2 - no university

5-10. Questions
   VALUES: Blank - missing
   1 - yes
   2 - no
   3 - maybe

The data files F1 to F10 and the work files VARBLE,WORK1, and WORK2 must exist in the user account. Unless COFTAB is changed as indicated in section 2.0, VARBLE, WORK1, and WORK2 must be 128 sectors long.

The CLEAR command removes any information which may be remaining in the files from the previous user.

With the small amount of data shown in the sample RUN most of the data files will be unused. Open files F1 to F10 so that the total number of sectors opened will accommodate the data stored as sequential strings. All unused files must be opened to at least 1 sector.

DEFINING THE VARIABLES FOR PROCESSING

The following names are assigned:

Variable (Question #)	Label	Column Field
1.	SEX	1
2.	AGE	2-3
3.	MSTAT	4
4.	ED	5
5.	VAR5	6
6.	VAR6	7
7.	VAR7	8
8.	VAR8	9
9.	VAR9	10
10.	VAR10	11

These variables are defined by using the VARIABLE command.

CROSS TABULATIONS

The following cross tabulations are used in the sample RUN:

1. Sex versus question 6 (including missing values).

2. Sex versus question 6 (excluding missing values).

3. Sex versus marital status (exluding missing values).

4. Males versus question 7 versus question 8 (excluding missing values).

5. Males versus education versus question 7 versus question 9 (including missing values for question 7 and question 9).

This is done by entering the commands into the command file and executing them with a RUN command.

```
OPE-F1,20
OPE-F2,20
OPE-F32←,20
OPE-F4,20
OPE-F5,20
OPE-F6,20
OPE-F7,20
OPE-F8,20
OPE-F9,20
OPE-F10,20
OPE-VARBLE,128
OPE-WORK1,128
OPE-WORK2,128

GET-$COFTAB
RUN
COFTAB



***** COFTAB *****
VERSION - 06/06/73

15:59   WEDNESDAY   DECEMBER 19   1973



**?CLE VAR
**?OBS
13921231232
276211221 3
1893213232
19131112232
127111322 2
112 1121313
12912 3 132
10721313221
1313 333331
13931222222
 5211323213
2333 312322
19112122113
22831233233
2833122 113
137323113 1
237 2231211
11631231231
28812212331
1551 211233
2042 123321
 502213311
22511332 1
 971 1132 1
26022313133
111122321 1
21412233322
 3432332331
149321  32
186122223 3
20912222313
251 2223333
12712311131
11121211 12
13312312 21
27722313211
1902133 212
16722313131
22111111  1
2781 231111
12511233131
```

```
296 2112231
16731211223
17022121 31
16012223121
2681 312122
1161232223
10911 32211
20111123332
152113 233
EOT

 50    DATA RECORDS STORED


**?
   VARIABLE 10 SEX,1,1,BLANK=MISSING 1=MALE 2=FEMALE
**?VAR 20 AGE,2,3
**?VAR 30 MSTAT,4,4,BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER
**?VAR 40 EED,5,5,BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY
**?VAR 50 VAR5,6,6
**?VAR 60 VAR6,7,7
**?VAR 70 VAR7,8,8
**?VAR 80 VAR8,9,9
**?VAR 90 VAR9,10,10
**?VAR 100 VAR10,11,11
**?LIS VAR
 10    SEX,1,1,BLANK=MISSING 1=MALE 2=FEMALE
 20    AGE,2,3
 30    MSTAT,4,4,BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER
 40    EED,5,5,BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY
 50    VAR5,6,6
 60    VAR6,7,7
 70    VAR7,8,8
 80    VAR8,9,9
 90    VAR9,10,10
 100   VAR10,11,11
**?EDI VAR 40;EED;ED
**?LIS VAR
 10    SEX,1,1,BLANK=MISSING 1=MALE 2=FEMALE
 20    AGE,2,3
 30    MSTAT,4,4,BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER
 40    ED,5,5,BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY
 50    VAR5,6,6
 60    VAR6,7,7
 70    VAR7,8,8
 80    VAR8,9,9
 90    VAR9,10,10
 100   VAR10,11,11
**?EDI VAR 50-;;,BLANK=MISSING 1=YES 2=NO 3=MAYBE
**?LIS VAR
 10    SEX,1,1,BLANK=MISSING 1=MALE 2=FEMALE
 20    AGE,2,3
 30    MSTAT,4,4,BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER
 40    ED,5,5,BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY
 50    VAR5,6,6,BLANK=MISSING 1=YES 2=NO 3=MAYBE
 60    VAR6,7,7,BLANK=MISSING 1=YES 2=NO 3=MAYBE
 70    VAR7,8,8,BLANK=MISSING 1=YES 2=NO 3=MAYBE
 80    VAR8,9,9,BLANK=MISSING 1=YES 2=NO 3=MAYBE
 90    VAR9,10,10,BLANK=MISSING 1=YES 2=NO 3=MAYBE
 100   VAR10,11,11,BLANK=MISSING 1=YES 2=NO 3=MAYBE
**?COUNT SEX,MSTAT,ED,VAR5,VAR6



VARIABLE:  SEX            BLANK=MISSING 1=MALE 2=FEMALE


 SYMBOL      FREQUENCY     PER CENTAGE

      1             28          56.00
      2             18          36.00
  BLANK              4           8.00

  TOTAL             50         100.00
```

```
VARIABLE:  MSTAT         BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER

 SYMBOL     FREQUENCY    PER CENTAGE

      1            23          46.00
      3            12          24.00
      2            11          22.00
  BLANK             4           8.00

  TOTAL            50         100.00


VARIABLE:  ED            BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY

 SYMBOL     FREQUENCY    PER CENTAGE

      2            23          46.00
      1            20          40.00
  BLANK             7          14.00

  TOTAL            50         100.00


VARIABLE:  VAR5          BLANK=MISSING 1=YES 2=NO 3=MAYBE

 SYMBOL     FREQUENCY    PER CENTAGE

      2            18          36.00
      3            16          32.00
      1            14          28.00
  BLANK             2           4.00

  TOTAL            50         100.00


VARIABLE:  VAR6          BLANK=MISSING 1=YES 2=NO 3=MAYBE

 SYMBOL     FREQUENCY    PER CENTAGE

      3            17          34.00
      1            17          34.00
      2            14          28.00
  BLANK             2           4.00

  TOTAL            50         100.00


**?COU VAR7,VAR8,VAR9,VAR10,VAR11
*****VAR11 IS NOT DEFINED AS A VARIABLE
**?COU VAR7,VAR8,VAR9,VAR10


VARIABLE:  VAR7          BLANK=MISSING 1=YES 2=NO 3=MAYBE

 SYMBOL     FREQUENCY    PER CENTAGE

      2            19          38.00
      3            15          30.00
      1            12          24.00
  BLANK             4           8.00

  TOTAL            50         100.00
```

VARIABLE: VAR8 BLANK=MISSING 1=YES 2=NO 3=MAYBE

SYMBOL	FREQUENCY	PER CENTAGE
2	17	34.00
3	15	30.00
1	13	26.00
BLANK	5	10.00
TOTAL	50	100.00

VARIABLE: VAR9 BLANK=MISSING 1=YES 2=NO 3=MAYBE

SYMBOL	FREQUENCY	PER CENTAGE
3	19	38.00
1	13	26.00
2	11	22.00
BLANK	7	14.00
TOTAL	50	100.00

VARIABLE: VAR10 BLANK=MISSING 1=YES 2=NO 3=MAYBE

SYMBOL	FREQUENCY	PER CENTAGE
1	21	42.00
3	12	24.00
2	11	22.00
BLANK	6	12.00
TOTAL	50	100.00

**?COU AGE

VARIABLE: AGE

SYMBOL	FREQUENCY	PER CENTAGE
39	2	4.00
91	2	4.00
27	2	4.00
52	2	4.00
33	2	4.00
37	2	4.00
16	2	4.00
25	2	4.00
60	2	4.00
11	2	4.00
09	2	4.00
67	2	4.00
83	1	2.00
12	1	2.00
29	1	2.00
88	1	2.00
55	1	2.00
04	1	2.00
50	1	2.00
07	1	2.00
97	1	2.00
31	1	2.00
76	1	2.00
14	1	2.00
34	1	2.00
49	1	2.00
86	1	2.00
89	1	2.00
51	1	2.00
77	1	2.00
90	1	2.00
28	1	2.00
21	1	2.00
78	1	2.00
96	1	2.00
70	1	2.00
68	1	2.00
01	1	2.00
TOTAL	50	100.00

```
**?RECODE AGE(01=' 0'-30;02=31-50;03=51-99)
**?LIS VAR 20
 20   AGE,2,3
**?EDI LIS VAR 20;;,1=YOUNGER THAN 30 2=31 TO 50 3=OVER 50
 20   AGE,2,3,1=YOUNGER THAN 30 2=31 TO 50 3=OVER 50
**?COUNT AGE
```

VARIABLE: AGE 1=YOUNGER THAN 30 2=31 TO 50 3=OVER 50

SYMBOL	FREQUENCY	PER CENTAGE
03	22	44.00
01	18	36.00
02	10	20.00
TOTAL	50	100.00

```
**?COM 10 XTAB SEX,VAR6
**?COM 20 XTAB SEX(1,2),VAR6(1,2,3)
**?COM 30 XTAB SEX)1,2),MSTAT(1,2,3),ED(1,2)
**?COM 40 XTAB SEX(1),VAR7(1,2,3),VAR8(1,2,3)
**?COM 50 XTAB VAR6,SEX
**?LIS COM
```

```
 10    XTAB SEX,VAR6
 20    XTAB SEX(1,2),VAR6(1,2,3)
 30    XTAB SEX)1,2),MSTAT(1,2,3),ED(1,2)
 40    XTAB SEX(1),VAR7(1,2,3),VAR8(1,2,3)
 50    XTAB VAR6,SEX
**?COM 30 XTAB SEX(1,2),MSTAT(1,2,3),ED(1,2)
**?COM 50 XTAB SEX(1),ED(1,2),VAR7,VAR9
**?COM 15 XTAB VAR6,SEX
**?LIS COM
 10    XTAB SEX,VAR6
 15    XTAB VAR6,SEX
 20    XTAB SEX(1,2),VAR6(1,2,3)
 30    XTAB SEX(1,2),MSTAT(1,2,3),ED(1,2)
 40    XTAB SEX(1),VAR7(1,2,3),VAR8(1,2,3)
 50    XTAB SEX(1),ED(1,2),VAR7,VAR9
**?RUN
!!? 10    XTAB SEX,VAR6



SEX:  ROWS                   BLANK=MISSING 1=MALE 2=FEMALE

VAR6:  COLUMNS               BLANK=MISSING 1=YES 2=NO 3=MAYBE


 CODE BLANK      1      2      3 TOTAL

BLANK      0      1      1      2      4
         0.0   25.0   25.0   50.0  100.0
         0.0    2.0    2.0    4.0    8.0

    1      2      9      7     10     28
         7.1   32.1   25.0   35.7  100.0
         4.0   18.0   14.0   20.0   56.0

    2      0      7      6      5     18
         0.0   38.9   33.3   27.8  100.0
         0.0   14.0   12.0   10.0   36.0

TOTAL      2     17     14     17     50
         4.0   34.0   28.0   34.0  100.0
         4.0   34.0   28.0   34.0  100.0


DEGREES OF FREEDOM:   6
CHI-SQUARE:   2.638
EXACT PROBABILITY OF CHI-SQUARE:  0.853
CONTINGENCY COEFFICIENT:  0.224
CORRECTED CONTINGENCY COEFFICIENT:  0.265
CRAMER'S V:  0.162
GOODMAN-KRUSKAL'S TAU-C:  0.012


!!? 15    XTAB VAR6,SEX



VAR6:  ROWS                  BLANK=MISSING 1=YES 2=NO 3=MAYBE

SEX:  COLUMNS                BLANK=MISSING 1=MALE 2=FEMALE
```

```
 CODE BLANK      1      2 TOTAL

BLANK      0      2      0      2
         0.0  100.0    0.0  100.0
         0.0    4.0    0.0    4.0

    1      1      9      7     17
         5.9   52.9   41.2  100.0
         2.0   18.0   14.0   34.0

    2      1      7      6     14
         7.1   50.0   42.9  100.0
         2.0   14.0   12.0   28.0

    3      2     10      5     17
        11.8   58.8   29.4  100.0
         4.0   20.0   10.0   34.0

TOTAL      4     28     18     50
         8.0   56.0   36.0  100.0
         8.0   56.0   36.0  100.0


DEGREES OF FREEDOM:    6
CHI-SQUARE:    2.638
EXACT PROBABILITY OF CHI-SQUARE:  0.853
CONTINGENCY COEFFICIENT:  0.224
CORRECTED CONTINGENCY COEFFICIENT:  0.265
CRAMER'S V:  0.162
GOODMAN-KRUSKAL'S TAU-C:  0.035


!!? 20    XTAB SEX(1,2),VAR6(1,2,3)



SEX:  ROWS                   BLANK=MISSING 1=MALE 2=FEMALE

VAR6:  COLUMNS               BLANK=MISSING 1=YES 2=NO 3=MAYBE


 CODE      1      2      3 TOTAL

    1      9      7     10     26
        34.6   26.9   38.5  100.0
        20.5   15.9   22.7   59.0

    2      7      6      5     18
        38.9   33.3   27.8  100.0
        15.9   13.6   11.4   40.9

TOTAL     16     13     15     44
        36.4   29.5   34.1  100.0
        36.4   29.5   34.1  100.0


DEGREES OF FREEDOM:    2
CHI-SQUARE:    0.557
EXACT PROBABILITY OF CHI-SQUARE:  0.761
CONTINGENCY COEFFICIENT:  0.112
CORRECTED CONTINGENCY COEFFICIENT:  0.145
CRAMER'S V:  0.113
GOODMAN-KRUSKAL'S TAU-C:  0.006


!!? 30    XTAB SEX(1,2),MSTAT(1,2,3),ED(1,2)
```

```
SEX:  1                    BLANK=MISSING 1=MALE 2=FEMALE

MSTAT:  ROWS               BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER

ED:  COLUMNS               BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY


 CODE      1       2 TOTAL

    1      4       8    12
        33.3    66.7 100.0
        16.0    32.0  48.0

    2      4       2     6
        66.7    33.3 100.0
        16.0     8.0  24.0

    3      4       3     7
        57.1    42.9 100.0
        16.0    12.0  28.0

TOTAL     12      13    25
        48.0    52.0 100.0
        48.0    52.0 100.0


DEGREES OF FREEDOM:    2
CHI-SQUARE:    2.106
EXACT PROBABILITY OF CHI-SQUARE:  0.350
CONTINGENCY COEFFICIENT:  0.279
CORRECTED CONTINGENCY COEFFICIENT:  0.360
CRAMER'S V:  0.290
GOODMAN-KRUSKAL'S TAU-C:  0.084


SEX:  2                    BLANK=MISSING 1=MALE 2=FEMALE

MSTAT:  ROWS               BLANK=MISSING 1=SINGLE 2=MARRIED 3=OTHER

ED:  COLUMNS               BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY


 CODE      1       2 TOTAL

    1      3       3     6
        50.0    50.0 100.0
        27.3    27.3  54.5

    2      1       2     3
        33.3    66.7 100.0
         9.1    18.2  27.2

    3      2       0     2
       100.0     0.0 100.0
        18.2     0.0  18.1

TOTAL      6       5    11
        54.5    45.5 100.0
        54.5    45.5 100.0


DEGREES OF FREEDOM:    2
CHI-SQUARE:    2.261
EXACT PROBABILITY OF CHI-SQUARE:  0.323
CONTINGENCY COEFFICIENT:  0.413
CORRECTED CONTINGENCY COEFFICIENT:  0.533
CRAMER'S V:  0.453
GOODMAN-KRUSKAL'S TAU-C:  0.206


!!? 40    XTAB SEX(1),VAR7(1,2,3),VAR8(1,2,3)
```

SEX: 1 BLANK=MISSING 1=MALE 2=FEMALE

VAR7: ROWS BLANK=MISSING 1=YES 2=NO 3=MAYBE

VAR8: COLUMNS BLANK=MISSING 1=YES 2=NO 3=MAYBE

CODE	1	2	3	TOTAL
1	1	4	2	7
	14.3	57.1	28.6	100.0
	4.5	18.2	9.1	31.8
2	2	5	3	10
	20.0	50.0	30.0	100.0
	9.1	22.7	13.6	45.4
3	3	1	1	5
	60.0	20.0	20.0	100.0
	13.6	4.5	4.5	22.7
TOTAL	6	10	6	22
	27.3	45.5	27.3	100.0
	27.3	45.5	27.3	100.0

DEGREES OF FREEDOM: 4
CHI-SQUARE: 3.688
EXACT PROBABILITY OF CHI-SQUARE: 0.548
CONTINGENCY COEFFICIENT: 0.379
CORRECTED CONTINGENCY COEFFICIENT: 0.464
CRAMER'S V: 0.290
GOODMAN-KRUSKAL'S TAU-C: 0.083

!!? 50 XTAB SEX(1),ED(1,2),VAR7,VAR9

SEX: 1 BLANK=MISSING 1=MALE 2=FEMALE

ED: 1 BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY

VAR7: ROWS BLANK=MISSING 1=YES 2=NO 3=MAYBE

VAR9: COLUMNS BLANK=MISSING 1=YES 2=NO 3=MAYBE

CODE	BLANK	1	2	3	TOTAL
BLANK	0	1	0	0	1
	0.0	100.0	0.0	0.0	100.0
	0.0	7.7	0.0	0.0	7.6
1	0	2	1	2	5
	0.0	40.0	20.0	40.0	100.0
	0.0	15.4	7.7	15.4	38.4
2	1	1	1	2	5
	20.0	20.0	20.0	40.0	100.0
	7.7	7.7	7.7	15.4	38.4
3	0	0	1	1	2
	0.0	0.0	50.0	50.0	100.0
	0.0	0.0	7.7	7.7	15.3
TOTAL	1	4	3	5	13
	7.7	30.8	23.1	38.5	100.0
	7.7	30.8	23.1	38.5	100.0

```
DEGREES OF FREEDOM:   9
CHI-SQUARE:   5.460
EXACT PROBABILITY OF CHI-SQUARE:  0.793
CONTINGENCY COEFFICIENT:  0.544
CORRECTED CONTINGENCY COEFFICIENT:  0.628
CRAMER'S V:  0.374
GOODMAN-KRUSKAL'S TAU-C:  0.141


SEX:  1                    BLANK=MISSING 1=MALE 2=FEMALE

ED:  2                     BLANK=MISSING 1=UNIVERSITY 2=NO UNIVERSITY

VAR7:  ROWS                BLANK=MISSING 1=YES 2=NO 3=MAYBE

VAR9:  COLUMNS             BLANK=MISSING 1=YES 2=NO 3=MAYBE


 CODE BLANK     1     2     3 TOTAL

BLANK     0     0     1     1     2
        0.0   0.0  50.0  50.0 100.0
        0.0   0.0   7.7   7.7  15.3

    1     1     0     0     2     3
       33.3   0.0   0.0  66.7 100.0
        7.7   0.0   0.0  15.4  23.0

    2     2     1     2     1     6
       33.3  16.7  33.3  16.7 100.0
       15.4   7.7  15.4   7.7  46.1

    3     0     0     1     1     2
        0.0   0.0  50.0  50.0 100.0
        0.0   0.0   7.7   7.7  15.3

TOTAL     3     1     4     5    13
       23.1   7.7  30.8  38.5 100.0
       23.1   7.7  30.8  38.5 100.0


DEGREES OF FREEDOM:   9
CHI-SQUARE:   5.417
EXACT PROBABILITY OF CHI-SQUARE:  0.797
CONTINGENCY COEFFICIENT:  0.542
CORRECTED CONTINGENCY COEFFICIENT:  0.626
CRAMER'S V:  0.373
GOODMAN-KRUSKAL'S TAU-C:  0.155


**?STOP

DONE
```

IDA
36755

TITLE: INTERACTIVE DATA ANALYSIS

DESCRIPTION:

IDA is an interactive system for statistical analysis that has been developed at the Graduate School of Business of the University of Chicago for implementation on HP 2000C and C'/F mini-computers. The system is fully conversational, permitting a statistical analysis to be implemented flexibly by a series of commands that can be accomplished in almost any sequence, according to the user's choice after seeing the results of previous commands. IDA is virtually self-documenting, and has a number of convenience features for the user, including multi-level prompts, data-editing, automatic updating, and recovering from errors. IDA has been used in teaching of statistics courses at different levels with gratifying response from students. It has also proved valuable as a tool for research.

INSTRUCTIONS:

Get and RUN program. Type "YES" in response to query, "DO YOU NEED HELP?"

Complete user instructions are included in material published by the HP Computer Curriculum Project: HP 5951-5606 CONVERSATIONAL STATISTICS $13.50. For further information contact:

HP Computer Curriculum Project
11000 Wolfe Road
Cupertino, CA 95014

Get and RUN "IDAFIL" for a list of data files included in the IDA package.
Get and RUN "IDAPRO" for a list of programs included in the IDA package.

ACKNOWLEDGEMENTS:

Robert Ling/Harry Roberts
Graduate School of Business
University of Chicago

RUN

```
GET-IDA
RUN
IDA

14 JUN 74 VERSION

GOOD MORNING.  NEED HELP ?YES

IDA  (AN ACRONYM FOR 'INTERACTIVE DATA ANALYSIS')

IS A SYSTEM OF PROGRAMS CHAINED TO THE CONTROL PROGRAM $IDA.

IN USING IDA, YOU ISSUE A COMMAND FOR THE TASK YOU WANT DONE,
IDA DOES IT, YOU EXAMINE THE RESULTS, AND ON THE BASIS OF THE
RESULTS YOU DECIDE WHICH TASK YOU WISH IDA TO EXECUTE NEXT AND
ISSUE THE APPROPRIATE COMMAND.
WHEN YOU DO NOT WANT TO DO MORE, YOU TYPE THE COMMAND 'QUIT'.

CURRENTLY THERE ARE OVER 100 IDA COMMANDS.

>   IS THE SYMBOL INDICATING THAT IDA IS READY FOR YOU TO ISSUE
       A COMMAND WORD INDICATING WHICH TASK IS TO BE EXECUTED NEXT.

PROMPTS ARE GIVEN BY IDA WHEN MORE INFORMATION IS NEEDED TO EXECUTE
       THE TASK INDICATED BY THE COMMAND.
       THE PROMPTS ARE DETAILED (LEVEL 1), LESS DETAILED (LEVEL 2)
       OR BRIEF (POSSIBLY CRYPTIC) IN LEVEL 3.
       PROMPT LEVEL 1 IS IN EFFECT UNLESS YOU ISSUE THE COMMAND
       CHGP  TO CHANGE THE PROMPT LEVEL.
       IT IS SUGGESTED YOU USE PROMPT LEVEL 1 THE FIRST TIME YOU
       ISSUE A COMMAND IN ORDER TO TAKE ADVANTAGE OF THE
       ADDITIONAL EXPLANATIONS AVAILABLE AT THAT LEVEL.

*  WILL PRECEDE A PROMPT IF FURTHER INFORMATION WILL BE
   FORTHCOMING IF YOU
   (1)  TYPE  ?   OR
   (2)  JUST SIT AND WAIT A BIT.

*  WANT MORE DETAILS ? YES
YOU CAN GET MORE INFORMATION ABOUT IDA
   (1)  FROM 'CONVERSATIONAL STATISTICS' AND ITS 'COMPUTER
        PREFACE', OR
   (2)  IF YOU GET-$IDA, RUN IT, AND
        ISSUE THE IDA COMMANDS:
          EXPL  TO GET AN EXPLANATION FOR A SPECIFIC COMMAND
          INFO  TO GET EXPLANATIONS OF ALL THE COMMANDS IN A
                GROUP--SUCH AS TRANSFORMATION COMMANDS
        OR, IN SOME CASES,
        ISSUE THE COMMAND AT PROMPT LEVEL 1.

IN ORDER TO ANALYSE DATA WITH IDA,
DATA MUST FIRST BE ENTERED IN THE IDA DATA MATRIX.
YOU CAN THINK OF THE DATA MATRIX AS A TABLE WITH NUMBER OF ROWS
EQUAL TO THE NUMBER OF OBSERVATIONS (QUESTIONNAIRES) AND NUMBER
OF COLUMNS EQUAL TO THE NUMBER OF VARIABLES.

YOU MAY ENTER DATA IN THE IDA DATA MATRIX BY
    (1)  USING DATA FILE(S) AND ONE OF THE FOLLOWING COMMANDS:
         ENTER, ENTS, ENRA, CRSP, OR EOBR;

    (2)  INPUTTING DATA DIRECTLY FROM THE TERMINAL WITH TAPE OR
         KEYBOARD, USING 'ENTER';

    (3)  ENTERING DATA GENERATED BY IDA, USING 'RAND' OR 'INDX'.
```

```
AFTER DATA IS ENTERED, YOU MAY EXECUTE OTHER COMMANDS TO:
DESIGNATE VARIABLES FOR ANALYSIS OF CROSS-SECTIONAL AND TIME-SERIES
   DATA BY SIMPLE AND MULTIPLE REGRESSION AND RELATED TECHNIQUES;
TRANSFORM THE DATA AND PLACE THE RESULTS IN THE DATA MATRIX;
ADD OTHER VARIABLES TO THE DATA MATRIX;
DELETE OBSERVATIONS;
RETRIEVE DELETED OBSERVATIONS;
SORT THE DATA INTO ASCENDING ORDER;
SAVE PART OR ALL OF THE DATA MATRIX OR FITTED OR RESIDUAL
   VALUES IN ONE OF YOUR FILES;
EXAMINE THE DATA OR FITTED OR RESIDUAL VALUES BY DISPLAYING
   THEM IN PLOTS OR HISTOGRAMS;
PRINT TABLES OF DATA VALUES AND CROSS TABULATIONS OF FREQUENCIES
   AND OF MEANS;
ANALYSE THE DATA IN VARIOUS WAYS;
COMPUTE AND PRINT OUT SUMMARY AND ONE SAMPLE STATISTICS,
PERFORM OTHER TASKS BY USING THE IDA COMMAND 'NEWC' AND A PROGRAM
   WRITTEN BY YOU TO BE USED WITH IDA.

OR, YOU CAN USE IDA TO:
CREATE NEW DATA FILES BY SAVING AN EDITED VERSION OF SOME OR ALL
   COLUMNS OF THE IDA DATA MATRIX WITH 'SAVF' OR BY USING 'CRFI'
   FOR LARGER SETS OF DATA;
LIST THE CONTENTS OF FILES WITH 'FILE';
COMPUTE NORMAL PROBABILITES WITH 'GAUS';
SELECT RANDOM SAMPLES WITH 'PSAM'.

YOU CAN NORMALLY ENTER A MAXIMUM OF 100 ROWS (OR OBSERVATIONS)
AND A MAXIMUM OF 19 COLUMNS (OR VARIABLES) OF DATA IN THE IDA
DATA MATRIX, BUT YOU CAN USE THE IDA COMMAND 'RDIM'  TO RE-
DIMENSION THE DATA MATRIX TO MORE ROWS (A MAX. OF 563) AT
THE EXPENSE OF FEWER COLUMNS (A MIN. OF 1).

YOU CAN STOP THE NORMAL EXECUTION OF IDA BY
    (1)  USING C-CONTROL IF IT IS AWAITING INPUT BY YOU, OR,
         OTHERWISE,
    (2)  USING THE 'BRK', 'BREAK', OR 'INTERRUPT' KEY.

IF YOU THEN WISH TO GET BACK TO THE COMMAND LEVEL, TYPE
'RUN-9998', THEN CARRIAGE RETURN
AND IDA WILL RESPOND WITH
                                  >
                                   THE COMMAND READINESS SYMBOL.

TO STOP USING IDA, TYPE THE IDA COMMAND
                                                QUIT.
TO GET A LIST OF IDA COMMANDS, TYPE THE IDA COMMAND,
                                                          COMM
TO GET ADDITIONAL DETAILS, TYPE THE IDA COMMAND,
                                                      INFO

> QUIT

DONE
```

SCOREF
36888-18035

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

DESCRIPTION:

This 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:

Enter data starting in line 500. Data can be entered in any order except the number of scores must be first.

This is a modified version of the program "SCORES" to sort the scores entered in DATA statements.

ACKNOWLEDGEMENTS: Donald E. Gettinger (original program)
Modified by Dr. L. Winrich and E. Schroeder
University of Wisconsin - La Crosse

```
RUN
SCOREF

MEAN =   61.8182
STANDARD DEVIATION =   19.Ø127

SCORE          DEVIATION       Z-SCORE        T-SCORE
-----          ---------       -------        -------
 99               37.2          1.96           69.56
 99               37.2          1.96           69.56
 89               27.2          1.43           64.3Ø
 89               27.2          1.43           64.3Ø
 87               25.2          1.32           63.24
 85               23.2          1.22           62.19
 75               13.2          Ø.69           56.93
 74               12.2          Ø.64           56.41
 73               11.2          Ø.59           55.88
 7Ø                8.2          Ø.43           54.3Ø
 69                7.2          Ø.38           53.78
 67                5.2          Ø.27           52.73
 66                4.2          Ø.22           52.2Ø
 64                2.2          Ø.11           51.15
 64                2.2          Ø.11           51.15
 64                2.2          Ø.11           51.15
 62                Ø.2          Ø.Ø1           5Ø.1Ø
 62                Ø.2          Ø.Ø1           5Ø.1Ø
 61               -Ø.8         -Ø.Ø4           49.57
 59               -2.8         -Ø.15           48.52
 58               -3.8         -Ø.2Ø           47.99
 57               -4.8         -Ø.25           47.47
 54               -7.8         -Ø.41           45.89
 49              -12.8         -Ø.67           43.26
 48              -13.8         -Ø.73           42.73
 47              -14.8         -Ø.78           42.21
 44              -17.8         -Ø.94           4Ø.63
 44              -17.8         -Ø.94           4Ø.63
 39              -22.8         -1.2Ø           38.ØØ
 37              -24.8         -1.31           36.95
 34              -27.8         -1.46           35.37
 29              -32.8         -1.73           32.74
 22              -39.8         -2.Ø9           29.Ø6

DONE
```

STAT1

TITLE: A HISTOGRAM, STANDARD DEVIATION, AND PLOT OF A SET OF NUMBERS 36881-18003

DESCRIPTION: This program provides a selection of any or all of the following:

1. A statistical analysis giving standard deviation, normalized deviation, the mean and, as an option, the median.
2. A histogram of the set of numbers.
3. A plot of the numbers with the mean indicated.
4. Automatic or manual selection of scale factors for the histogram and the plot.

INSTRUCTIONS: Before running the program enter the following data beginning with line 1000:

1000 DATA X_1, X_2, X_3, X_4 ....

where X_i = DATA POINTS (limit 1000 points)

SPECIAL CONSIDERATIONS: If more than 1000 data points are required, redimension G(N) in line 6.

The median is sorted with a technique requiring less than a third of the time of other methods thereby saving considerable terminal time for large groups of numbers.

If numbers are outside the limit of the plot they have indicated by an "0" at the edge of the plot. Valid numbers are indicated by "*".

If more than one histogram bar contains "Maximum Frequency" each is printed separately giving the limits of their occurence.

Automatic scale selection is based on standard deviation of the number set. Positive, negative and decimal numbers are acceptable.

Provisions are made for removing a fixed offset or "TARE" from the set of data.

The end of each run is identified with the date and time.

ACKNOWLEDGEMENTS: A. E. Brown
Saratoga Systems, Inc.

RUN

```
GET-STAT1
1000  DATA 0,.5,.86603,1,.86603,.5,0,-.5,-.86603,-1,-.86603,-.5,0
RUN
STAT1

DO YOU WANT AN HISTOGRAM    1=YES   0=NO ?1
DO YOU WANT A PLOT          1=YES   0=NO ?1
DO YOU WANT THE MEDIAN      1=YES   0=NO ?1
AUTOMATIC SCALE FACTOR      1=YES   0=NO ?1
THIS IS AN ANALYSIS OF WHAT
?OFFSET SINE WAVE
WHAT ARE THE DIMENSIONS  ?UNITS
WHAT IS THE TARE READING ?.5
-


                    THIS IS A STATISTICAL ANALYSIS OF
                             OFFSET SINE WAVE
                                 (UNITS)


   NUMBER OF POINTS      =  13
   TARE OF DATA          =  .5
   MEAN OF DATA          = -.5
   MEDIAN OF DATA        = -.5
   STANDARD DEVIATION    =  .707109
   NORMALIZED DEVIATION  = -1.41422
   SAMPLE SIZE           =  .2
   MAXIMUM FREQUENCY     =  3
     OCCURS BETWEEN -.6          AND -.4


                                  % OF MAXIMUM
             0        20        40        60        80       100
             !         !         !         !         !         I
             +----.----+----.----+----.----+----.----+----.----+
-3          -I
             I
             I
             I
             I
-2          -I
             I
             I****************
             I*********************************
             I
-1          -I*********************************
             I
             I**************************************************
             I
             I
 0          -I*********************************
             I*********************************
             I****************
             I
             I
 1          -I
```

```
                            UNITS
-3                 -1                  1                   3
!         !         !         !         !         !         !
+----.----+----.----+----.----+----.----+----.----+----.----+
                        -M-
                         *
                              *
                                 *
                                   *
                                 *
                              *
                         *
                    *
                *
              *
                *
                    *
                         *
```

```
 1/14/75   18:36


DONE
```

CONTRIBUTED PROGRAM **BASIC**

UNLETH
36888-18021

TITLE: UNILETH STATISTICS PACKAGE

DESCRIPTION:

A set of related statistical analysis program modules designed for interactive use from a common data base. Package contains modules for: 1. Data Matrix Loading, 2. Factor Analysis, 3. Multiple Descriminant Analysis, 4. Two and Three Factor Analysis of Variance, and 5. Cross Tabulations. Subset of modules is selectable by user by initial commands.

Program names are: MATIN, EDITM, MDRS, CORS, SEVS, VORS, DISCRM, AEVS, DISC2, AVAR23, AVR23+, STRGIN, and COUNTS.

INSTRUCTIONS:

I. MATRIX DATA LOADING

MATIN

This program will transfer numeric data from paper tape or keyboard entry to random access file(s) in matrix form.

Before running the program, open files of sufficient length to accomodate the data. As each row of the matrix requires two records, there must be twice the number of records of file space available as there are rows in the matrix.

For example: If a matrix with 140 rows is to be entered into file, 140 X 2 = 280 records are required. A 2000B file has a maximum of 128 records, thus open two files of 128 records each and one of 24 records (128 X 2 + 24 = 280). NOTE: If more than one file is required then all but the last file must be of length 128 records. Be sure to declare the file(s) into which the data is to be read by:

```
10 FILES  FILE 1, FILE 2, FILE 3, ...
```

As a BASIC program can access only 16 files and each of these may have a maximum of 128 records, the matrix is restricted to 16 X 128/2 = 1024 rows or fewer. As each row takes up 2 records, the number of columns is restricted to 32 X 2 = 64.

INSTRUCTIONS: Continued On Next Page.

SPECIAL CONSIDERATIONS:

These programs are written for use on an HP 2000B. HP 2000C/F users should open files limited to 64 words per record; e.g., OPE-FILE 1, 128, 64. Limitations -- Number of variables = number of columns of data matrix $\leqslant 64$. Number of subjects limited only by file space.

The major reference for this package is "Fortran Programming for the Behavioral Sciences" by D.J. Veldman published by Holt, Rinehart and Winston in 1967. Most of the programs are BASIC translations of the FORTRAN routines presented in this text (slight modifications were made where deemed necessary; these are noted in the individual program documentation).

ACKNOWLEDGEMENTS:

Warren Nelson
University of Lethbridge

INSTRUCTIONS: Continued

If data is to be read in from paper tape:

1. Each data item on a line is to be separated from the next item by a comma.
2. An X-OFF character must appear at the end of each line.
3. Each row of the matrix must begin on a new line.
4. A row of the matrix may take more than one line.
5. No line should end in a comma.

EXAMPLE: To input the following matrix of 4 rows and 5 columns:

Row #1	1	2	3	4	5
Row #2	6	7	8	9	10
Row #3	11	12	13	14	15
Row #4	16	17	18	19	20

The data tape may appear as:

1, 2, 3 *(X-OFF)*
4, 5 *(X-OFF)*
6, 7, 8, 9, 10 *(X-OFF)*
11, 12, 13, 14 *(X-OFF)*
15 *(X-OFF)*
16, 17, 18, 19, 20 *(X-OFF)*

EDITM

This program will edit data stored by MATIN or a similar routine. Data may be listed, changed, added, deleted, or dumped to paper tape.

Before running, declare the files of data by:

```
10 FILES    FILE 1, FILE 2, FILE 3, ...
```

If data is to be added, then additional space must be opened and declared. Two records must be opened for each row added.

COMMAND	FUNCTION
LIST	Causes listing of a specified portion of the matrix.
CHANGE	Allows one row of data to be changed.
ADD	Allows one row to be inserted into the matrix.
DELETE	Causes a specified row to be deleted from the matrix.
DUMP	Causes the matrix to be output onto paper tape in a form acceptable to MATIN.

INSTRUCTIONS: Continued

II. FACTOR ANALYSIS

CODING OF DATA

The general form of the data is a matrix with NS rows and NV columns (where NS is the number of subjects and NV is the number of variables). Missing data must be coded with a constant that is not a valid observation (e.g., 999); blanks are not an acceptable means of indicating missing data as the BASIC language ignores these. As the data is to be entered by the program MATIN, the data tape must conform to the standards as given.

RUNNING THE FACTOR ANALYSIS CHAIN

The factor analysis chain requires a number of standard files to be opened before running. Do this by typing:

```
OPEN-CORR,128
OPEN-CORR1,128
OPEN-VEC,128
OPEN-S,128  (Required only if there is missing data.)
```

FACTOR ANALYSIS CHAIN

This chain consists of 4 separate BASIC programs linked together by the chain command. As well as the mean and standard deviation of each variable, the correlation matrix, the trace of the correlation matrix, eigenroots, percentage of total variance for each component, the unrotated principal axis factor loadings, percentage of total variance for each factor, the percentage of variance of each variable extracted and the V load matrix is output.

The number of variables is restricted to 64 or fewer.
The number of subjects is restricted to 768 or fewer.

CORS

This program is one of two alternative programs in the factor analysis chain. CORS computes means, sigmas and intercorrelations from data stored in file by MATIN or a similar routine; no missing data is allowed.

MDRS

This program is one of two alternative programs in the factor analysis chain. MDRS computes means, sigmas and intercorrelations from data stored in file by MATIN or a similar routine; missing data is allowed.

SEVS

This program, the third in the factor analysis chain, extracts eigenroots and denormal vectors from a symmetric matrix.

The logic of this program has been changed slightly from that given by Veldman. Instead of a fixed number of iterations (25) to compute an eigen vector, the iteration will continue until 59 is less than .0001.

$$S9 = \sqrt{\sum_{i=1}^{NV} (U_i - V_i)^2}$$

Where U and V are successive vectors. This method will eliminate unnecessary calculations and will also increase accuracy.

VORS

This program, the fourth in the factor analysis chain, does orthogonal Verimax rotation of factor axes.

INSTRUCTIONS: Continued

MEANS, STANDARD DEVIATIONS AND CORRELATIONS (CORS OR MDRS)

A. No Missing Data Allowed.

a. Mean $\mu_j = \frac{\Sigma X_{kj}}{N}$

b. Standard Deviation $\sigma_j^2 = \frac{\Sigma X^2_{kj}}{N} - \mu_j^2$

c. Correlation $r_{ij} = \frac{\Sigma X_{ki} X_{kj}/N - \mu_i \mu_j}{\sigma_i \sigma_j}$

B. Missing Data Allowed.

a. Mean $\mu_i = \frac{\Sigma X_{ki}}{N}$

b. Standard Deviation $\sigma_i^2 = \frac{\Sigma X^2_{ki}}{N} - \mu_i^2$

c. Correlation $r_{ij} = \frac{\Sigma X_{ki} Y_{kj}/N_{ij} - \mu_{i(j)} \mu_{j(i)}}{\sigma_{i(j)} \sigma_{j(i)}}$

where

1. $\mu_{i(j)}$ is the mean of the i^{th} variable taken over those subjects with valid data in the j^{th} variable.

2. $\sigma_{i(j)}$ is the standard deviation of the i^{th} variable taken over those subjects with valid data in the j^{th} variable.

3. N_{ij} is the number of subjects with valid data in the i^{th} variable and the j^{th} variable.

EXTRACTION OF EIGENROOTS AND DENORMAL VECTORS (SEVS)

This routine is based upon a procedure originally outlined by Hotelling. If in a given problem M is the number of variables and K is the number of factors extracted, then the trace (T) of the correlation matrix (R) is given by:

$T = U_M' R_{\Delta M} U_M$

When all the factors are extracted, the trace will also equal the sum of the eigenvalues (E) thus:

$T = U_K' E_K$ when $K = M$

If $K < M$ the percentage of the variance in R extracted by the K factors is given by:

$P = U_K' E_K T^{-1} 100$

If V_{MK} is the matrix of factor loadings:

$E_{\Delta K} = V_{KM}' V_{MK}$

The column vectors of V may be normalized by:

$F_{MK} = V_{MK} E_{\Delta K}^{-1/2}$

INSTRUCTIONS: Continued

If all M possible factors are extracted from R then:

$$R_{MM}V_{MK} = V_{MK}E_{\Delta K}$$

and

$$R_{MM}F_{MK} = F_{MK}E_{\Delta K}$$

If K < M then the outer products of V and F may only approximate R:

$$\hat{R}_{MM} = V_{MK}V'_{KM} = F_{MK}E_{\Delta K}F'_{KM}$$

As recommended by R. Kaiser only those eigenvalues exceeding 1.0 are retained. Unlike Veldman's version which utilizes a fixed number iterations to obtain the eigenvectors, this program requires the square root of the sum of the squares of the differences of successive vectors be less than .0001.

VARIMAX ROTATION OF FACTOR AXES (VORS)

The computational procedure used was derived from a formula suggested by Kaiser.

$$Y_{NK} = Z_{NM}V_{MK}E^{-2}_{\Delta K}V'_{KM}W_{MK}$$

where
V is a matrix of unrotated loadings.
E is a diagonal matrix of roots.
W is the Varimax rotated matrix of loadings.

III. MULTIPLE DISCRIMINANT ANALYSIS

CODING OF DATA

The general form of the data is a matrix with NS rows and NV columns (where NS is the number of subjects and NV is the number of variables). No missing data is allowed. As the data is to be entered by the program MATIN, the data tape must conform to the standards as given in that program's documentation.

RUNNING THE MULTIPLE DISCRIMINANT ANALYSIS CHAIN

The multiple discriminant analysis chain requires a number of standard files to be opened before running. Do this by typing:

```
OPEN-A,128
OPEN-W,128
OPEN-C,128
OPEN-S,128
```

MULTIPLE DISCRIMINANT ANALYSIS CHAIN

This chain consists of 3 separate BASIC programs linked together by the chain command. The general procedure is based on "Multivariate Procedures for the Behavioral Sciences" by W.W. Cooley and P.R. Lohnes with major modifications by Veldman. The direct factoring of $W^{-1}A$ and the internal computation of correlations between original variables and discriminant functions are the responsibility of Veldman. Discriminant score weights are followed by the correlations between the original variables and discriminant functions. Wilks' Lambda is computed and tested for significance. Chi-square tests are performed for each discriminant function. Group centroids and univariate analysis of variance are calculated for the original variables.

The number of subjects is restricted to 768 or fewer.
The number of variables is restricted to 64 or fewer.

DISCRM

This program, the first of three in the Multiple Discriminant Analysis Chain, reads the raw data from file, accumulates sums and cross products, and computes covariance.

INSTRUCTIONS: Continued

AEVS

This program, the second of three in the Multiple Discriminant Analysis Chain, extracts roots and vectors from a square asymmetric matrix.

This routine is much like the program SEVS included in the Factor Analysis Chain, except that both "right" and "left" eigenvectors are extracted and their outer product is used to deflate the A matrix after extraction of each root.

DISC2

This program, the last of three in the Multiple Discriminant Analysis Chain, computes discriminant-score weights, correlations of discriminant and original variables, Wilks' Lambda, F-ratio and probability as well as performing chi-square tests and univariate analysis of variance.

For each group of subjects the matrices P, T and W are formed from the raw scores X by:

$P_{MM} = X'_{MN}X_{NM}$ raw cross products

$T_M = X'_{MN}U_N$ raw sums

$W_{MM} = P_{MM} - T_M T'_M N^{-1}$ deviation cross products

where: N = the number of subjects in the group
M = the number of variables.

These matrices and the total N are accumulated over all groups and the following matrices are developed:

$C_{MM} = N^{-1}(P_{MM} - T_M T'_N N^{-1})$ covariance matrix

$A_{MM} = NC_{MM} - W_{MM}$ among-groups matrix

where P, T, W and N refer to the accumulated matrices.

Note that even though W^{-1} and A are symmetric matrices, $W^{-1}A$ is not and thus cannot be factored by the method used in the factor analysis chain.

IV. DOUBLE OR TRIPLE FACTOR ANALYSIS OF VARIANCE

CODING OF DATA

The general form of the data is a matrix with NS rows and NV columns (where NS is the number of subjects and NV is the number of variables). Missing data must be coded with a constant that is not a valid observation (e.g., 999); blanks are not an acceptable means of indicating missing data as the BASIC language ignores these. As the data is to be entered by the program MATIN, the data tape must conform to the standards as given.

RUNNING DOUBLE OR TRIPLE FACTOR ANALYSIS OF VARIANCE

Double or triple factor analysis of variance requires a number of standard files to be opened before running. Do this by typing:

```
OPEN-M,128
OPEN-L,128
OPEN-SX,128
OPEN-G,128
```

DOUBLE OR TRIPLE FACTOR ANALYSIS OF VARIANCE

This chain of two separate BASIC programs extends single classification analysis of variance to permit classification of subjects into "levels" on two or three independent variables simultaneously. Tests of significance are computed for each of the two or three "factors" as well as for their interactive effect upon the dependent variable.

INSTRUCTIONS: Continued

The number of subjects is restricted to 768 or fewer.

The method used is outlined by B.J. Winer in "Statistical Principles in Experimental Design" published in 1962 by McGraw-Hill. This method, unlike the usual routines, allows unequal numbers of subjects to be used in each cell of the design.

AVAR23

This program, the first of two in the Double or Triple Factor Analysis of Variance Chain, reads the raw data from file and computes cell variances and means as well as the number of valid observations for each variable.

"FORTRAN Programming for the Behavioral Sciences" by D.J. Veldman is the reference.

AVR23+

This program, the second in the Double or Triple Factor Analysis of Variance Chain, computes analysis of cell means, F-ratio and produces a source table and relevant cell means. The same reference is used as in AVAR23.

V. CROSS TABULATIONS

STRGIN

This program will transfer string data from paper tape to sequential file(s).

Before running the program declare the files into which the data is to be read by:

```
10 FILES    FILE 1, FILE 2, ...
```

in the order they are to be filled. Make certain that there is sufficient file space to contain all the data and that an X-OFF character ends each line of the paper tape. A string containing "EOT" as the first three letters will cause program completion.

The characters "Control Q", "Control Shift N", and "Control Shift O" are special control characters and should be avoided.

COUNTS

This program is designed to simulate the counting function of a card sorter. String data is read from file(s) and counts are performed on columns specified by the user. In addition, the user has the option of distribution(s) on:

1. Single columns
2. Two columns simultaneously
3. Three columns simultaneously

Before running, declare the files of data by:

```
10 FILES    FILE 1, FILE 2, ...
```

Make certain the files are ordered correctly (i.e., in such a manner that the last string of the last file begins with the letters EOT as this will cause completion of the count).

The maximum number of strings of data is limited to 99,999 while the maximum number of strings with a specific character in a given column is limited to 9,999.

RUN

I. MATRIX DATA LOADING

```
OPE-A,128
OPE-W,128
OPE-C,128
OPE-S,128

OPE-INPT,128
GET-MATIN
RUN\
10 FILES INPT
RUN
MATIN

HAVE YOU ENTERED THE FILES STATEMENT?YES
HOW MANY ROWS ARE THERE?16
HOW MANY COLUMNS ARE THERE?8

PLEASE MOUNT DATA TAPE AND MOVE READER CONTROL LEVER TO
START.

ROW # 1
?25,21,22,20,26,261←,19,23

ROW # 2
?260,30,30,26,28,20,24,28

ROW # 3
?20,25,20,23,18,24,21,29

ROW # 4
?30,28,29,29,28,23,28,30

ROW # 5
?23,25,29,19,20,27,28,28

ROW # 6
?28,27,30,22,19,25,30,26

ROW # 7
?28,24,27,27,17,21,30,26

ROW # 8
?25,29,29,27,26,25,26,25

ROW # 9
?26,30,30,24,29,24,14,29

ROW # 10
?28,29,30,26,25,28,30,28

ROW # 11
?24,28,30,29,27,23,21,28

ROW # 12
?26,29,26,27,28,19,30,27

ROW # 13
?30,27,26,24,25,21,28,25

ROW # 14
?29,29,29,28,25,19,30,27

ROW # 15
?29,25,28,26,24,21,30,29

ROW # 16
?29,26,30,20,25,20,30,28


MATRIX ENTERED INTO FILE(S).

DONE
```

```
GET-EDITM
10 FILES1← INPT
RUN
EDITM

HOW MANY ROWS ARE THE IN THE MATRIX?16
HOW MANY COLUMNS?8
DO YOU WISH A LISTING OF THE COMMANDS?YES

          COMMAND   FUNCTION

          ..................................................
          LIST      CAUSES A LISING OF SPECIFIED ROWS OF THE MATRIX

          CHANGE    ALLOWS ONE ROW OF DATA TO BE CHANGED

          ADD       ALLOWS ONE ROW TO BE INSERTED INTO THE MATRIX

          DELETE    CAUSES A SPECIFIED ROW TO BE DELETED

          DUMP      CAUSES THE MATRIX TO BE OUTPUT ONTO PAPER
                    TAPE IN A FORM ACCEPTABLE TO 'MATIN'
COMMAND?LIST
FROM WHICH ROW TO WHICH ROW DO YOU WANT LISTED (EG.4,6)?2,2

ROW # 2
 260            30              30              26              28
 20             24              28
MORE EDITING?YES
COMMAND?CHANGE
WHICH ROW DO YOU WISH TO CHANGE?2
PRESENT STATUS OF ROW 2

 260            30              30              26              28
 20             24              28
DO YOU STILL WISH TO CHANGE THE ROW?YES
INPUT NEW DATA (EG. 3,4,5,6,7)

?26,30,30,26,28,20,24,28
MORE EDITING?NO

DONE
```

II. FACTOR ANALYSIS

```
OPE-CORR,128
OPE-C-VEC,128
OPE-CORR1,128

GET-CORS
120 FILES INPT
RUN
CORS


HOW MANY SUBJECTS ARE THERE?16
HOW MANY VARIABLES ARE THERE?8
MAXIMUM NUMBER OF FACTORS?3
MEANS

 26.625         27              27.8125         24.8125         24.375
 22.875         26.1875         27.25


SIGMAS

 2.6897         2.4238          2.9202          3.1269          3.6891
 2.7585         4.8117          1.7854
```

CORRELATION MATRIX

1	.1821	.4685	.2666	.2409
-.3938	.556	-.0325		
.1821	1	.6181	.5525	.5452
-.215	.0161	.39		
.4685	.6181	1	.2631	.3198
-.0495	.2694	.3087		
.2666	.5525	.2631	1	.3529
-.3723	.1394	.2547		
.2409	.5452	.3198	.3529	1
-.2104	-.3032	.1091		
-.3938	-.215	-.0495	-.3723	-.2104
1	-.2808	-.1079		
.556	.0161	.2694	.1394	-.3032
-.2808	1	-.0127		
-.0325	.39	.3087	.2547	.1091
-.1079	-.0127	1		

PRINCIPAL AXIS ANALYSIS OF CORRELATION MATRIX

TRACE = 8

70.57 PCT OF TRACE WAS EXTRACTED BY 3 ROOTS

EIGENROOTS

1	2	3	4	5
2.9197	1.662	1.0637		

PCT OF TOT VAR FROM EACH COMPONENT

1	2	3	4	5
36.5	20.8	13.3		

UNROTATED PRINCIPAL AXIS FACTOR LOADINGS

1	2	3	4	5
.6125	-.5974	-.0757		
.8054	.3802	.1163		
.7276	-.0058	.4891		
.701	.0999	-.2479		
.5851	.4818	-.3937		
-.5045	.3246	.5502		
.3156	-.8384	.2481		
.4228	.3318	.4737		

```
VARIMAX ROTATION ANALYSIS


PCT. OF TOT. VAR. FROM EACH FACTOR

 1               2               3               4               5

 24.548          23.688          22.3313


PCT. OF VAR OF EACH VARIABLE EXTRACTED
 1               2               3               4               5


 73.77           80.67           76.87           56.28           72.95
 66.26           86.42           51.33

V LOAD
 1               2               3               4               5

 .3032          -.7969           .104
 .5719          -.015            .6924
 .1591          -.354            .7861
 .672           -.1966           .2692
 .7976           .2069           .2247
-.628            .4722           .2127
-.185           -.9073           .0815
 .0554           .0825           .7095
```

```
DONE
KIL-INPT
KIL-CORR
KIL-VEC
KIL-CORR1
```

III. MULTIPLE DISCRIMINANT ANALYSIS

```
OPE-A,128
OPE-W,128
OPE-C,128
OPE-S,128

OPE-INPT,128
GET-MATIN
10 FILES INPT
RUN
MATIN

HAVE YOU ENTERED THE FILES STATEMENT?YES
HOW MANY ROWS ARE THERE?16
HOW MANY COLUMNS ARE THERE?8

PLEASE MOUNT DATA TAPE AND MOVE READER CONTROL LEVER TO
START.

ROW # 1
?25,21,22,20,26,26,19,23

ROW # 2
?20,25,20,23,18,24,21,29

ROW # 3
?26,30,30,26,28,20,24,28
```

```
ROW # 4
?30,28,29,29,28,23,28,30

ROW # 5
?25,29,29,27,262←,25,26,25

ROW # 6
?2
  26,30,30,24,29,24,14,29

ROW # 7
?28,29,30,26,25,28,30,28

ROW # 8
?24,28,30,29,27,23,21,28

ROW # 9
?26,29,26,27,28,19,30,27

ROW # 10
?30,27,26,24,25,21,28,25

ROW # 11
?29,29,29,28,25,19,30,27

ROW # 12
?29,25,28,26,24,21,30,29

ROW # 13
?29,26,30,20,25,20,30,28

ROW # 14
?23,25,29,19,20,27,28,28

ROW # 15
?28,27,30,22,19,25,30,26

ROW # 16
?28,24,27,27,17,21,30,26


MATRIX ENTERED INTO FILE(S).

DONE

GET-EDITM
10 FILES INPT
RUN
EDITM

HOW MANY ROWS ARE THE IN THE MATRIX?16
HOW MANY COLUMNS?8
DO YOU WISH A LISTING OF THE COMMANDS?YES

          COMMAND   FUNCTION

          ....................................................
          LIST      CAUSES A LISING OF SPECIFIED ROWS OF THE MATRIX

          CHANGE    ALLOWS ONE ROW OF DATA TO BE CHANGED

          ADD       ALLOWS ONE ROW TO BE INSERTED INTO THE MATRIX

          DELETE    CAUSES A SPECIFIED ROW TO BE DELETED

          DUMP      CAUSES THE MATRIX TO BE OUTPUT ONTO PAPER
                    TAPE IN A FORM ACCEPTABLE TO 'MATIN'
COMMAND?CHANGE
WHICH ROW DO YOU WISH TO CHANGE?6
PRESENT STATUS OF ROW 6

 226            30              30              24              29
 24             14              29
DO YOU STILL WISH TO CHANGE THE ROW?YES
INPUT NEW DATA (EG. 3,4,5,6,7)
```

```
?26,30,30,24,29,24,14,29
MORE EDITING?YES
COMMAND?LIST
FROM WHICH ROW TO WHICH ROW DO YOU WANT LISTED (EG.4,6)?9,9

ROW # 9
 26             29             26             27             28
 19             30             27
MORE EDITING?NO

DONE


GET-DISCRM
70 FILES INPT
RUN
DISCRM

NUMBER OF VARIABLES?8


NUMBER OF GROUPS?3


NUMBER OF SUBJECTS IN GROUP 1     ?2


NUMBER OF SUBJECTS IN GROUP 2     ?11


NUMBER OF SUBJECTS IN GROUP 3     ?3






TRACE = 14.2885


 100  PCT. OF TRACE EXTRACED BY 2    ROOTS.


WILKS LAMBDA = .019

D.F. =  16    AND  12

F-RATIO = 4.665

P = .0055

ROOT           % VARIANCE     CHI-SQUARE     D.F.           P

 1               76.37         26.014        9             .0026
 2               23.63         15.501        7             .0306


CENT.
 38.708         12.3233

 51.6327        10.9154

 47.3464        5.6343
```

COREL

```
 .6235          -.0843

 .7825           .2026

 .8591          -.3633

 .5501           .2559

 .5551           .7362

-.4058          -.2013

 .3773          -.4533

 .3388           .104

UNIVARIATE F-TESTS D.R.B.=  2    DFW=  13

VARIABLE       F-RATIO          P
 1               3.6817       .0531
 2               9.4527       .0032
 3              22.7661       .0002
 4               3.1689       .0746
 5              15.2046       .0006
 6               1.4474       .2703
 7               2.6416       .1078
 8                .8323       .5398

MEAN

 1              2               3

 22.5           27.4545         26.3333
 23             28.1818         25.3333
 21             28.8182         28.6667
 21.5           26              22.6667
 22             26.3636         18.6667
 25             22.0909         24.3333
 20             26.4545         29.3333
 26             27.6364         26.6667

DONE

KIL-A
KIL-W
KIL-C
KIL-S
KIL-INPT
```

IV. DOUBLE OR TRIPLE FACTOR ANALYSIS OF VARIANCE

```
OPE-M,128
OPE-L,128
OPE-G,128
OPE-SX,128
OPE-INPT,128
10 FILES INPT
GET-MATIN
10 FILES INPT
RUN
MATIN
```

```
HAVE YOU ENTERED THE FILES STATEMENT?YES
HOW MANY ROWS ARE THERE?13
HOW MANY COLUMNS ARE THERE?1

PLEASE MOUNT DATA TAPE AND MOVE READER CONTROL LEVER TO
START.

ROW # 1
?30

ROW # 2
?28

ROW # 3
?25

ROW # 4
?27

ROW # 5
?2

ROW # 6
?29

ROW # 7
?29

ROW # 8
?28

ROW # 9
?29

ROW # 10
?27

ROW # 11
?29

ROW # 12
?25

ROW # 13
?26


MATRIX ENTERED INTO FILE(S).

DONE


GET-EDT-ITM
10 FILES INPT
RUN
EDITM

HOW MANY ROWS ARE THE IN THE MATRIX?13
HOW MANY COLUMNS?1
DO YOU WISH A LISTING OF THE COMMANDS?YES

          COMMAND   FUNCTION

          ..................................................
          LIST      CAUSES A LISING OF SPECIFIED ROWS OF THE MATRIX

          CHANGE    ALLOWS ONE ROW OF DATA TO BE CHANGED

          ADD       ALLOWS ONE ROW TO BE INSERTED INTO THE MATRIX

          DELETE    CAUSES A SPECIFIED ROW TO BE DELETED

          DUMP      CAUSES THE MATRIX TO BE OUTPUT ONTO PAPER
                    TAPE IN A FORM ACCEPTABLE TO 'MATIN'
```

```
COMMAND?CHANGE
WHICH ROW DO YOU WISH TO CHANGE?5
PRESENT STATUS OF ROW 5

 2             DO YOU STILL WISH TO CHANGE THE ROW?YES
INPUT NEW DATA (EG. 3,4,5,6,7)

?24
MORE EDITING?NO

DONE


GET-AVAR23
80 FILES INPT
RUN
AVAR23

NUMBER OF DEPENDENT VARIABLES?1
NUMBER OF LEVELS FOR THE A FACTOR?2
NUMBER OF LEVELS FOR THE B FACTOR?2
NUMBER OF LEVELS FOR THE C FACTOR?1
NUMBER OF SUBJECTS PER ABC CELL?9999
MISSING DATA CODED WITH WHAT NUMBER?-999
NUMBER OF SUBJECTS IN GROUP 1     ?2
NUMBER OF SUBJECTS IN GROUP 2     ?4
NUMBER OF SUBJECTS IN GROUP 3     ?3
NUMBER OF SUBJECTS IN GROUP 4     ?4



ANALYSIS OF VARIABLE 1

SOURCE         M.S.            D.F.           F-RATIO          P

TOTAL          3.587            12
BETWEEN        5.625            3
   A           .022             1                .0076          .9302
   B           16.334           1                5.6183         .0403
  AB           .519             1                .1784          .6842
WITHIN         2.907            9

MEANS FOR ALL EFFECTS.

A MAIN

 27.625         27.7083


B MAIN

 28.8333        26.5


A BY B

ROW # 1
 29             26.25

ROW # 2
 28.6667        26.75

SUBJECTS PER CELL. BLOCKS =C LEVELS.

  AB

 ROW #   1
 2              4
 ROW #   2
 3              4
```

```
DONE
KIL-INPT
KIL-M
KIL-L
KIL-SX
KIL-G




OPE-INPT,128
OPE-M,128
OPE-L,128
OPE-SX,128
OPE-G,128

GET-MATIN
RUN
MATIN

HAVE YOU ENTERED THE FILES STATEMENT?NO
THIS PROGRAM REQUIRES THE ENTERING OF A FILES STATEMENT
BEFORE RUNNING; PLEASE CHECK THE PROGRAM DOCUMENTATION.

DONE
10 FILES INPT
RUN
MATIN

HAVE YOU ENTERED THE FILES STATEMENT?YES
HOW MANY ROWS ARE THERE?16
HOW MANY COLUMNS ARE THERE?2

PLEASE MOUNT DATA TAPE AND MOVE READER CONTROL LEVER TO
START.

ROW # 1
?25    ,-999

ROW # 2
?26    , 30

ROW # 3
?20    ,-999

ROW # 4
?30    ,28

ROW # 5
?23    , 25

ROW # 6
?28    , 27

ROW # 7
?28    , 24

ROW # 8
?25    , 29

ROW # 9
?26    ,-999

ROW # 10
?28    , 29

ROW # 11
?24    , 28

ROW # 12
?26    , 29

ROW # 13
?30    , 27
```

```
ROW # 14
?29    , 29

ROW # 15
?29    , 29

ROW # 16
?29    , 26


MATRIX ENTERED INTO FILE(S).

DONE

GET-EDITM
10 FILES INPT
RUN
EDITM

HOW MANY ROWS ARE THE IN THE MATRIX?16
HOW MANY COLUMNS?2
DO YOU WISH A LISTING OF THE COMMANDS?YES

          COMMAND   FUNCTION

          ...................................................
          LIST      CAUSES A LISING OF SPECIFIED ROWS OF THE MATRIX

          CHANGE    ALLOWS ONE ROW OF DATA TO BE CHANGED

          ADD       ALLOWS ONE ROW TO BE INSERTED INTO THE MATRIX

          DELETE    CAUSES A SPECIFIED ROW TO BE DELETED

          DUMP      CAUSES THE MATRIX TO BE OUTPUT ONTO PAPER
                    TAPE IN A FORM ACCEPTABLE TO 'MATIN'
COMMAND?LIST
FROM WHICH ROW TO WHICH ROW DO YOU WANT LISTED (EG.4,6)?1,16

ROW # 1
 25           -999

ROW # 2
 26            30

ROW # 3
 20           -999

ROW # 4
 30            28

ROW # 5
 23            25

ROW # 6
 28            27

ROW # 7
 28            24

ROW # 8
 25            29

ROW # 9
 26           -999

ROW # 10
 28            29

ROW # 11
 24            28

ROW # 12
 26            29

ROW # 13
 30            27
```

```
ROW # 14
 29              29

ROW # 15
 29              29

ROW # 16
 29              26
MORE EDITING?YES
COMMAND?DUMP
TURN ON TAPE PUNCH AND PRESS 'HERE IS' FOR LEADER
 25    ,-999
 26    , 30
 20    ,-999
 30    , 28
 23    , 25
 28    , 27
 28    , 24
 25    , 29
 26    ,-999
 28    , 29
 24    , 28
 26    , 29
 30    , 27
 29    , 29
 29    , 29
 29    , 26
MORE EDITING?YES
COMMAND?CHANGE
WHICH ROW DO YOU WISH TO CHANGE?15
PRESENT STATUS OF ROW 15

 29              29              DO YOU STILL WISH TO CHANGE THE ROW?YES
INPUT NEW DATA (EG. 3,4,5,6,7)

?29    , 25
MORE EDITING?NO

DONE



GET-AVAR23
80 FILES INPT
RUN
AVAR23

NUMBER OF DEPENDENT VARIABLES?2
NUMBER OF LEVELS FOR THE A FACTOR?2
NUMBER OF LEVELS FOR THE B FACTOR?2
NUMBER OF LEVELS FOR THE C FACTOR?2
NUMBER OF SUBJECTS PER ABC CELL?2
MISSING DATA CODED WITH WHAT NUMBER?-999



ANALYSIS OF VARIABLE 1
```

SOURCE	M.S.	D.F.	F-RATIO	P
TOTAL	7.717	15		
BETWEEN	6.25	7		
A	16.	1	1.7778	.2177
B	16.	1	1.7778	.2177
C	1.	1	.1111	.745
AB	6.25	1	.6944	.5667
AC	2.25	1	.25	.6342
BC	2.25	1	.25	.6342
ABC	0	1	0	1
WITHIN	9	8		

```
MEANS FOR ALL EFFECTS.

A MAIN

 25.625         27.625


B MAIN

 25.625         27.625

C MAIN

 26.875         26.375




A BY B

ROW # 1
 25.25          26

ROW # 2
 26             29.25

A BY C

ROW # 1

 25.5           25.75
ROW # 2

 28.25          27


B BY C

ROW # 1
 26.25          25
ROW # 2
 27.5           27.75


CELL MEANS   BLOCKS = C LEVELS.


AB

ROW # 1
 25.5           25.5
ROW # 2
 27             29.5


AB

ROW # 1
 25             26.5
ROW # 2
 25             29
```

SUBJECTS PER CELL. BLOCKS =C LEVELS.

AB

ROW # 1
2 2
ROW # 2
2 2

AB

ROW # 1
2 2
ROW # 2
2 2

ANALYSIS OF VARIABLE 2

SOURCE	M.S.	D.F.	F-RATIO	P
TOTAL	3.515	12		
BETWEEN	3.526	7		
A	.045	1	.013	.9098
B	16.409	1	4.6883	.0815
C	3.682	1	1.0519	.3538
AB	.409	1	.1169	.7431
AC	.409	1	.1169	.7431
BC	.045	1	.013	.9098
ABC	3.682	1	1.0519	.3538
WITHIN	3.5	5		

MEANS FOR ALL EFFECTS.

A MAIN

27.625 27.75

B MAIN

28.875 26.5

C MAIN

28.25 27.125

A BY B

ROW # 1
29 26.25

ROW # 2
28.75 26.75

A BY C

ROW # 1

28 27.25
ROW # 2

28.5 27

```
B BY C

ROW # 1
 29.5           28.25
ROW # 2
 27             26


CELL MEANS   BLOCKS = C LEVELS.


AB

ROW # 1
 30             26
ROW # 2
 29             28


AB

ROW # 1
 28             26.5
ROW # 2
 28.5           25.5


SUBJECTS PER CELL. BLOCKS =C LEVELS.

  AB

 ROW #   1
 1              2
 ROW #   2
 1              2


  AB

 ROW #   1
 1              2
 ROW #   2
 2              2


DONE
KIL-INPT
KIL-M
KIL-L
KIL-SC
NO SUCH ENTRY
KIL-SX
KIL-G
```

V. CROSS TABULATIONS

```
GET-STRGIN



10 FILES TEST1,TEST2
RUN
STRGIN

HAVE YOU ENTERED THE FILES STATEMENT ('YES' OR 'NO')?YES
ARE YOU CERTAIN THE FILES ARE IN THE ORDER YOU DESIRE AND
THAT THEY ARE OF SUFFICIENT LENGTH TO ACCOMODATE ALL YOUR DATA
?YES
HOW MANY FILES HAVE YOU DECLARED?2

MOUNT PAPER TAPE AND MOVER TAPE READER CONTROL TO 'START'.
?1234567890
?1234567890
?AAAAAAAAAA
?BBBBBBBBBB
?CCCCCCCCCC
?DDDDDDDDDD
?EEEEEEEEEE
?FFFFFFFFFF
?GGGGGGGGGG
?HHHHHHHHHH
?IIIIIIIIII
?123456789012345
?123456789012345
?ABCDEFGHIJKLMNO
?PQRSTUVWXYZ!#$%
?EOT
?EOT
?EOT
?EOT
?EOT
DATA STORED IN FILE.

DONE


GET-COUNTS
10 FILES TS←\
10 FILES TEST1,TEST2
RUN
COUNTS

HAVE YOU ENTERED THE FILES STATEMENT ('YES' OR 'NO')?YES
ARE YOU CERTAIN THE FILES ARE NAMED IN THE CORRECT ORDER?YES


HOW MANY FILES OF DATA HAVE YOU USED?2

WHICH OF THE FOLLOWING COUNTS DO YOU WISH
          1. COUNT ON A SINGLE COLUMN
          2. COUNT ON TWO COLUMNS
          3. COUNT ON THREE COLUMNS

TYPE 'SINGLE','TWO' OR 'THREE'?SINGLE
ON HOW MANY COLUMNS DO YOU WISH TO COUNT ?3
WHICH COLUMNS (EG. 1,4,7,10,14,6,34)
?2,6,3


COUNT ON COLUMN  2
```

CHAR.	1	2	3	4	5	6	7	8	9	0	OTHER	TOT.
FREQ.	0	4	0	0	0	0	0	0	0	0	11	15
PCT.	0.0	26.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	73.3	100.0

COUNT ON COLUMN 6

CHAR.	1	2	3	4	5	6	7	8	9	Ø	OTHER	TOT.
FREQ.	Ø	Ø	Ø	Ø	Ø	4	Ø	Ø	Ø	Ø	11	15
PCT.	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	26.7	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	73.3	100.Ø

COUNT ON COLUMN 3

CHAR.	1	2	3	4	5	6	7	8	9	Ø	OTHER	TOT.
FREQ.	Ø	Ø	4	Ø	Ø	Ø	Ø	Ø	Ø	Ø	11	15
PCT.	Ø.Ø	Ø.Ø	26.7	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	73.3	100.Ø

DO YOU WISH TO DO ANY OTHER COUNTS?YES
PLEASE TYPE 'SINGLE','TWO', OR 'THREE'?TWO
ON WHICH COLUMNS DO YOU WISH TO COUNT (EG. 4,17)?2,5

COL.: 2 :	COLUMN 5 1	2	3	4	5	6	7	8	9	Ø	OTHER	TOTAL
1 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
2 :	Ø	Ø	Ø	Ø	4	Ø	Ø	Ø	Ø	Ø	Ø	4
3 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
4 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
5 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
6 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
7 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
8 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
9 :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
Ø :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø
O :	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	11	11
T :	Ø	Ø	Ø	Ø	4	Ø	Ø	Ø	Ø	Ø	11	15

PERCENTAGE DISTRIBUTION

COL.: 2 :	COLUMN 5 1	2	3	4	5	6	7	8	9	Ø	OTHER	TOTAL
1 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
2 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	##.#	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	100.Ø
3 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
4 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
5 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
6 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
7 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
8 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
9 :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
O :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	##.#	100.Ø
T :	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	26.6	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	73.3	100.Ø

ACNODE
36057

TITLE: AC CIRCUIT ANALYSIS PROGRAM

DESCRIPTION: This program computes node voltages by inverting an admittance matrix created from a nodal description of an electronic circuit. Circuit elements allowed include resistors, inductors, transformers, independent current sources, and voltage current sources.

INSTRUCTIONS: Data line numbers 1-999 allowed

Data R$ -- Alpha or numeric designator code
Data M,N, -- # of elements, # of nodes
Data J_5, G_1, G_2,...,G_n -- # of node voltages to be printed out, nodes desired
Data L$, F_1, F_2, S -- Log or Linear frequency step, start frequency, stop frequency, step size or steps/decade
Data-Circuit elements -- statements in any order

Additional information attached.

SPECIAL CONSIDERATIONS:

Works with HP 7200A plotter
Limited to 10 nodes (other than ground -- node "0")
Unlimited # of elements
Transformers non-ideal ($.0001 \leq k \leq .9999$)
Matrix inversion can blow up if all elements connected to a node are lossless and resonant at frequency of interest

ACKNOWLEDGEMENTS: Jim Thomason
Hewlett-Packard/Microwave Division

This program computes node voltages (magnitude and phase), over a given frequency range, from a list of circuit elements. The program gathers the whole circuit into an admittance matrix, based on the element connections and values, and then solves for node voltages at each frequency.

Elements allowed include Resistors, Capacitors, Inductors, Transformers (non-ideal), Independent Current Sources, and Voltage-Dependent Current Sources - (*ACNODE also allows admittance elements).

```
10 DATA R$            where R$ = "A" for alphanumeric
                                 element descriptions
                            R$ = "N" for numeric element
                                 descriptions

20 DATA M, N          where M  = No. of circuit elements
                            N  = Highest numbered node

30 DATA J,J1,J2,.... where J  = No. of nodes for which
                                 output is desired; J1,
                                 J2 are the nodes included in J
```

<u>NOTE</u>: J = 0 causes all node voltages to be printed.
J1, J2,... are not entered in this case.

```
40 DATA L,F1,F2,S     where L  = "LOG" or "LIN" (1 or 2)
                            F1 = Start frequency
                            F2 = Stop frequency
                            S  = Steps per decade (Log) or
                                 frequency increment (Lin)
```

<u>NOTE</u>: Use numeric entry for L where numeric description of circuit elements is used.

<u>NOTE</u>: Frequency increment must be positive.

C. Circuit elements may be entered in any order after the above data is entered. This is possible because all entries are converted to admittance and placed in the circuit admittance matrix according to node numbers.

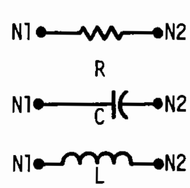
D. Data input form for circuit elements.

RESISTOR

```
100 DATA "R21", N1, N2, X       where N1, N2 are the two
or                              nodes connected to the
100 DATA 1, N1, N2, X           circuit element
                                X = value of the element
                                    (ohms, farads, henries)
```

CAPACITOR

```
100 DATA "CE2", N1, N2, X
or
100 DATA 2, N1, N2, X
```

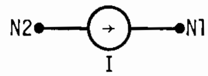


INDUCTOR

```
100 DATA "L", N1, N2, X
or
100 DATA 5, N1, N2, X
```

INDEPENDENT CURRENT SOURCE

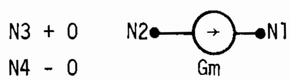
```
100 DATA "I", N1, N2, X, Y
or
100 DATA 3, N1, N2, X, Y
```



where N1 = "INTO" node
N2 = "OUT OF" node
X = Real part of current (Amps)
Y = Imaginary part of current (Amps)

DEPENDENT CURRENT SOURCE

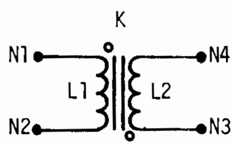
```
100 DATA "IV", N1, N2, N3, N4, X
or
100 DATA 4, N1, N2, N3, N4, X
```



where N1, N2 as above
N3 = Positive control node
N4 = Negative control node
X = Real part of Gm (mhos)

TRANSFORMER

```
100 DATA "TRF", N1, N2, N3, N4, L1, L2, K
or
100 DATA 7, N1, N2, N3, N4, L1, L2, K
```



where N1 = Dotted primary node
N2 = Other primary node
N3 = Dotted secondary node
N4 = Other secondary node

<u>NOTE</u>: $0.0001 \leq K \leq 0.9999$
Do <u>not</u> use K=0 or K=1.0

L1 = Primary inductance (Henries)
L2 = Secondary inductance (Henries)
K = Coupling coefficient

<u>PSEUDO VOLTAGE SOURCES</u>

The program does not allow for some useful elements, such as voltage sources or current dependent current sources, but good approximations for these elements are usually possible.

For example, a 1.0 amp current source paralleled with 1.0 ohm makes a reasonable 1.0 volt source for a circuit with input impedance greater than 100 ohms. Also, by putting 1.0 ohm and one extra node in series with the voltage control path, a current-controlled current source may be fashioned.

These values should be fashioned to fit the parameters of the individual circuit. One should be careful to avoid the temptation to use sources such as 1000 amps and .001 ohms = 1.000 volts, because the values may cause resolution errors in the computer.

<u>INPUT AND OUTPUT IMPEDANCE</u>

The impedance looking into any node (from ground) may be found by driving that node with a 1.0 amp current source and removing all other independent sources. The voltage at the driven node will be equal to the impedance looking into the node.

RUNNING THE PROGRAM

A. The data can be merged with the program in several ways.

1) The main program may be loaded into core and then the data entered via the keyboard or punched tape.

2) The data may be stored under a program name. In this case, the data statements should be loaded onto core first and then ACNODE is appended to the data.

For example, suppose the data statements are stored under the name "DATA1". The sequence of commands would be as follows:

```
   (HP)
GET-DATA1
APP-$ACNODE
RUN
```

USING THE HP 7200A PLOTTER

Turn on the plotter and position graph paper before asking for a plot.

The program will ask if you want graphical output and if you respond with (Y), it will ask which quantity (node voltage, dB, or phase) you wish to plot versus frequency. It also asks for the extreme values of that quantity, which will correspond to the top and bottom limits set on the plotter.

As soon as these questions are answered, the plot will begin. The teletype may be muted if desired during the plot, since its output will not normally be meaningful anyway. Disable the muting after the plot is finished to return system control to the teletype.

The horizontal scale, frequency, is plotted in log or linear mode, as requested in the data statement. Be sure that the graph paper you are using corresponds to that scale (i.e., do not use three decade log paper if you have asked for a five decade frequency range).

You may make as many plots or tables (on the terminal) as you like without changing the graph paper (by rerunning the program). The plotter will not respond to anything unless called by the program.

RUN

```
LIST
ACNODE

5  DATA "A"
10  DATA 20,10
20  DATA 3,1,4,10
30  DATA "LOG",1000,1.01E+07,2
40  DATA "I",1,0,1,0
50  DATA "R",1,0,1
60  DATA "R1",1,2,100
70  DATA "REB",2,3,375
80  DATA "RPI",3,5,1625
90  DATA "CPI",3,5,8.3E-11
100  DATA "RMU",3,4,1.4E+07
110  DATA "CMU",3,4,1.5E-12
120  DATA "IVQ1",5,4,3,5,.08
130  DATA "RO",4,5,71000.
140  DATA "R2",5,6,50
150  DATA "R3",6,0,2000
160  DATA "C1",0,6,4.7E-07
170  DATA "TRF1",4,0,7,0,.1,.2,.9999
180  DATA "C2",7,8,.000001
190  DATA "R4",8,0,1000
200  DATA "R5",8,9,1500
210  DATA "C3",9,0,1.E-09
220  DATA "R6",9,10,5000
230  DATA "C4",10,0,3.E-10
STOP

RUN
ACNODE

GRAPHICAL OUTPUT (HP 7200A PLOTTER): (Y OR N)?N
```

NODE	FREQUENCY	VOLTAGE	DB	PHASE
1	1000	.999992	0	0
4	1000	1.21881	1.719	-77.36
10	1000	1.69973	4.608	-69.58
1	3162.28	.999967	0	0
4	3162.28	3.81041	11.619	-113.9
10	3162.28	5.3669	14.594	-114.99
1	10000.	.999908	-.001	0
4	10000.	6.57101	16.353	-153.94
10	10000.	9.14808	19.227	-165.5
1	31622.8	.999887	-.001	0
4	31622.8	6.65723	16.466	179.53
10	31622.8	8.26111	18.341	142.39
1	100000.	.999884	-.001	0
4	100000.	5.34979	14.567	172.54
10	100000.	3.50764	10.9	83.64
1	316228.	.999883	-.001	0
4	316228.	4.65644	13.361	-179.83
10	316228.	.634542	-3.951	33.92
1	1.00000E+06	.999888	-.001	-.01
4	1.00000E+06	4.85668	13.727	-163.75
10	1.00000E+06	7.11578E-02	-22.956	7.15
1	3.16229E+06	.999934	-.001	-.03
4	3.16229E+06	7.93942	17.996	-142.8
10	3.16229E+06	7.64719E-03	-42.33	-11.84
1	1.00000E+07	.998157	-.016	-.14
4	1.00000E+07	26.4475	28.448	167.1
10	1.00000E+07	9.77322E-04	-60.199	-88.04

DONE

VOLUME III
CONTENTS

600 MANAGEMENT SCIENCES AND OPERATIONS RESEARCH

700 BUSINESS AND MANUFACTURING APPLICATIONS

600

700

VOLUME III
CONTENTS (Continued)

700 BUSINESS AND MANUFACTURING APPLICATIONS (Continued)

DECSN
36065

TITLE: TOP MANAGEMENT DECISION GAME

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.) An income statement for each team is printed out. The program recalculates sales units when the combination of production cost and beginning inventory are too low to meet sales units as generated by the first part of the program. Each income statement is completely formated to 7 significant digits and each income statement is printed on an 11 inch sheet of paper.

INSTRUCTIONS:

Data is put into lines 351 to 372. Line 370 is for beginning inventory for each team. This is to be entered in order as to team number. Line 371 is for training expense and it is to be entered the same as line 370. Line 372 is for units available for sale. This figure is from form IV line 3 and this also is to be entered the same as line 370 and line 371.

Line 370, at the start of the game, is 310000 for all teams. Line 372, at the start of the game, is 96000 for all teams. These lines are also printed in the new data printout block. Line 372, in the new data block, will only be ending inventory from the previous period and the beginning inventory for the next period less the new production for that period.

Because of line 371 there is no need to enter for training in the Data block. The (6) is automatically entered in the Data block.

The first time incentive is instituted it will be understated in the income statement and it should be corrected manually. Incentive coding is to be entered as:

- 10 = either skilled or unskilled labor
- 20 = both skilled and unskilled labor
- 30 = semiskilled labor only
- 40 = semiskilled and one other type of labor
- 50 = all three types of labor

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. Even with a system 2000 computer, the results can be made available in like manner with a terminal available in the classroom. 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, promotion, 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.

The cumulative effects (promotion, training, capacity, research and incentive wages) are carried over from period to period by the data entries in 361 through 368. These data appear as Matrix 3 in the printout. After being corrected for training (as explained below) they can be immediately entered for the next period of play.

They can be entered exactly as they appear in the printout (e.g., "364 DATA 70, 1, 12, 525, 80, 3, 235). If play is concluded after the current period, to be resumed at a later date, these entries should be made at that time.

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. For example, if team 4 had initiated a training program last period, and this period's data line had been "364 DATA 70,1,12,525,80,2.35", it should be reentered as "364 DATA 70,1,12,525,80,9.235". If no training changes were made last period, the line can be entered just as it appears.

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. The first matrix printed out will simply show the team decisions for the referee's verification, the second will give team results and the third the data for entry in the next period play. Summary figures for total potential sales and total promotion expenditure are given between the second and third matrices.

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							

SAMPLE 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), $					

INITIAL CONDITIONS

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)				

INITIAL CONDITIONS

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)				

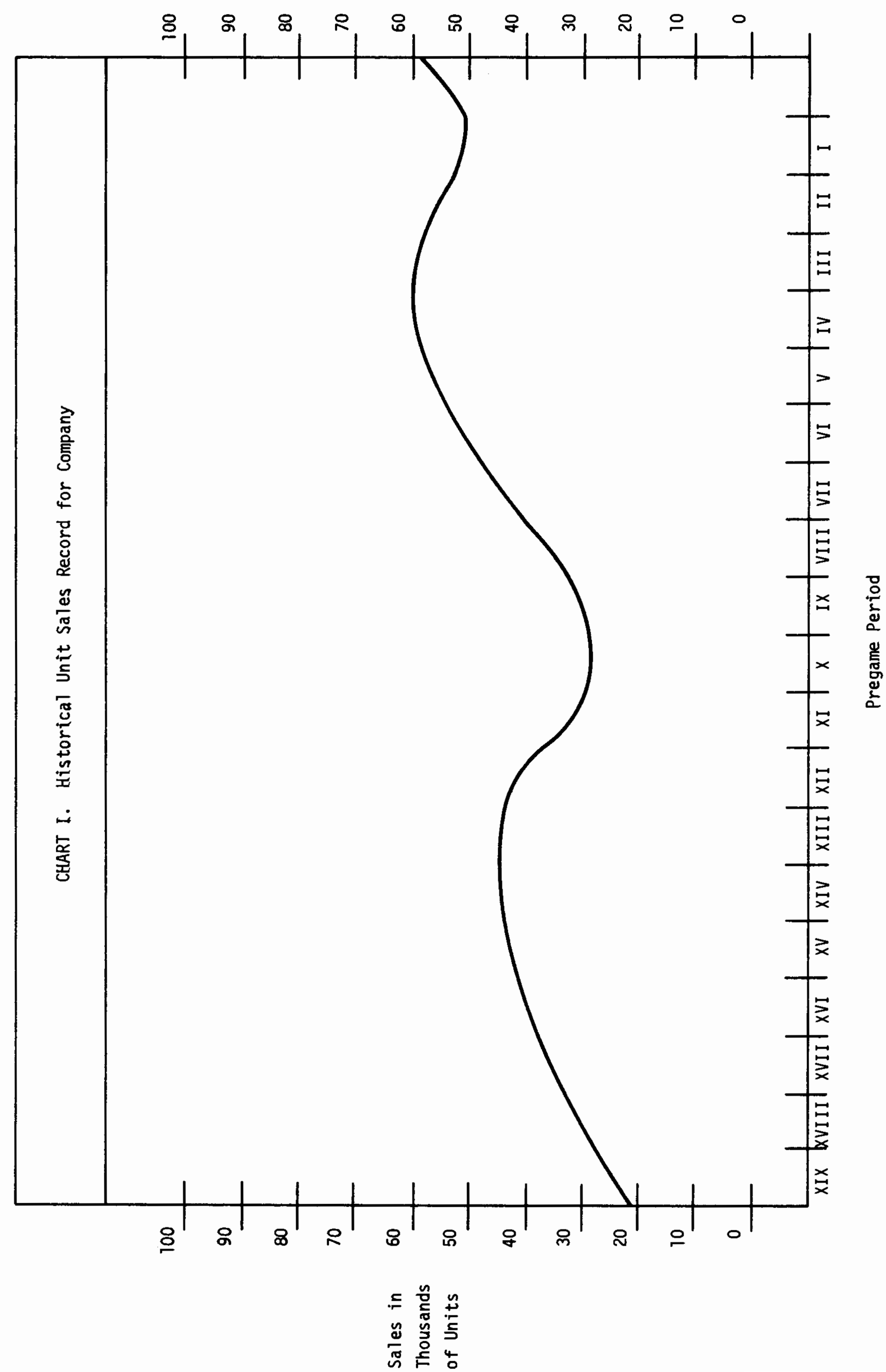
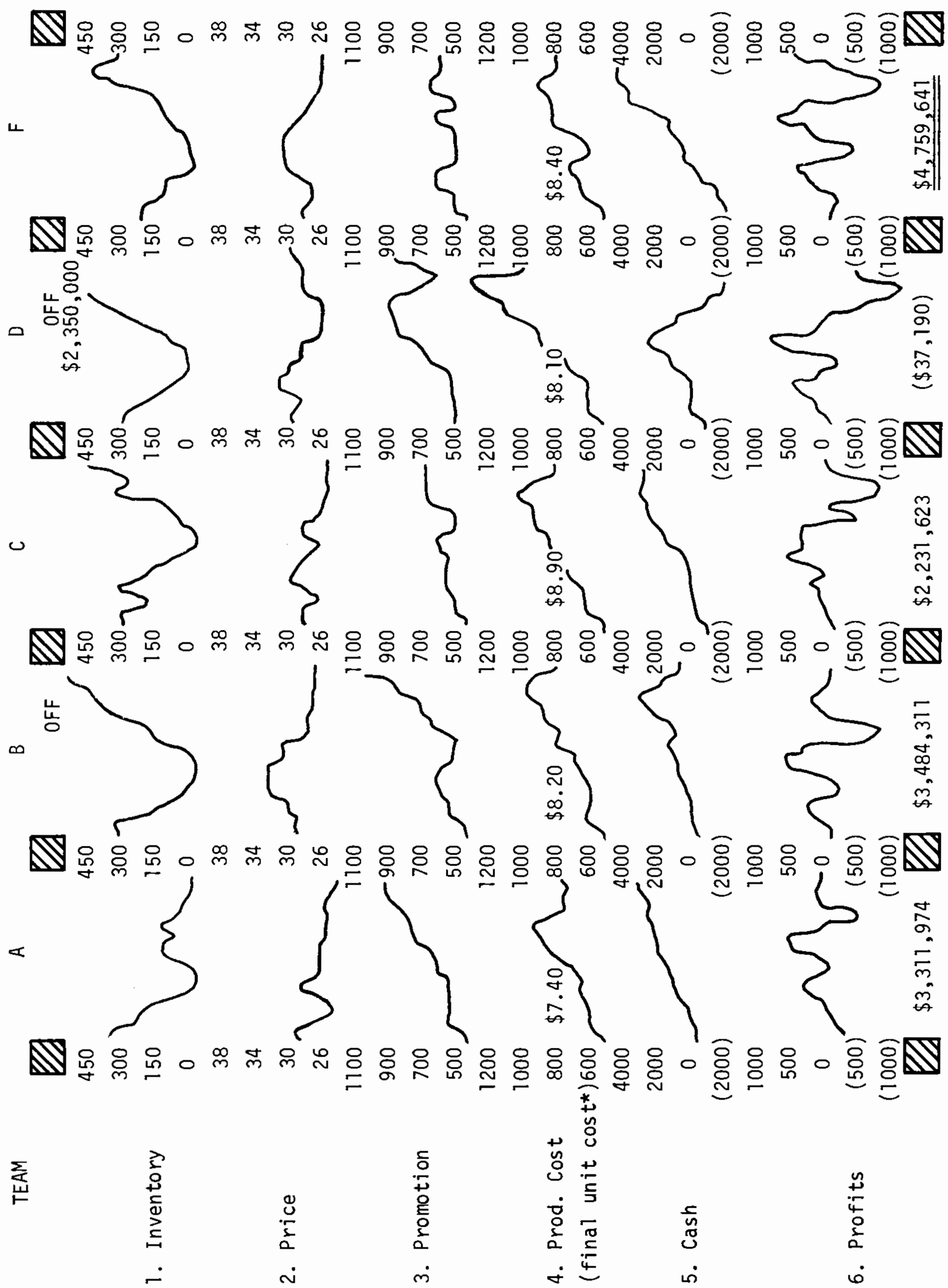


CHART I. Historical Unit Sales Record for Company
100
90
80
70
60
50
40
30
20
10
0
I
II
III
IV
V
VI
VII
VIII
IX
X
XI
XII
XIII
XIV
XV
XVI
XVII
XVIII
XIX
Pregame Period
Sales in
Thousands
of Units

CHART II FINAL ANALYSIS SHEET



```
RUN

RUN
DECSN

NO.  PRICE  PROMOTION   PRODUCTION  CAPACITY   RESEARCH   INCENTIVE

 1     30    500         65           75         100         30
 2     30    850         100          80         130         20
 3     31    500         55           75         30          10
 4     29    650         90           80         10          0
 5     30    500         65           75         80          30
 6     30    550         90           80         100         30
 7     28    450         70           80         50          20
 8     27    700         90           86         60          10

TEAM NO.        SALES          PROD COST       ADMIN COST      UNIT COST

 1               90011.1        640.25          700             9.85
 2               105377.        1354.37         735             13.5438
 3               81906.7        550             700             10
 4               94503.2        1105            735             12.2778
 5               82772.2        650             700             10
 6               89066.8        954.465         735             10.6052
 7               86677.1        700             735             10
 8               107639.        987.955         777             10.9773

TOTAL POTENTIAL SALES,      737953.     UNITS
TOTAL PROMOTION,     $ 4700

 361  DATA 55   , .985        , 0    , 700  , 75   , 0
 362  DATA 90   , .985        , 0    , 735  , 80   , 0
 363  DATA 55   , 1     , 36   , 700  , 75   , 0
 364  DATA 70   , 1     , 12   , 735  , 80   , 0
 365  DATA 55   , 1     , 96   , 700  , 75   , 0
 366  DATA 60   , .985        , 0    , 735  , 80   , 0
 367  DATA 50   , 1     , 60   , 735  , 80   , 0
 368  DATA 75   , .985        , 0    , 777  , 86   , 0

DONE
```

IATATA
36876

TITLE: IATATA ANALYSIS OF IATA AND ATA UNITIZATION PROGRAM

DESCRIPTION:

IATATA is used in the analysis of the IATA and ATA Unitization (Container) Program. IATATA is structured so that a minimum amount of knowledge is required of the IATA and ATA Unitization Programs as established by the two airline groups. IATATA at the time of its revision (April 1974) is current in all IATA and ATA rules and regulations, with respect to each unitization program. It includes all container types in both programs.

INSTRUCTIONS:

Open two files IATAID (4 records long) and CONTNR (4 records long). Then load and run program LOADR. This program initializes the two files and after it is run it need not be saved. Next load and save IATATA.

IATA containers, IATA ID numbers 1 to 9 have several variations as follows:

P = Pallet with net only.

NSI = Non-Structural Igloo.

SI = Structural Igloo.

Therefore, to access the proper version of each container, indicate the full number such as 3NSI. The author designated IATA 8* so the difference could be shown between LD-1 and LD-3.

The following sample problems reflect some IATA and ATA container shipments. The shipper owned container comparisons do not reflect in the non-IATA or ATA shipments the costs of the equivalent cardboard container. The current revision now handles the application of two pivot weights and two over pivot rates for IATA ULD's utilized on Atlantic. Sample IATA problems 2, 3, and 6 reflect the new questions required with appropriate responses. Some IATA problems supplied courtesy of Mike Baumann, Manager Cargo Services Training, Pan American World Airlines and some ATA problems courtesy of Les Milligan, Area Manager Cargo Sales, Trans World Airlines.

SAMPLE PROBLEMS

It has been suggested that shipper owned containers may carry little or no cost value because of the fact they have been used many times. The author assumed in all cases the shipment was one direction only and the full cost of the container is included in the analysis.

Continued on Next Page.

SPECIAL CONSIDERATIONS:

REFERENCES

IATA Unitization Program - a pamphlet issued 15 November 72 by International Air Transport Association courtesy of Pam American World Airlines.

IATA Unit Load Devices Manual, First Edition, issued by authority of Traffic Director, International Air Transport Assn. Geneva, Swtizerland.

Cargo Air Tariff - published by Air Canada, et al., Amsterdam, The Netherlands.

Airline Cargo Tariff - published by Scandinavian Airline System and Swissair.

CAB Tariff 131, issued by Airline Tariff Publishers, Inc., Agent, Washington, D.C. on behalf of the U.S. airlines.

ACKNOWLEDGEMENTS:

Jeff Johnson
HP International Commercial Services East

INSTRUCTIONS: Continued

Sample IATA Problems

1. You have 904 pounds of electronic measuring instruments (parts), Specific Commodity Rate 8550 JFK to STR which moves at $ .41/pound (in a 2200 pound consolidation). You wish to use a DSC-225 IATA registered container with actual tare of 96 pounds costing $16.44 each. Evaluate.

   See Sample RUN -- IATA Problem #1.

2. You have 1105 pounds of literature (Specific Commodity 7103W) and 572 pounds of electronic measuring instruments (8550) which will fit in an IATA 8 (LD-1) between JFK and AMA. The rate for literature is .37/lb and instruments is .59. The container rate is $644.00 for 1676 pounds pivot with a rate of .26/lb over pivot. Second pivot weight is 1934. Actual tare weight of container is 285 pounds. Evaluate.

   See Sample RUN -- IATA Problem #2.

3. You have 5926 pounds of machinery moving LAX to BKK which will fit in an IATA 5 container. The specific commodity rate per pound is $1.13 whereas the container rate is $3438.00 for the first 3638 pounds and an over pivot rate of $ .88/pound. The actual tare weight of the container is 550 pounds. Evaluate.

   See Sample RUN -- IATA Problem #3.

4. You have 300 pounds of electronic parts (Specific Commodity 8550) which fit in a DSC-221 (COS) moving in a consolidation between SFO and SIN at $ .98/pound. The actual tare is 18 pounds. Cost of the container is $5.44. Evaluate.

   See Sample RUN -- IATA Problem #4.

5. You have 660 pounds of electronic parts (Specific Commodity 8550) and 440 pounds of General Cargo (Q) which fit in a C08 shipper owned container moving in a consolidation between SFO and SIN at $ .98/pound and $1.55/pound respectively. The actual tare of the container is 93 pounds and costs $16.66. Evaluate.

   See Sample RUN -- IATA Problem #5.

6. You have 5500 pounds of Data Processing System (Specific Commodity 4316) moving in a 3P container NYC to LON. The specific commodity rate is .39. The minimum container charge is $1611 for 4410 lbs. second weight break is 5071 lbs. with the over pivot rates of $.30 and $.22 respectively. Evaluate.

   See Sample RUN -- IATA Problem #6.

Sample ATA Problems

1. You are shipping between SFO and JFK 100 pounds of printed matter (4915) at .234/pound and 22 pounds of general cargo at $ .52/pound which will fit in an E container costing $5.44 with an actual tare of 18 pounds. The general cargo rate is .2775.

   See Sample RUN -- ATA Problem #1.

2. Same shipment as in Problem #1 except the following changes:

   100 pounds of 4915 at .234/pound.
   82 pounds of General Cargo at .2854/pound.

   See Sample RUN -- ATA Problem #2.

3. The following QD container shipments are SFO to JFK with an actual tare of 13 pounds and the container costs $4.00 containing the weights shown of cast aluminum wheels (7616).

   a. Net Weight: 92
      Rate/Pound: .2853
      General Cargo Rate: .2775

   b. Net Weight: 187
      Rate/Pound: .2775
      General Cargo Rate: .2775

   See Sample RUN -- ATA Problem #3.

4. Using a B container SFO to JFK costing $50.00 with actual tare of 200 pounds containing 2000 pounds of printed matter (4915) at .2035 and 2000 pounds of cast aluminum wheels (7616) at .177.

   See Sample RUN -- ATA Problem #4.

5. An A-3 container SFO to JFK containing 10,000 pounds of cast aluminum wheels (7616) at .1635 for which the first 3200 pounds in the container costs $608.00 and the excess above is rated at .137/pound.

   See Sample RUN -- ATA Problem #5.

INSTRUCTIONS: Continued

6. An A-3 container SFO to JFK contains: 5,000 pounds 7616 at .1635/pound, 5,000 pounds of 0001 at .208/pound, and 2,000 pounds of general cargo at .223.

   See Sample RUN -- ATA Problem #6.

7. You have 4,000 pounds of floral stock (0625) at .117/pound and 3,500 pounds of grapes (0816) at .1355/pound loaded in an LD-7 moving SFO to JFK for which the base rate is $595.00 for the first 3100 pounds and the over pivot rate is .137. The actual tare of the container is 550 pounds.

   See Sample RUN -- ATA Problem #7.

RUN

```
GET-IATATA
RUN
IATATA

WOULD YOU LIKE A CONTAINER SUMMARY?YES
```

SUMMARY OF UNIT LOAD DEVICES

IATA					ATA(US)			
IATA ID	MIN CHARG WGT	TARE	EXT VOLUME	MAX GROSS WGT	ATA ID	MIN CHRG WGT	TARE	MAX GROSS WGT
1	13200	0	1280.0	25000	NONE	0	0	0
2	6297	1000	668.0	15000	NONE	0	0	0
2A	0	0	564.0	0	NONE	0	0	0
2B	0	0	463.0	0	NONE	0	0	0
3P	0	265	465.0	13300	A'S	0	0	0
3NSI	0	550	465.0	13300	A'S	0	0	0
3SI	0	550	465.0	13300	A'S	0	0	0
4P	0	230	365.0	10000	A'S	0	0	0
4NSI	0	500	365.0	10000	A'S	0	0	0
4ANSI	0	500	340.0	8000	A'S	0	0	0
5P	0	265	375.0	8300	LD-7	0	0	10200
5NSI	0	550	360.0	8300	LD-7	0	550	10200
5SI	0	550	350.0	8300	LD-9	0	685	10200
6P	0	0	265.0	5680	NONE	0	0	0
7*P	0	265	198.0	5000	NONE	0	0	0
7P	0	338	198.0	5250	NONE	0	0	0
7NSI	0	0	217.0	5250	NONE	0	0	0
8	0	280	170.0	3500	LD-1	1300	370	3500
8	0	280	162.4	3500	LD-P	1100	350	3500
8*	0	280	160.0	3500	LD-3	1100	340	3500
9P	0	280	160.0	2500	NONE	0	0	0
9NSI	0	280	167.0	2500	NONE	0	0	0
CO1	0	565	377.0	10000	A'S	0	0	0
CO2	0	480	320.0	10000	A'S	0	0	0
CO3	0	293	195.0	6686	B	1800	200	5000
CO4	0	277	184.0	6686	B	1800	200	5000
CO5	0	241	160.0	4841	B	1800	200	5000
CO6	0	225	150.0	4841	B	1800	200	5000
CO7	841	142	94.5	4050	LD-N	0	100	2400
CO8	551	93	61.9	3303	D	500	63	2000
CO9	396	45	44.4	2666	NONE	0	0	0
CO0	267	30	30.0	2666	NONE	0	0	0
COJ	565	96	63.4	3383	D	500	63	2000
COS	160	18	18.0	1691	E	130	18	500
VAR.	0	0	393.0	0	A-1	3000	0	13000
VAR.	0	0	457.5	0	A-2	3100	0	13000
VAR.	0	0	476.0	0	A-3	3200	0	13000
NONE	0	0	98.9	0	B-2	900	100	2500
NONE	0	0	12.0	0	QD	100	13	400
NONE	0	0	277.8	0	LD-5	2200	630	5000
NONE	0	0	340.0	0	LD-6	0	0	7000
NONE	0	0	256.6	0	LD-11	1800	0	7000

```
NOTE:0 IN MINIMUM CHARGEABLE WEIGHT COLUMN INDICATES PIVOT WEIGHT VARIES
BY TARIFF CONFERENCE(OR ORIGIN-DESTINATION COMBINATIONS). ZEROES IN
VARIOUS OTHER FIELDS MEANS DATA NOT AVAILABLE OR NOT APPLICABLE. COJ
AND COS CONTAINERS ARE FURTHER BROKEN DOWN INTO IATA REGISTRATION
NUMBERS FOR VARIATION OF THESE TWO CATEGORIES. IF YOU ARE GOING TO WORK
WITH THESE TYPES, MAKE SURE YOU ENTER THE CORRECT REGISTRATION NUMBER
(I.E. DSC-221).
```

SAMPLE IATA PROBLEM No. 1

```
IATA OR ATA?IATA
CONTAINER OR REGISTRATION NUMBER?DSC-225
ACTUAL TARE WEIGHT OF CONTAINER?96
HOW MANY COMMODITIES?1
WGT  1=?904
CONTAINER COST?16.44
RATE  1=?.41
ORIGIN AIRPORT?JFK
DESTINATION AIRPORT?STR



            ANALYSIS OF SHIPPING IN SHIPPER OWNED IATA
                 CONTAINERS VS. NORMAL PACKAGING
                         FROM JFK TO STR
                     USING DSC-225 CONTAINER

         GROSS COST OF SHIPMENT OF 1000 LBS      $ 410.00
         PLUS CONTAINER COST                        16.44
         LESS CONTAINER REBATE                     -16.20
         LESS TARE WEIGHT ALLOWANCE                -39.36
         NET COST OF SHIPMENT                      370.88

         COST IF NET CONTENTS ARE SHIPPED LOOSE    370.64
         COST IF SHIPPED IN NON-IATA CONTAINER
         OF EQUAL TARE WEIGHT                      410.00
```

SAMPLE IATA PROBLEM No. 2

```
MORE?YES
IATA OR ATA?IATA
CONTAINER OR REGISTRATION NUMBER?8
ELIGIBLE FOR SHIPPER OWNED CONTAINER DISCOUNT?NO
MINIMUM CHARGEABLE NET WEIGHT?1676
ACTUAL TARE WEIGHT OF CONTAINER?285
HOW MANY COMMODITIES?2
WGT  1=?1105
WGT  2=?572
RATE  1=?.31
RATE  2=?.48
ORIGIN AIRPORT?JFK
DESTINATION AIRPORT?AMS
CONTAINER CHARGE?501
RATE/LB FOR EXCESS ABOVE PIVOT?.26



            ANALYSIS OF SHIPPING IN AIRLINE OWNED IATA
              CONTAINER VS. SPECIFIC COMMODITY RATES
                         FROM JFK TO AMS
                     USING 8       CONTAINER

         MINIMUM CHARGEABLE WEIGHT OF  1676 LBS   $  501.00
         PLUS EXCESS OF                   1 LBS        0.26
         TOTAL CONTAINER COST                     $  501.26

         COST IF NET CONTENTS ARE SHIPPED LOOSE   $  617.11
         SAVINGS OR LOSS(-)                       $  115.85
```

SAMPLE IATA PROBLEM No. 3

```
MORE?YES
IATA OR ATA?IATA
CONTAINER OR REGISTRATION NUMBER?5NSI
ELIGIBLE FOR SHIPPER OWNED CONTAINER DISCOUNT?NO
MINIMUM CHARGEABLE NET WEIGHT?3638
ACTUAL TARE WEIGHT OF CONTAINER?550
HOW MANY COMMODITIES?1
WGT  1=?5926
RATE  1=?1.13
ORIGIN AIRPORT?LAX
DESTINATION AIRPORT?BKK
CONTAINER CHARGE?3438
RATE/LB FOR EXCESS ABOVE PIVOT?.88


              ANALYSIS OF SHIPPING IN AIRLINE OWNED IATA
                CONTAINER VS. SPECIFIC COMMODITY RATES
                           FROM LAX TO BKK
                        USING 5NSI    CONTAINER

         MINIMUM CHARGEABLE WEIGHT OF  3638 LBS  $ 3438.00
         PLUS EXCESS OF                2288 LBS    2013.44
         TOTAL CONTAINER COST                    $ 5451.44

         COST IF NET CONTENTS ARE SHIPPED LOOSE  $ 6696.38
         SAVINGS OR LOSS(-)                      $ 1244.94
```

SAMPLE IATA PROBLEM No. 4

```
MORE?Y
IATA OR ATA?I
CONTAINER OR REGISTRATION NUMBER?DSC-221
ACTUAL TARE WEIGHT OF CONTAINER?18
HOW MANY COMMODITIES?1
WGT  1=?300
CONTAINER COST?5.44
RATE  1=?.98
ORIGIN AIRPORT?SFP←O
DESTINATION AIRPORT?SIN


              ANALYSIS OF SHIPPING IN SHIPPER OWNED IATA
                   CONTAINERS VS. NORMAL PACKAGING
                           FROM SFO TO SIN
                        USING DSC-221 CONTAINER

         GROSS COST OF SHIPMENT OF  318 LBS     $ 311.64
         PLUS CONTAINER COST                        5.44
         LESS CONTAINER REBATE                     -4.00
         LESS TARE WEIGHT ALLOWANCE               -17.64
         NET COST OF SHIPMENT                     295.44

         COST IF NET CONTENTS ARE SHIPPED LOOSE   294.00
         COST IF SHIPPED IN NON-IATA CONTAINER
         OF EQUAL TARE WEIGHT                     311.64
```

SAMPLE IATA PROBLEM No. 5

```
MORE?Y
IATA OR ATA?I
CONTAINER OR REGISTRATION NUMBER?ØØ8
DATA NOT ON FILE.
IATA OR ATA?I
CONTAINER OR REGISTRATION NUMBER?OO8
DATA NOT ON FILE.
IATA OR ATA?I
CONTAINER OR REGISTRATION NUMBER?CO8
ELIGIBLE FOR SHIPPER OWNED CONTAINER DISCOUNT?YES
ACTUAL TARE WEIGHT OF CONTAINER?93
HOW MANY COMMODITIES?2
WGT  1=?66Ø
WGT  2=?44Ø
CONTAINER COST?16.44
RATE  1=?.98
RATE  2=?1.55
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?SIN
```

```
         ANALYSIS OF SHIPPING IN SHIPPER OWNED IATA
             CONTAINERS VS. NORMAL PACKAGING
                    FROM SFO TO SIN
                 USING CO8     CONTAINER

    GROSS COST OF SHIPMENT OF 1193 LBS      $1472.95
    PLUS CONTAINER COST                        16.44
    LESS CONTAINER REBATE                     -15.8Ø
    LESS TARE WEIGHT ALLOWANCE               -144.15
    NET COST OF SHIPMENT                     1329.44

    COST IF NET CONTENTS ARE SHIPPED LOOSE   1328.8Ø
    COST IF SHIPPED IN NON-IATA CONTAINER
    OF EQUAL TARE WEIGHT                     1419.94
```

SAMPLE IATA PROBLEM No. 6

```
IATA OR ATA?IATA
CONTAINER OR REGISTRATION NUMBER?8
ELIGIBLE FOR SHIPPER OWNED CONTAINER DISCOUNT?NO
MINIMUM CHARGEABLE NET WEIGHT?1676
ACTUAL TARE WEIGHT OF CONTAINER?285
HOW MANY COMMODITIES?2
WGT  1=?11Ø5
WGT  2=?572
RATE  1=?.37
RATE  2=?.59
ORIGIN AIRPORT?JFK
DESTINATION AIRPORT?AMS
CONTAINER CHARGE?644
SECOND PIVOT WEIGHT?1934
FIRST OVER PIVOT RATE=?.33
```

```
         ANALYSIS OF SHIPPING IN AIRLINE OWNED IATA
           CONTAINER VS. SPECIFIC COMMODITY RATES
                    FROM JFK TO AMS
                  USING 8       CONTAINER

    MINIMUM CHARGEABLE WEIGHT OF  1676 LBS  $  644.ØØ
    PLUS EXCESS OF                   1 LBS       Ø.33
    TOTAL CONTAINER COST                    $  644.33

    COST IF NET CONTENTS ARE SHIPPED LOOSE  $  746.33
    SAVINGS OR LOSS(-)                      $  1Ø2.ØØ
```

```
MORE?Y
IATA OR ATA?IATA
CONTAINER OR REGISTRATION NUMBER?3P
ELIGIBLE FOR SHIPPER OWNED CONTAINER DISCOUNT?NO
MINIMUM CHARGEABLE NET WEIGHT?4410
ACTUAL TARE WEIGHT OF CONTAINER?265
HOW MANY COMMODITIES?1
WGT  1=?5500
RATE  1=?.39
ORIGIN AIRPORT?NYC
DESTINATION AIRPORT?LON
CONTAINER CHARGE?1611
SECOND PIVOT WEIGHT?5071
FIRST OVER PIVOT RATE=?.30
SECOND OVER PIVOT RATE=?.22


             ANALYSIS OF SHIPPING IN AIRLINE OWNED IATA
               CONTAINER VS. SPECIFIC COMMODITY RATES
                          FROM NYC TO LON
                       USING 3P      CONTAINER

          MINIMUM CHARGEABLE WEIGHT OF  4410 LBS  $ 1611.00
          PLUS EXCESS OF                1090 LBS     292.68
          TOTAL CONTAINER COST                    $ 1903.68

          COST IF NET CONTENTS ARE SHIPPED LOOSE  $ 2145.00
          SAVINGS OR LOSS(-)                      $  241.32


MORE?N

DONE
```

SAMPLE ATA PROBLEM No. 1

```
MORE?Y
IATA OR ATA?ATA
CONTAINER NUMBER?E
ACTUAL TARE WEIGHT OF CONTAINER?18
HOW MANY COMMODITIES?2
WGT  1=?100
WGT  2=?22
CONTAINER COST?5.44
RATE  1=?.234
RATE  2=?.52
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
GENERAL CARGO RATE?.2775


             ANALYSIS OF SHIPPING IN SHIPPER OWNED ATA
                 CONTAINERS VS. NORMAL PACKAGING
                         FROM SFO TO JFK
                      USING E       CONTAINER

          GROSS COST OF SHIPMENT OF  130 LBS     $  36.07
          PLUS CONTAINER COST                        5.44
          NET COST OF SHIPMENT                      41.51

          COST IF NET CONTENTS ARE SHIPPED LOOSE    34.84
          COST IF SHIPPED IN NON-ATA  CONTAINER
          OF EQUAL TARE WEIGHT                      39.05
```

SAMPLE ATA PROBLEM No. 2

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?E
ACTUAL TARE WEIGHT OF CONTAINER?18
HOW MANY COMMODITIES?2
WGT  1=?100
WGT  2=?82
CONTAINER COST?5.44
RATE  1=?.234
RATE  2=?.2854
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
GENERAL CARGO RATE?.2775


          ANALYSIS OF SHIPPING IN SHIPPER OWNED ATA
               CONTAINERS VS. NORMAL PACKAGING
                       FROM SFO TO JFK
                    USING E       CONTAINER

     GROSS COST OF SHIPMENT OF  164 LBS     $  45.51
     PLUS CONTAINER COST                        5.44
     NET COST OF SHIPMENT                      50.95

     COST IF NET CONTENTS ARE SHIPPED LOOSE    46.80
     COST IF SHIPPED IN NON-ATA  CONTAINER
     OF EQUAL TARE WEIGHT                      51.01
```

SAMPLE ATA PROBLEM No. 3

```
MORE?Y
IATA OR ATA?ATA
CONTAINER NUMBER?OD
DATA NOT ON FILE.
IATA OR ATA?ATA
CONTAINER NUMBER?QD
ACTUAL TARE WEIGHT OF CONTAINER?13
HOW MANY COMMODITIES?1
WGT  1=?92
CONTAINER COST?4
RATE  1=?.2853
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
GENERAL CARGO RATE?.2775


          ANALYSIS OF SHIPPING IN SHIPPER OWNED ATA
               CONTAINERS VS. NORMAL PACKAGING
                       FROM SFO TO JFK
                    USING QD      CONTAINER

     GROSS COST OF SHIPMENT OF  100 LBS     $  27.75
     PLUS CONTAINER COST                        4.00
     NET COST OF SHIPMENT                      31.75

     COST IF NET CONTENTS ARE SHIPPED LOOSE    26.25
     COST IF SHIPPED IN NON-ATA  CONTAINER
     OF EQUAL TARE WEIGHT                      29.96
```

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?QD
ACTUAL TARE WEIGHT OF CONTAINER?13
HOW MANY COMMODITIES?1
WGT  1=?187
CONTAINER COST?4
RATE  1=?.2775
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
GENERAL CARGO RATE?.2775


               ANALYSIS OF SHIPPING IN SHIPPER OWNED ATA
                    CONTAINERS VS. NORMAL PACKAGING
                            FROM SFO TO JFK
                         USING QD      CONTAINER

          GROSS COST OF SHIPMENT OF  169 LBS      $  46.90
          PLUS CONTAINER COST                         4.00
          NET COST OF SHIPMENT                       50.90

          COST IF NET CONTENTS ARE SHIPPED LOOSE     51.89
          COST IF SHIPPED IN NON-ATA  CONTAINER
          OF EQUAL TARE WEIGHT                       55.50
```

SAMPLE ATA PROBLEM No. 4

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?B
ACTUAL TARE WEIGHT OF CONTAINER?200
HOW MANY COMMODITIES?2
WGT  1=?2000
WGT  2=?2000
CONTAINER COST?50
USING THE MIXED SHIPMENT RULE (RULE 12,C.A.B.131),YOU MAY INPUT RATES
EQUAL TO  4000 LBS FOR EACH COMMODITY IN THE CONTAINER.
RATE  1=?,←.2035
RATE  2=?.177
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
DAYLIGHT OR REGULAR?REGULAR
CONTAINER CHARGE?312
RATE/LB FOR EXCESS ABOVE PIVOT?.137


               ANALYSIS OF SHIPPING IN SHIPPER OWNED ATA
                  CONTAINER VS. SPECIFIC COMMODITY RATES
                            FROM SFO TO JFK
                         USING B       CONTAINER

          MINIMUM CHARGEABLE WEIGHT OF  1800 LBS  $  312.00
          PLUS EXCESS OF                2200 LBS     301.40
          PLUS CONTAINER COST                         50.00
          TOTAL CONTAINER COST                    $  663.40

          MIXED SHIPMENT RULE COST                $  781.00
          SAVINGS OR LOSS(-)                      $  117.60
```

SAMPLE ATA PROBLEM No. 5

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?A-3
ACTUAL TARE WEIGHT OF CONTAINER?550
HOW MANY COMMODITIES?1
WGT  1=?10000
RATE  1=?.1635
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
DAYLIGHT OR REGULAR?REGULAR
CONTAINER CHARGE?608
RATE/LB FOR EXCESS ABOVE PIVOT?.137



              ANALYSIS OF SHIPPING IN AIRLINE OWNED ATA
                CONTAINER VS. SPECIFIC COMMODITY RATES
                           FROM SFO TO JFK
                        USING A-3     CONTAINER

         MINIMUM CHARGEABLE WEIGHT OF  3200 LBS  $  608.00
         PLUS EXCESS OF                6800 LBS     931.60
         TOTAL CONTAINER COST                    $ 1539.60

         COST IF NET CONTENTS ARE SHIPPED LOOSE  $ 1635.00
         SAVINGS OR LOSS(-)                      $   95.40
```

SAMPLE ATA PROBLEM NO. 6

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?A-3
ACTUAL TARE WEIGHT OF CONTAINER?550
HOW MANY COMMODITIES?3
WGT  1=?5000
WGT  2=?5000
WGT  3=?2000
USING THE MIXED SHIPMENT RULE (RULE 12,C.A.B.131),YOU MAY INPUT RATES
EQUAL TO 12000 LBS FOR EACH COMMODITY IN THE CONTAINER.
RATE  1=?.1635
RATE  2=?.208
RATE  3=?.223
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
DAYLIGHT OR REGULAR?REGULAR
CONTAINER CHARGE?608
RATE/LB FOR EXCESS ABOVE PIVOT?.137



              ANALYSIS OF SHIPPING IN AIRLINE OWNED ATA
                CONTAINER VS. SPECIFIC COMMODITY RATES
                           FROM SFO TO JFK
                        USING A-3     CONTAINER

         MINIMUM CHARGEABLE WEIGHT OF  3200 LBS  $  608.00
         PLUS EXCESS OF                8800 LBS    1205.60
         TOTAL CONTAINER COST                    $ 1813.60

         MIXED SHIPMENT RULE COST                $ 2183.50
         SAVINGS OR LOSS(-)                      $  369.90
```

SAMPLE ATA PROBLEM NO. 7

```
MORE?Y
IATA OR ATA?A
CONTAINER NUMBER?LD-7
MINIMUM CHARGEABLE NET WEIGHT?3100
ACTUAL TARE WEIGHT OF CONTAINER?550
HOW MANY COMMODITIES?2
WGT  1=?4000
WGT  2=?3500
USING THE MIXED SHIPMENT RULE (RULE 12,C.A.B.131),YOU MAY INPUT RATES
EQUAL TO  7500 LBS FOR EACH COMMODITY IN THE CONTAINER.
RATE  1=?.117
RATE  2=?.1355
ORIGIN AIRPORT?SFO
DESTINATION AIRPORT?JFK
DAYLIGHT OR REGULAR?REGULAR
CONTAINER CHARGE?595
RATE/LB FOR EXCESS ABOVE PIVOT?.137


              ANALYSIS OF SHIPPING IN AIRLINE OWNED ATA
                CONTAINER VS. SPECIFIC COMMODITY RATES
                          FROM SFO TO JFK
                       USING LD-7    CONTAINER

         MINIMUM CHARGEABLE WEIGHT OF  3100 LBS  $  595.00
         PLUS EXCESS OF                4400 LBS     602.80
         TOTAL CONTAINER COST                    $ 1197.80

         MIXED SHIPMENT RULE COST                $  867.25
         SAVINGS OR LOSS(-)                      $ -330.55


MORE?N

DONE
```

This is the only ATA example where the mixed shipment rule would definitely be cheaper to rate the shipment than the straight container ruling method.

INZOUT
36088

TITLE: INPUT/OUTPUT ANALYSIS ON ECONOMIC FLOWS

DESCRIPTION:

A hypothetical economy is divided into a certain number of industries, and is analyzed as to the inter-industry flows of goods and services over a period of time. The analysis is based upon the data for a past period, and can be used to predict future flows under different conditions of consumer demand.

INSTRUCTIONS:

In this example, the economy is divided into three industries. The number of industries, and their respective names, can be altered by changing the dim-statements and output routines. With the exception of these routines, the program is general, and will accept data for any 'M' number of industries.

Data is read in the following order:

```
The flows from:  Industry #1 to Industry #1
                 Industry #1 to Industry #2
                 Industry #1 to Industry #3, etc. to #M
                 Industry #1 to the Final Consumer
                 Industry #2 to Industry #1
                 Industry #2 to Industry #2
                 Industry #2 to Industry #3, etc. to #M
                 Industry #2 to the Final Consumer
                 Industry #3 to Industry #1, etc. to #M
                 Industry #3 to the Final Consumer
                 ...
                 Industry #M to the Final Consumer
```

This is followed by a revised forecast of consumer demand from Industry #1, Industry #2, Industry #3,...,Industry #M

In this example:

```
                 Industry #1 is Agriculture
                 Industry #2 is Industry
                 Industry #3 is Service
```

SPECIAL CONSIDERATIONS:

INZOUT is restricted as written, to 3 industries. To increase this number, change dimensions in lines 9200, 9205 and 9210. A,B, and X must be M x M. T, V, and C must be M, and D must be 2M + 1. Also change the output routine.

ACKNOWLEDGEMENTS:

RUN

```
9900  DATA 25,12,8,75
9901  DATA 15,75,65,99
9902  DATA 10,51,88,34
9903  DATA 82,85,40
9999  END

RUN
INZOUT

*  INPUT/OUTPUT ANALYSIS  *

**********************************************************************

INITIAL INPUT/OUTPUT TABLE:
--------------------------


                              TO SECTOR
 FROM SECTOR                                        CONSUMERS TOTAL
              AGRICULTURE    INDUSTRY    SERVICES

AGRICULTURE       25            12          8          75        120
INDUSTRY          15            75          65         99        254
SERVICES          10            51          88         34        183

SECTOR INCOME     70            116         22         208

  TOTAL           120           254         183                  557


----------------------------------------------------------------------

REVISED INPUT/OUTPUT TABLE NUMBER 1     :
-----------------------------------


                              TO SECTOR
 FROM SECTOR                                        CONSUMERS TOTAL
              AGRICULTURE    INDUSTRY    SERVICES

AGRICULTURE       26.7413      11.3027     8.31437     82     128.358
INDUSTRY          16.0448      70.642      67.5542     85     239.241
SERVICES          10.6965      48.0366     91.458      40     190.191

SECTOR INCOME     74.8757      109.26      22.8645     207

  TOTAL           128.358      239.241     190.191            557.791

**********************************************************************

DONE
```

CONTRIBUTED PROGRAM **BASIC**

SIMLAT
36888-18019

TITLE: SIMULATION OF INVESTMENT RETURNS WITH SENSITIVITY ANALYSIS

DESCRIPTION: This program simulates the effects of an investment. From data entered, the computer will randomly select (from a normal distribution) values for the sales, fixed costs, variable costs, and life of an investment. It will then report the earnings in each year of the life and the internal rate of return over the life of the project. Summaries of results are presented.

INSTRUCTIONS: To use this program, enter in data statements (beginning in line 9ØØØ) the following information:

1. The amount of the original investment
2. The mean and std. dev. of the sales
3. The mean and std. dev. of the fixed costs
4. The mean and std. dev. of the variable cost ratio
5. The mean and std. dev. of the life of the project.

SPECIAL CONSIDERATIONS: FOR INSTRUCTIONAL PURPOSES
Suitable Courses: Corp Finance, Survey of Business, Managerial Account.

Student Background required: None

Illustrates use of Simulation technique for evaluation of capitol investment opportunities under uncertainty. See instructions on program run. Also shows use of simulation to predict project earnings.

ACKNOWLEDGEMENTS: Robert C. Lake
Auburn University in Montgomery

RUN

```
RUN
SIMLAT

WANT INSTRUCTIONS?YES
THIS PROGRAM SIMULATES THE EFFECTS OF AN INVESTMENT.
FROM DATA ENTERED, THE COMPUTER WILL RANDOMLY SELECT
(FROM A NORMAL DISTRIBUTION) VALUES FOR THE SALES, FIXED
COSTS, VARIABLE COSTS, AND LIFE OF AN INVESTMENT.  IT
IT WILL THEN REPORT THE EARNINGS IN EACH YEAR OF THE
LIFE AND THE INTERNAL RATE OF RETURN OVER THE LIFE OF THE
PROJECT.  SUMMARIES OF RESULTS ARE PRESENTED.

TO USE THIS PROGRAM, ENTER IN DATA STATEMENTS (BEGINNING
IN LINE 9000) THE FOLLOWING INFORMATION: 1. THE AMOUNT
OF THE ORIGINAL INVESTMENT; 2. THE MEAN AND STD. DEV.
OF THE SALES; (3) THE MEAN AND STD. DEV. OF THE FIXED
COSTS; (4) THE MEAN AND STD. DEV. OF THE VARIABLE COST
RATIO; (5) THE MEAN AND STD. DEV. OF THE LIFE OF THE
PROJECT.

DONE


9000 DATA 90000.
9001 DATA 100000.,10000
9002 DATA 10000,1000
9003 DATA .6,.08
9004 DATA 10,2

RUN
SIMLAT

WANT INSTRUCTIONS?NO
NUMBER OF ITERATIONS DESIRED?3
DO YOU WANT DETAILS FOR EACH ITERATION?YES
DO YOU WANT SUMMARY OF EACH ITERATION?YES


                  MEAN            STD. DEV.

SALES             100000.          10000
FIXED COST        10000            1000
VAR. EXP. RATIO   .6               .08
LIFE              10               2
INVESTMENT        90000.




ITERATION 1     LIFE = 11    YEARS

                   FIXED           VARIABLE       COMPUTED
  SALES           EXPENSES        EXP. RATIO      EARNINGS
 ---------        --------        ----------      --------
 97260.5           10046.3         .493052         31077.8
 101586.           11358.9         .651964         14799.1
 111379.           10829.2         .628713         22342.5
 103241.           10030.3         .686484         14155.6
 100381.           10414.1         .583741         23188.6
 90974.7           9274.55         .632956         15935.4
 77107.            10645.          .567554         14517.8
 110703.           9012.39         .59055          28133.2
 108328.           11213.1         .512799         33382.4
 129595.           10369.3         .72792          16709.2
 119440.           9298.67         .498897         42371.1

IRR = 32.18        PERCENT
```

ITERATION 2   LIFE = 7   YEARS

SALES	FIXED EXPENSES	VARIABLE EXP. RATIO	COMPUTED EARNINGS
105208.	9862.33	.662593	12778.5
96651.9	9712.39	.536654	22213.7
98416.	9735.15	.609547	15834.5
112735.	9393.06	.560905	27251.2
93588.2	8364.04	.563352	19643.9
88735.9	10279.6	.529819	18585.2
93634.6	9394.6	.614	13891.2

IRR = 28.62   PERCENT

ITERATION 3   LIFE = 8   YEARS

SALES	FIXED EXPENSES	VARIABLE EXP. RATIO	COMPUTED EARNINGS
89304.6	10785.2	.597657	13895.9
100792.	8997.01	.631543	16890.3
103212.	10280.4	.496154	30472.7
99261.	10134.7	.486079	29627.6
102372.	10099.8	.659976	13459
108772.	9555.33	.622041	20306.
97382.4	10447.6	.565616	20603.8
98781.6	8527.84	.73496	6403.23

IRR = 29.87   PERCENT

SUMMARY OF SIMULATION RESULTS

ITERATION	LIFE	AVERAGE ANNUAL EARNINGS	INTERNAL RATE OF RETURN
1	11	23328.4	32.18
2	7	18599.8	28.62
3	8	18957.3	29.87
MEAN		20295.2	30.22
STANDARD DEVIATION		2149.81	1.47

CUMULATIVE PROBABILITY DISTRIBUTION

PROBABILITY	EARNINGS GREATER THAN	I.R.R. GREATER THAN
.9987	13845.7	25.8
.9772	15995.6	27.27
.8413	18145.4	28.75
.5	20295.2	30.22
.1587	22445.	31.7
.0228	24594.8	33.18
.0013	26744.6	34.65

WANT TO CHANGE A VARIABLE?NO

DONE

TSBILL
36888-18039

TITLE: TIMESHARE BILLING PROGRAM

DESCRIPTION:

The Timeshare Billing System consists of a set of "BASIC" programs and files designed to process customer usage data on HP 2000 Timeshare systems, generating customer invoices for system usage and miscellaneous changes on a monthly basis. The system is capable of simultaneously billing from 1 up to 9 timeshare systems. The billing system is particularly suited for grow-installations, as only one parameter in one program needs be changed to modify the number of timeshare systems billed.

There are 12 programs in this package: CHECK, COMBIN, CUS, CUSLIS, CUSORT, DRIVER, EDITR, IDCHEK, INV1, LOADER, OVERLA, USAGE.

Procedurally, the system can be split into two main sections:

1. DAILY PROCEDURES consist of taking a system "REP"ort to paper tape (for each timeshare system to be billed). Each report tape is then loaded to its own "Daily" file by means of a basic program. Another program may be run to edit the daily files if paper tape read errors occur. The daily procedure is considered completed upon the successful loading and editing (if necessary) of an error free daily file for each timeshare system to be billed.

2. MONTH END PROCEDURES begin on the last day of the current month to be billed. Immediately after each system's report tape has been generated, that system is "RESET" (RES-ALL,Ø), setting all user's time counters back to Ø. After loading to daily files, a series of basic programs is run to compile, from the daily files, a "Master Usage" file, consisting of a monthly total time and monthly average disc storage figure for each non-zero usage I.D./System. Another permanently kept file, containing customer information, is then updated to reflect the current status of all system users. After the master usage file has been created and customer information has been updated, a series of programs leading to invoices and miscellaneous reports are run.

   The month end procedures are considered completed upon successful generation of invoices and subsequent deletion of customer information no longer required.

ORDER 36888-90039 FOR COMPLETE USER INSTRUCTIONS AND DOCUMENTATION:

Complete documentation contains:

1. SYSTEM PROCEDURAL FLOWCHARTS, broken out into daily and month end procedures.
2. A USER'S GUIDE which parallels the procedural flowcharts.
3. Definitions, formats, and comments on major system FILES.
4. Program FLOWCHARTS, VARIABLE DEFINITIONS, and GOSUB EXPLANATIONS.
5. PROGRAM LISTINGS.

ACKNOWLEDGEMENTS: HEWLETT-PACKARD/BAEDP

VOLUME IV
CONTENTS

801 MATHEMATICS (EDUCATION)

VOLUME IV
CONTENTS (Continued)

810 PROGRAMMING AND COMPUTER SCIENCE (EDUCATION)

820 ENGINEERING (EDUCATION)

801 810 820 830 833 850 860 870 871 880 885

VOLUME IV
CONTENTS (Continued)

830 ECONOMICS (EDUCATION)

833 SCIENCE (EDUCATION)

833 SCIENCE (EDUCATION) Continued

NAME	TITLE	PROGRAM NUMBER
NZYMC	:ENZYMATIC REACTION RATES	36303A
ORBIT	:INTEGRATES EQUATIONS OF MOTION	36121A
ORG1	:TEACHES STRAIGHT ALKANE NOMENCLATURE AND NAMING OF BRANCHED ALKANES	36826A
ORG5	:TEACHES NOMENCLATURE OF ALKYL HALIDES AND ALCOHOLS	36888-18011
ORGCHE	:DRILL ON ORGANIC COMPOUND NOMENCLATURE	36646A
PZHYP	:SCIENTIFIC METHOD AND HYPOTHESIS	36759A
PZKEPL	:KEPLER'S 3RD LAW	36779A
PHOSYN	:PHOTOSYNTHESIS EXPERIMENT	36305A
PHOTEL	:PHOTOELECTRIC EFFECT	36317A
PHOTON	:ENERGY LEVEL PROBLEM	36318A
PHPOH	:PH, PHO, PCT, DISSOCIATION	36617A
PLANK	:A PHOTOELECTRIC SIMULATION	36350A
POLUT	:WATER POLLUTION SIMULATION	36639B
POP	:POPULATION GROWTH	36844A
PRCNT	:PERCENT COMPOSITION	36311A
PRJTL	:PROJECTILE MOTION	36631A
REFLCT	:LEAST TIME PRINCIPLE AND LIGHT	36319A
REFLEC	:COMPUTER-AUGMENTED PHYSICS TOPICS	36672A
REFRAC	:COMPUTER-AUGMENTED PHYSICS TOPICS	36669A
SLITS	:YOUNG'S DOUBLE SLIT EXPERIMENT	36351B
SNELL	:SNELL'S LAW	36320A
SPACE	:SPACECRAFT ORBITS	36321A
SPCTRA	:OPTICAL ABSORPTION SPECTRA SIMULATION, 2-SPECIES EQUILIBRIUM MIXTURES	36184A
STERL	:FLY POPULATION CONTROL	36641B
STOICH	:MASS VOLUME	36618A
TITER	:SIMULATES TITRATION OF A BASE BY AN ACID	36888-18006
TRIVIA	:COMMON NAME QUIZ FOR CHEMISTRY STUDENTS	36888-18012
USPOP	:HUMAN POPULATION PROJECTION	36802A
VFIELD	:POTENTIAL FIELD PICTURE	36322A
VLOCTY	:INSTANTANEOUS VELOCITY	36323A
WATER1	:WATER BUDGET PROBLEM CHECK	36326A
WATER2	:WATER BUDGET	36327A
WAVES	:SUM OF TWO WAVES	36324B
MCMAN	:CARDIOVASCULAR SIMULATION	36888-18002

850 FINE ARTS (EDUCATION)

KEYSIG	:GIVES MAJOR SCALES	36258A
MUSIC	:TWELVE TONE COMPOSITION PROGRAM	36888-18028
MUSIC2	:TRIAD SOLVING PROGRAM	36276A
MUSIC5	:FINDS DOMINANT SEVENTHS	36259A

801
810
820
830
833
850
860
870
871
880
885

VOLUME IV
CONTENTS (Continued)

860 SOCIAL SCIENCE (EDUCATION)

870 ENGLISH (EDUCATION)

871 FOREIGN LANGUAGES (EDUCATION)

880 BUSINESS (EDUCATION)

801
810
820
830
833
850
860
870
871
880
885

VOLUME IV
CONTENTS (Continued)

885 EDUCATIONAL ADMINISTRATION

CLEARB
36888-18025

TITLE: CWF SUBPROGRAM

DESCRIPTION:

CLEARB is one of a series of user-written subprograms that extend the capabilities of Hewlett-Packard's licensed instructional product: Course Writing Facility (CWF). For more information about CWF, users should consult the Course Writing Facility Reference Manual, part number 24383-90001.

Sometimes an author would like to fill a buffer full with one repeated character. Perhaps he wishes to clear a buffer with spaces so that he can check buffer contents without interference from previous contents, or perhaps he is building up a buffer using variable length loads. In the latter case, if the buffer has previously been filled with "end of buffer" characters (E^c) there will always be an "eob" at the end of the message in the buffer.

The CWF author can simply invoke this sub-program with a call of the form "fn /clearb".

INSTRUCTIONS:

CALLING FORMAT

fn /clearb/bn,n

bn - a buffer from b0 to b5 (default b0)

n - any character, as well as $\underline{N^c}$ (line feed), $\underline{E^c}$ (end of buffer), or $\underline{0^c}$ (carriage return) (default = spaces).

SAMPLE CALLS

fn /clearb

fn /clearb/b2,*

fn /clearb/b4

fn /clearb/b5,E^c

The last statement would cause buffer 5 to be filled with end of buffer characters (E^c) in positions 0 through 97.

COURSE FACILITIES AFFECTED

This sub-program fills positions 0 through 97 of any buffer (b0 - b5) with the specified character, and places a carriage return (0^c) and an eob (E^c) into positions 98 and 99.

SPECIAL CONSIDERATIONS:

HP 24383A, Course Writing Facility must be present in the system for this subprogram to RUN.

ACKNOWLEDGEMENTS:

Jutta Kernke
Hewlett Packard

PROGRAMMING AND COMPUTER SCIENCE (EDUCATION) (810)

CONTRIBUTED PROGRAM **BASIC**

CLEARF
36888-18026

TITLE: CWF SUBPROGRAM

DESCRIPTION:

CLEARF is one of a series of user-written sub-programs that extend the capabilities of Hewlett-Packard's licensed instructional product: Course Writing Facility (CWF). For more information about CWF, users should consult the Course Writing Facility Reference Manual, part number 24383-90001.

When a CWF author invokes this sub-program with a call of the form "fn "fn /clearf", various course facilities can be set to zero (all counters, all the switches, and/or all the parameters) or to blanks (all the buffers) on one operation.

INSTRUCTIONS:

CALLING FORMAT

fn /clearf/n

n - an integer from 1 to 7 such that

1 - counters set to Ø

2 - buffers set to blanks

3 - switches set to Ø

4 - parameters set to Ø

5 - reset counters and switches

6 - reset counters, switches and buffers

7 - reset all four

COURSE FACILITIES AFFECTED

This function resets the facilities as specified

SAMPLE CALL

fn /clearf/6

The above statement would cause all counters and switches to be reset to 0, and all buffers to be filled with blanks.

SPECIAL CONSIDERATIONS:

HP 24383A, Course Writing Facility must be present in the system for this subprogram to run.

ACKNOWLEDGEMENTS:

Jutta Kernke
Hewlett Packard

LENGTH
36888-18024

TITLE: CWF SUBPROGRAM

DESCRIPTION:

LENGTH is one of a series of user-written sub-programs that extend the capabilities of Hewlett-Packard's licensed instructional product: Course Writing Facility (CWF). For more information about CWF, users should consult the Course Writing Facility Reference Manual, part number 24383-90001.

When an author wishes to find the position of the first "end of buffer" character (E^C) in a buffer, he can invoke this sub-program by simply calling fn /length. This function then returns the "eob" position in a counter. If no "end of buffer" character is found, 100 is returned in the counter.

INSTRUCTIONS:

CALLING FORMAT

fn /length/cn/bn

cn - a counter from c0 to c30 (no default)

bn - a buffer from b0 to b5 (default b0)

SAMPLE CALLS

fn /length/c2/b3

fn /length/c1

COURSE FACILITIES AFFECTED

The first statement would return in C2 the position of the first "end of buffer" character in B3. The second statement would return the first "eob" in B0 in C1; since no buffer is specified, the default is B0.

SPECIAL CONSIDERATIONS:

HP 24383A, Course Writing Facility must be present in the system for this subprogram to RUN.

ACKNOWLEDGEMENTS:

Jutta Kernke
Hewlett Packard

P-TUD
36888-18016

TITLE: PILOT TUTORIAL

DESCRIPTION:

This is the fourth program in the PILOT tutorial series. It is written in PILOT to teach the syntax and usage of PILOT.

This portion discusses the DEMAND and the COMPUTE statements.

INSTRUCTIONS:

This program is stored on 3 files named P-TUD1, P-TUD2, P-TUD3 for use on the HP 2000E systems and is stored on 1 file named P-TUD for the 2000F system.

The scratch file should be at least three records.

SPECIAL CONSIDERATIONS:

on 2000E: P-TUD1,48
P-TUD2,48
P-TUD3,48

on 2000F: P-TUD,70

ACKNOWLEDGEMENTS:

Lawrence E. Turner
Department of Physics and Computer Science
Pacific Union College

RUN

```
OPE-SCR,5
GET-$PILOTF
RUN
PILOTF

NAME OF PILOT PROGRAM ?$P-TUD
NAME OF SCRATCH FILE  ?SCR


?RUN

                                PILOT


HI!  I AM READY TO GO AGAIN, HOW ABOUT YOU??

CONSIDER THIS SECTION OF A PROGRAM:

   NAME TWO STATES BORDERING ON CALIFORNIA ?OREGON,ARIZONA
   VERY GOOD!

THIS PROGRAM TAKES A TOTAL OF 8 STATEMENTS (INCLUDING A NEGATIVE RETORT
WHICH YOU DID NOT GET SINCE YOU ANSWERED CORRECTLY).  IT ALSO TAKES
INTO ACCOUNT ANY COMBINATION OF THE THREE POSSIBLE STATES!

WHAT STATEMENT IS NECESSARY FOR THIS ?DEMAND
EXCELLENT, I CAN TELL YOU DID YOUR HOMEWORK!
LET'S LOOK AT A LISTING OF THE PROGRAM.

        7000 T:NAME TWO STATES BORDERING ON CALIFORNIA +
        7010 A:
        7020 M:ARIZON,AZ
        7030 M:NEVAD,NV
        7040 M:OREG,OR
        7050 .D:2
        7060 Y:VERY GOOD!
        7070 N:NOPE.

THE HEART OF THIS IS STATEMENT 7050, THE DEMAND STATEMENT.
WHAT DO STATEMENTS 7020, 7030, AND 7040 DO ?MATCH
YES, THEY CHECK FOR THE POSSIBLE CORRECT STATES.  NOW IF STATEMENT 7050
WERE LEFT OUT, THEN WHAT RESPONSE WOULD SET THE MATCH FLAG TO THE 'YES'
STATE ?R-OREGON
EXACTLY!  ONLY 'OREGON' WOULD INITIATE 'VERY GOOD!'.  WHAT THE DEMAND
STATEMENT DOES IS TO CHANGE THE MATCH FLAG TO 'YES' IF TWO OR MORE OF
THE EXECUTED MATCH STATEMENTS SINCE THE LAST EXECUTED ANSWER STATEMENT
HAVE BEEN SUCCESSFUL, OTHERWISE IT IS SET TO 'NO'!
WHAT OTHER PILOT STATEMENTS CAN CHANGE THE MATCH FLAG ?ANSWER
CORRECT, BUT YOU FORGOT ONE.  BOTH THE ANSWER AND THE MATCH CAN ALSO
SET THE MATCH FLAG.
IF STATEMENT 7050 WERE TO READ '7050 .D:1', THEN HOW MANY CORRECT
STATES WOULD HAVE TO BE ENTERED IN ORDER TO GET 'VERY GOOD!' ?1
EXACTLY!  THE INTEGER OBJECT OF THE DEMAND STATEMENT (IN THIS CASE 1)
TELLS HOW MANY EXECUTED MATCH STATEMENTS MUST BE SUCCESSFUL SO THAT THE
'Y' CONDITION STATEMENTS WILL BE EXECUTED.
CONSIDER:

        7050 T:YOU GOT +
        7060 .D:1
        7070 Y:ONE+
        7080 .D:2
        7090 Y:, TWO+
        7100 .D:3
        7110 Y:, THREE+
        7120 .D:1
        7130 Y: OF THEM!
        7140 N:NOT EVEN ONE!
```

```
TRY IT!

   NAME TWO STATES BORDERING ON CALIFORNIA ?OREGON,ARIZONA
   YOU GOT ONE, TWO OF THEM!

TRY IT AGAIN!

   NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA
   YOU GOT ONE, TWO OF THEM!

TRY IT AGAIN!

   NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA,OREGON
   YOU GOT ONE, TWO, THREE OF THEM!

ONE MORE TIME.

   NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA,OREGON
   YOU GOT ONE, TWO, THREE OF THEM!

THERE ARE, OF COURSE, MANY VARIATIONS MADE POSSIBLE BY THE DEMAND
STATEMENT.  CONSIDER:

        7050 .D:3
        7060 Y:VERY, VERY GOOD!  YOU GOT ALL THREE!
        7070 JY:*NEXT
        7080 .D:2
        7090 Y:CORRECT!
        7100 JY:*NEXT
        7110 .D:1
        7120 Y:YOU DID GET ONE RIGHT.
        7130 N:YOU DIDN'T EVEN GET ONE!
        7140 *NEXT R:

WHY DO YOU THINK THE JUMP STATEMENTS ARE NECESSARY??
?

IF .D:3 IS SATISFIED, SO IS .D:2 AND .D:1!!  THUS IF THREE ARE FOUND,
THEN YOU MUST BRANCH AROUND THE SECTIONS THAT CHECK FOR TWO AND ONE.
NOTE THAT THE DEMAND STATEMENT CHECKS FOR 'AT LEAST'.  IT DOES NOT
CHECK FOR 'EXACTLY'.
OK, WHAT DO YOU THINK .D:0 SETS THE MATCH FLAG TO ?
??YES
RIGHT ON!  SINCE IN ALL CASES AT LEAST ZERO MATCH STATEMENTS HAVE BEEN
SATISFIED, .D:0 ALWAYS SETS THE MATCH FLAG TO 'YES'.  THIS IS A WAY OF
FORCING THE MATCH FLAG IRREGARDLESS OF PREVIOUS MATCHES.
AN EXAMPLE:

        3560 T:CAN YOU NAME THE CLOSEST STAR TO THE EARTH +
        3570 A:
        3580 M:YES,OF COURS,RIGHT,CERTAIN,SURE,Y
        3590 Y:OK, WHAT IS IT +
        3600 *STAR AY:
        3610 M:SUN,SOL
        3620 Y:VERY GOOD!
        3630 JY:*NEXT
        3640 T:NO WAY, HERE'S A HINT:  YOU ONLY SEE IT DURING THE DAY.
        3650 T:NOW TRY IT +
        3660 .D:0
        3670 J:*STAR

IF STATEMENT 3660 WERE MISSING, WOULD *STAR AY: BE EXECUTED AFTER THE
JUMP (J:*STAR) ?NO
THAT'S RIGHT!  IN FACT, THERE WOULD BE NO WAY TO GET OUT OF THE LOOP!!
ACTUALLY THERE ARE PERHAPS OTHER WAYS OF ACHIEVING THIS SAME RESULT
WITHOUT RESORTING TO THE .D:0, BUT AT TIMES IT IS CONVENIENT.
THERE IS ONE FURTHER POINT TO CONSIDER.  IF THERE ARE TOO MANY
ALTERNATIVES IN A MATCH OBJECT TO FIT ON ONE LINE, THEN HOW DO YOU
'CONTINUE' A MATCH??
?CONDITION
ACTUALLY, BY THE USE OF AN 'N' CONDITION YOU CAN 'CONTINUE' AS LONG AS
YOU WISH.
THE QUESTION TO BE ANSWERED IS:  DOES THIS AFFECT THE FUNCTION OF THE
DEMAND STATEMENT?
WHAT DO YOU THINK ?NO
RIGHT!  IT DOES WORK PROPERLY IF YOU THINK OF THE TWO OR MORE
STATEMENTS 'CONTINUED' TOGETHER AS A SINGLE MATCH STATEMENT.  AT THE
MOST THERE WILL BE ONE SUCCESSFUL MATCH AMONG THEM.  TO SEE THIS, EXAM:
```

```
4070 T:NAME A DIGIT AND A COLOR.
4080 A:
4090 M:ZER,ONE,TWO,THRE,FOUR,FIVE
4100 MN:SIX,SEVEN,EIGHT,NINE
4110 M:BLUE,RED,GREEN,ORANGE,GREY
4120 MN:PINK,BLACK,PURPLE,VIOLET
4130 MN:YELLOW,WHITE,BROWN
4140 .D:2
4150 Y:GROOVY!
4160 N:I GIVE UP!
```

IF 'TWO' IS TYPED IN, HOW MANY SUCCESSFUL MATCH STATEMENTS WILL
EXECUTED BEFORE 4140 ?1
CORRECT. ONLY NUMBER 4090 WILL BE SUCCESSFUL, IN FACT 4100 WILL NEVER
BE EXECUTED!
NOW WHAT IF 'NINE AND RED' ?2
RIGHT. BOTH 4100 AND 4110 ARE SUCCESSFUL. NUMBER 4090 IS EXECUTED BUT
FAILS. 4120 AND 4130 ARE NOT EVEN EXECUTED.
IF 'NOTHING' IS TYPED IN ?0
OK, THEY ALL WILL BE EXECUTED, BUT NONE SUCCESSFULLY.
NOTICE THAT EACH SET OF 'CONTINUED' MATCH STATEMENTS ACTS LIKE ONLY ONE
STATEMENT AS FAR AS BOTH THE MATCH FLAG AND THE DEMAND STATEMENT.

WHAT DO YOU THINK IS THE MAXIMUM NUMBER OF SUCCESSFUL MATCHES THAT CAN
BE 'DEMANDED' ?99
I GUESS YOU REALLY DID YOUR HOMEWORK! THE LARGEST ALLOWED OBJECT OF A
DEMAND STATEMENT IS 99. THUS YOU CAN HAVE UP TO 99 MATCHES TO BE
'DEMANDED'! IT IS RARE THAT YOU WILL EVER NEED THIS MANY FOR A VERY
SIMPLE REASON.
WHAT IS THE LONGEST RESPONSE ALLOWED BY PILOT ?72
CLOSE! EXACTLY! UP TO 72 CHARACTERS MAY BE TYPED IN AT ANY TIME.
THIS IS THE MOST THAT PILOT CAN CHECK FOR POSSIBLE MATCHES AT ANY ONE
TIME.

THAT COMPLETES EVERYTHING THERE IS TO KNOW ABOUT DEMAND! YOU NOW KNOW
(OR HAVE BEEN EXPOSED TO) MOST OF THE PILOT LANGUAGE STATEMENTS. YOU
CAN CAUSE THE COMPUTER TO PRINT OUR STUFF, ASK QUESTIONS, MATCH THE
RESPONSE, MAKE APPROPRIATE COMMENTS, JUMP, AND NOW DEMAND. THERE ARE
ONLY THREE MORE THINGS TO KNOW AND YOU WILL BE AN EXPERT! THESE ARE:
HOW TO MODIFY THE VALUE OF A NUMERIC VARIABLE, SUBROUTINE JUMPING, AND
A MORE POWERFUL MATCH STATEMENT.

LET'S REVIEW WHAT YOU KNOW ABOUT NUMERIC VARIABLES. FIRST HOW MANY
DIFFERENT NUMERIC VARIABLES ARE THERE ?26
PRECISELY!
WHAT PILOT STATEMENT(S) CAN MODIFY THE VALUE STORED IN A NUMERIC
VARIABLE ?DONT KNOW
SO FAR WE HAVE ONLY DISCUSSED HOW THE ANSWER STATEMENT CAN MODIFY THE
VALUE OF A VARIABLE.
WHAT PART OF AN ANSWER STATEMENT ALLOWS THIS ?AFTER THE COLON
BEAUTIFUL! AND WHAT SPECIAL SYMBOL SIGNIFIES A NUMERIC VARIABLE ?#
OF COURSE!
NAME TWO WAYS IN WHICH A NUMERIC VARIABLE MAY BE USED.
?MATCH AND TYPE
WELL, YOU GOT ONE. THE FIRST WAY WE STUDIED WAS TO OUTPUT THE VALUE IN
A TYPE STATEMENT. THE SECOND WAS IN A CONDITION. HERE IS A PART OF A
PROGRAM TO ILLUSTRATE:

```
4520 T:WHAT IS YOUR AGE +
4530 *PLEASE A:#A
4540 TB:GIMME A NUMBER!
4550 JB:*PLEASE
4560 T(A<=0):WHAT?  A NEGATIVE NUMBER!
4570 T(A>150):A FOSSIL IF I EVER SAW ONE!
4580 T:ARE YOU REALLY #A YEARS OLD???
```

WHAT DOES THE CONDITION 'B' DO ?DONT KNOW
I ALMOST FEEL LIKE GIVING UP ON YOU! THE STATEMENTS WITH A 'B'
CONDITION ARE EXECUTED ONLY IF NO VALID NUMBER WERE ENTERED. IT ALLOWS
TESTING OR CHECKING FOR A NUMBER IN THE RESPONSE.
SOMETIMES IT IS DESIRED TO MODIFY THE VALUE OF A NUMERIC VARIABLE OTHER
THAN IN AN ANSWER STATEMENT. THERE ARE THREE PILOT STATEMENTS THAT DO
THIS. THE FIRST AND MOST POWERFUL IS THE COMPUTE STATEMENT. WHAT DO
YOU THINK IS THE INSTRUCTION FOR THE COMPUTE STATEMENT ?.C
AHA, YOU ARE TOO TRICKY! THE INSTRUCTION FOR THE COMPUTE STATEMENT IS
SIMPLY A 'C'.
LET'S LOOK AT A FEW EXAMPLE COMPUTE STATEMENTS:

```
7350 C:A=5
8120 CY:B=A+32*X
8320 C(Z>3):S=T*(2+N)
9040 C:H=K
9450 C:N=N+1
```

```
THE OBJECT OF THE COMPUTE STATEMENT BEGINS WITH WHAT ?CHARACTER
OH MY ACHING HEADS!  WILL YOU WAKE UP??  THE FIRST THING IN THE COMPUTE
OBJECT IS A NUMERIC VARIABLE.  IT IS THE ONE WHOSE VALUE IS TO BE
MODIFIED.
WHAT IS THE NUMERIC VARIABLE THAT IS TO BE MODIFIED IN STATEMENT 8320
ABOVE ?S
ALRIGHT, THE VARIABLE S WILL BE MODIFIED WHEN 8320 IS EXECUTED.
WHAT IS THE NEXT ITEM IN ALL COMPUTE STATEMENT AFTER THE LEADING
NUMERIC VARIABLE ?=
GOOD.  ALL COMPUTE STATEMENTS MUST HAVE AN EQUALS SIGN (=).  THE PART
TO THE RIGHT OF THE EQUALS SIGN IS VERY GENERALLY TERMED AN ARITHMETIC
EXPRESSION OR SIMPLY EXPRESSION.

          C:<NUMERIC VARIABLE>=<EXPRESSION>

THIS IS HOW YOU WOULD DIAGRAM A GENERAL COMPUTE STATEMENT.  THE '<>'
SURROUND A SINGLE SYNTACTICAL PILOT ENTITY.  I HAVE LEFT OFF THE
STATEMENT NUMBER AND ANY OPTIONAL LABEL OR CONDITION, THEY ARE MOST
CERTAINLY ALLOWED.  WHERE DOES THE CONDITION GO??
?
          .
          .
          .
```

P-TUE
36888-18017

TITLE: PILOT TUTORIAL

DESCRIPTION:

This is the fifth and last program in the tutorial series in PILOT to teach the elements of the PILOT language.

This program deals with additional usage of the COMPUTE statement, the INTEGER FUNCTION, the RANDOM NUMBER FUNCTION, USE and END, and the extended MATCH.

INSTRUCTIONS:

This program is stored in 3 files: P-TUE1, P-TUE2, P-TUE3 for the HP 2000E system and on the one file P-TUE for the HP 2000F system.

The scratch file should be at least 2 records in length.

SPECIAL CONSIDERATIONS:

2000E: P-TUE1, 48; P-TUE2, 48; P-TUE3, 48

2000F: P-TUE, 70

ACKNOWLEDGEMENTS:

Lawrence E. Turner, Jr.
Department of Physics and Computer Science
Pacific Union College

RUN

```
OPE-SCR,5
GET-$PILOTF
RUN
PILOTF

NAME OF PILOT PROGRAM ?$P-TUE
NAME OF SCRATCH FILE  ?SCR


?RUN

                                   PILOT



GREETINGS!  THIS IS THE FINAL PROGRAM IN THIS SERIES.  YOU ARE ALMOST
FINISHED.  ACTUALLY, I HOPE THIS IS JUST A BEGINNING OF YOUR USAGE OF
PILOT!

LAST TIME WE DISCUSSED THE COMPUTE STATEMENT AND AT LEAST ONE EXAMPLE
OF HOW IT COULD BE USED.  DO YOU REMEMBER THAT USE ?YES
VERY GOOD, WHAT IS IT ?
??FORGOT
WELL, WHAT I HAD IN MIND WAS THE USE AS A COUNTER.  AS IN:

             1730 C:M=M+1
             2960 CY:P=P+1
             4110 C(X<17):D=D+1

BESIDES THE COMPUTE STATEMENT, WHAT PILOT STATEMENT CAN MODIFY THE
VALUE OF A NUMERIC VARIABLE ?ANSWER STATEMENT
RIGHT ON!  SO FAR WE HAVE ONLY EXAMINED THE ANSWER STATEMENT.  THIS
ALLOWS A QUESTION WHICH ASKS FOR A NUMERIC RESPONSE.
THE COMPUTE STATEMENT MAY BE USED IN CONJUNCTION WITH THIS TO HELP
EXTRACT THE INTENDED RESPONSE MORE ACCURATELY.  CONSIDER:

         6310 T:HOW MANY DAYS IN ONE WEEK +
         6320 A:#D
         6330 M:SEVEN
         6340 CY:D=7
         6350 T(D=7):EXACTLY!  +
         6360 T(D#7):SORRY, +
         6370 T:THERE ARE SEVEN DAYS IN ONE WEEK.

WHAT IS THE PURPOSE OF STATEMENTS 6330 AND 6340??
?TO ALLOW FOR NUMERIC 7 AND ALPHABETIC SEVEN
RIGHT, THEY CATCH THE RIGHT RESPONSE IF THE PERSON HAPPENED TO TYPE
'SEVEN' INSTEAD OF '7'.  BOTH ARE CORRECT.
LET'S EXECUTE THIS PROGRAM.

   HOW MANY DAYS IN ONE WEEK ?7
   EXACTLY!  THERE ARE SEVEN DAYS IN ONE WEEK.

NOW TRY IT AND ENTER 'SEVEN' SPELLED OUT.

   HOW MANY DAYS IN ONE WEEK ?SEVEN
   EXACTLY!  THERE ARE SEVEN DAYS IN ONE WEEK.

OFTEN ONE ONLY WISHES THAT THE RESPONSE BE CLOSE TO THE CORRECT
ANSWER.  AGAIN THE COMPUTE STATEMENT TO THE RESCUE!

         8400 T:HOW MANY CENTIMETERS IN ONE INCH +
         8410 A:#X

THE CORRECT ANSWER IS 2.54; HOWEVER, TO BE CONSIDERED CORRECT YOU MIGHT
ALLOW ANY NUMBER BETWEEN 2.5 AND 2.58.  THIS WOULD ALLOW ANY RESPONSE
THAT IS WITHIN 0.04 OF THE CORRECT ANSWER TO BE COUNTED.  A VERY SIMPLE
WAY IS TO COMPUTE THE DIFFERENCE BETWEEN THE CORRECT ANSWER AND THE
RESPONSE.
HOW WOULD YOU DO THIS ?C:D=2.54-X
I BELIEVE THAT IS CORRECT!  A COMPUTE STATEMENT THAT LOOKS LIKE THIS
SHOULD DO THE TRICK:
```

```
        842Ø C:D=2.54-X
```

IF THE RESPONSE IS: '2.5' WHAT IS STORED IN D ?.Ø4
EXACTLY! D HAS Ø.Ø4 STORED IN IT. SINCE 2.5 IS AN ALLOWED 'CORRECT ANSWER', HOW WOULD YOU CAUSE THE APPROPRIATE MESSAGE TO BE TYPED OUT??
?C:←-T(D<=Ø.Ø4):CORRECT!
GOOD. BY USING D IN A CONDITION ON A TYPE STATEMENT YOU CAN PRINT THE CORRECT RETORT. CONSIDER:

```
        84ØØ T:HOW MANY CENTIMETERS IN ONE INCH +
        841Ø A:#X
        842Ø C:D=2.54-X
        844Ø T(D<=Ø.Ø4):VERY GOOD!
        845Ø T(D>Ø.Ø4):NOT QUITE.
```

THIS IS COOL FOR GUESSES LESS THAN 2.54, BUT WHAT WOULD BE PRINTED IF '57' WERE ENTERED ?VERY GOOD!
EXACTLY! NOW IS '57' A 'CORRECT RESPONSE' ?NO
OF COURSE NOT, THUS WE HAVE A BUG TO FIX!! WHAT WE REALLY WANT IS THE ABSOLUTE DIFFERENCE FOR D. SO IF D IS LESS THAN ZERO, WE WANT TO CHANGE ITS SIGN. THIS CAN BE DONE WITH ONE PILOT STATEMENT:

```
        843Ø C(D<Ø):
```

WHAT IS THE CORRECT OBJECT ?D=-D
VERY GOOD!
THUS OUR PROGRAM BECOMES:

```
        84ØØ T:HOW MANY CENTIMETERS IN ONE INCH +
        841Ø A:#X
        842Ø C:D=2.54-X
        843Ø C(D<Ø):D=-D
        844Ø T(D<=Ø.Ø4):VERY GOOD!
        845Ø T(D>Ø.Ø4):NOT QUITE.
```

EXCEPT FOR INTEGER RESPONSES IT IS BEST TO GO THROUGH SOMETHING LIKE THIS FOR NUMERIC RESPONSES, EVEN FOR SIMPLE DECIMAL NUMBERS. THE COMPUTER CANNOT REPRESENT MOST DECIMALS EXACTLY, HENCE SOMETHING LIKE: 732Ø T(X=.1): ... MAY NEVER BE EXECUTED EVEN IF 'Ø.1' HAS BEEN PREVIOUSLY STORED IN X. SOMEWHERE, PERHAPS IN THE SEVENTH DECIMAL PLACE, X MAY DIFFER SLIGHTLY FROM Ø.1, AND THE TEST WILL FAIL.

ANOTHER USEFUL WAY TO MODIFY THE VALUE OF A NUMERIC VARIABLE IS TO REPLACE IT WITH THE GREATEST INTEGER LESS THAN OR EQUAL TO THE ORIGINAL NUMBER.
FOR 4.7, WHAT IS THIS INTEGER ?4
RIGHT ON!
HOW ABOUT Ø.247 ?Ø
BEAUTIFUL!
OK, TRY 6.ØØ ?6
GOOD.
AND -2.7 ?-3
EXCELLENT!
FINALLY, -7.ØØ ?-7
PERFECT! NOTE THAT FOR POSITIVE NUMBERS THIS GREATEST INTEGER FUNCTION IS EQUIVALENT TO JUST THROWING AWAY ANY DECIMAL PORTION; HOWEVER, THIS SIMPLE-MINDED DESCRIPTION IS NOT ADEQUATE FOR NEGATIVE NUMBERS.
THIS OPERATION IS DIFFICULT, IF NOT IMPOSSIBLE, TO DO WITH THE COMMON ARITHMETIC OPERATORS AS ALLOWED IN THE COMPUTE STATEMENT, YET IT IS SUFFICIENTLY USEFUL TO BE INCLUDED IN THE PILOT LANGUAGE.
A NEW INSTRUCTION IS USED, WHAT DO YOU THINK IT IS ?I
THAT IS A GOOD GUESS, BUT THE INTEGER FUNCTION (REALLY THE GREATEST INTEGER THAT IS LESS THAN OR EQUAL TO) IS DESIGNATED BY THE EXTENDED INSTRUCTION '.I'. EXAMPLES ARE:

```
          371Ø .I:X
          429Ø .I:S
          666Ø .I(X>9):X
```

WHAT ARE THE OBJECTS OF THESE INTEGER FUNCTION STATEMENTS ?NUMERIC VARI
NUMERIC VARIABLES
EXACTLY! THE OBJECT IS A SINGLE NUMERIC VARIABLE. ITS VALUE IMMEDIATELY BEFORE THE EXECUTION OF THE STATEMENT IS REPLACED BY THE GREATEST INTEGER THAT IS LESS THAN OR EQUAL TO IT.
TO SEE A POSSIBLE USE, CONSIDER:

```
3250 T:THREE IS WHAT PERCENT OF 24 +
3260 A:#P
3270 .I:P
3280 T(P=12):EXCELLENT!
3290 T(P#12):NOPE.
```

WHAT IS THE SMALLEST NUMBER FOR WHICH 'EXCELLENT' WILL BE PRINTED ?12
RIGHT ON!
HERE THE CORRECT ANSWER IS 12.5, BUT ANYTHING FROM 12 UP TO (BUT NOT INCLUDING) 13 IS ACCEPTABLE. THE .I VERY EASILY SETS UP THE RANGE.
A MORE IMPORTANT USE IS FOR ROUNDING. THE INTEGER FUNCTION AS GIVEN DOES NOT ROUND, IT MERELY TRUNCATES. ROUNDING CAN BE ACHIEVED IN QUITE A STRAIGHTFORWARD MANNER BY:

```
4560 C:X=X+.5
4570 .I:X
```

WHAT WILL BE STORED IN X IF IT PREVIOUSLY HAD 4.00 ?4
EXACTLY.
HOW ABOUT 4.6 ?5
COOL!
OK, TRY -2.1 ?-2
RIGHT!
AND WHAT DOES -5.7 BECOME ?-5
MY, MY, NO. -5.7 + 0.5 IS -5.2 WHICH BECOMES -6.

NOTE THAT THE WAY THE INTEGER FUNCTION IS DEFINED MAKES THE ROUNDING OPERATION WORK FOR BOTH POSITIVE AND NEGATIVE VALUES. IF IT SIMPLY DISCARDED THE DECIMAL PART, THEN THERE WOULD HAVE TO BE SPECIAL TESTS AND HASSELING TO DO THE ROUNDING.
YOU CAN ROUND TO ANY POINT YOU WISH. CONSIDER THE FOLLOWING TO ROUND TO THE NEAREST HUNDRETH (PENNY??):

```
7310 C:Q=Q*100+.5
7320 .I:Q
7330 C:Q=Q/100
```

OK, IF Q ORIGINALLY CONTAINS 3.14159, WHAT IS THE NEW VALUE ?

.
.
.

CONTRIBUTED PROGRAM **BASIC**

PILOTE
36637

TITLE: PILOT 73 AUTHOR LANGUAGE FOR HP 2000E

DESCRIPTION:

The PILOTE package contains 2 programs NAMed PILOT and PILOTE.

PILOTE is an author language for Computer Assisted Instruction (CAI). It is an implementation of PILOT 73 for a Hewlett-Packard 2000E Time-shared BASIC system. PILOT 73 is a hybridization of several author languages used in the San Francisco Bay area. Early in 1973 a group organized by Stanford Research Institute met and agreed upon a standardization called PILOT 73. These antecedents of PILOT have been used extensively for writing dialog type CAI programs.

There are several advantages of such an author language. First, the syntax is relatively simple so that within a short time most anyone can master the language. Even more importantly, the construction of the PILOT language is such as to facilitate "dialog" type programming with a minimum of effort. One is able to write effective programs quickly.

To be useful, any author language must be written as an interactive language. PILOTE is implemented in BASIC for HP 2000E time-shared system. This permits it to be used simultaneously with BASIC on this system.

INSTRUCTIONS:

P-DEMO is included here as a demonstration of the features of the PILOT language.

'PILOTE' is the syntax and entry portion of the system. The PILOT language program is stored on up to 3 files as source statements. Line numbers allow for editing (insertion, replacement, and deletion) of the PILOT language statements.

'PILOT' is the program that executes the PILOT language program.

continued on following page.

SPECIAL CONSIDERATIONS:

See PILOTF (HP 36756) for the version of PILOT 73 modified for the HP 2000F.

See P-TUA (HP 36758) and P-TUB (HP 36775A) which are tutorial programs written in PILOT to teach the syntax and usage of PILOT.

See P-HYP (HP 36759) a PILOT language program concerning the scientific method and its relation to hypothesis and observation.

Complete user instructions including classroom implementation of this program is available: HP 5951-5660 PILOT Users Manual.

For ordering information of this manual, contact:

HP Computer Curriculum Project
11000 Wolfe Road
Cupertino, California 95014

ACKNOWLEDGEMENTS: Lawrence E. Turner, Jr.
Pacific Union College

INSTRUCTIONS continued

A PILOT program consists primarily of statements and questions typed by the computer. The student's responses may be analyzed and appropriate action is then taken by the computer dependent on exactly what the student has typed. Thus a dialog between the computer the student may be established.

The syntax of PILOTE is relatively easy to master.

```
 10  R:  THIS IS A PILOT PROGRAM
 20  R:
 30  T:HELLO!  I AM YOUR FRIENDLY COMPUTER
 40  T:  WHAT IS YOUR NAME +
 50  A:
 60  T:WELL $NAME HOW DO YOU USUALLY USE A COMPUTER??
 70  A:
 80  T:
 90  T:THAT SOUNDS PRETTY GOOD!!
100  E:
```

This sample program contains four out of the thirteen possible statement types in PILOT. The first statement is a REMARK statement which is not executed, statement 30 is a TYPE statement. Everything to the right of the colon is printed on the terminal. The plus (+) as a last character in line 40 prevents the carriage return-line feed that would normally occur at the end of the line. Thus statement 50, an ANSWER statement prints a question mark (?) at the second position after the last "E" in statement 40. After printing the question mark the computer waits for the user to type in something from the terminal followed by a carriage return (ret). Whatever is typed-in is stored in the string variable NAME and is printed out in statement 60, where $NAME is replaced by whatever the student typed-in for the response to statement 50. The last statement is an END statement which terminates the program.

A main virtue of a computer is the ability to analyze a student's response and to make decisions on the basis of what the student has typed. In PILOT this is accomplished by the MATCH statement and the use of a condition.

```
10  T:WHAT IS YOUR SEX +
20  A:
30  M:FEMALE,GIRL,WOMAN,LADY
40  TY:I HOPE YOU ARE NOT A MEMBER OF WOMEN'S LIB!
50  TN:I HOPE YOU ARE NOT A CHAUVINIST PIG!
```

In this case the MATCH statement (number 30) has four possible alternatives separated by a comma (,).

Some very powerful extensions are contained in the other PILOT statements.

The more important of these are: the JUMP statement allows one to branch to any labeled statement, again depending on a condition, the DEMAND statement (.D) provides for accumulating the results of more than one MATCH statement to check for several different responses in a single answer. With the COMPUTE statement the programmer can define variables which have numeric values associated with them. These variables may be given values, manipulated, and ultimately used in conditions or even printed on the terminal.

Complete User Instructions for PILOT are contained in the PILOT Users Manual (HP 5951-5660).

HOW TO ENTER A PILOT PROGRAM

The PILOT interpretive system consists of two BASIC programs. The first is entitled 'PILOTE' and is used to enter the PILOT program and provides editing and syntax checking. The second is entitled 'PILOT" and is used to execute the PILOT program.

The PILOT program is stored on one to three disk files, but three program files must be declared during the syntax phase. The execution of 'PILOTE' may be halted at each input with the command STOP or with a 'ctrl C'. The 'break' should be used only during a LIST.

The three program files, in which the PILOT program is stored, are designated by the user. They should contain at least two records each. In practice the entire PILOT program may fit into one or two files. The PILOT program is stored three, four, five, or six statements per record. Thus a maximum length of 864 PILOT statements is allowed.

These files need not be the same size. A fourth scratch file of at least two records must be opened before entering a PILOT program. It may be longer but only the first two records are used in editing the three program files. A files statement must be declared in line 1 of 'PILOT' before executing.

EXAMPLE:

```
OPEN-FILE1,3Ø
OPEN-FILE2,3Ø
OPEN-FILE3,2Ø
OPEN-SCRACH,2

GET-PILOTE
1 FILES FILE1,FILE2,FILE3,SCRACH
```

Since a total of 80 records have been opened the PILOT program must be less than 480 statements. The fourth file, 'SCRACH' must be at least 2 records.

Tape dumped by the LIS command may be read back if the terminal has an automatic reader control. Place the tape in the reader and turn it to start after a question mark has been printed by PILOTE for a new statement. It will proceed to read in the tape. At the end of the tape an extra ? will have been read in, typing a carriage return will clear this.

PILOT COMMAND	PURPOSE
LISt	The LISt command is used to list all or only a portion of the PILOT program on the terminal.
NUMber	The NUMber command is used to renumber the line numbers of an existing PILOT program.
PURge	The PURge command completely erases the PILOT program; that is, the program files are filled with eof's.
SIZe	The SIZe command gives the number of statements, the number of records, and the number of the last statement.
STOp	The STOp command is used to achieve an orderly exit from the program 'PILOTS'. The files are left in such a way as to allow future modification or execution.

In all cases only the first three letters of the command are required.

HOW TO RUN A PILOT PROGRAM

The BASIC program to execute a PILOTE program is entitled 'PILOT'. It requires, in addition to the PILOT program files, a scratch file as long as necessary. These are to be entered in statement 1.

In the following example, a PILOT program has been stored on the files: FILE1 and FILE2.

```
OPEN-SCR,48
GET-PILOT
1 FILES FILE1,FILE2,SCR
RUN
PILOT

NUMBER PROGRAM FILES?2
.
.
.
```

The first part of the execution of 'PILOT' sets up reference tables, checks for unreferenced JUMP statement and USE statements, and sets up a response table. The reference table consists of statement numbers and label names for all the labeled PILOT statements. The response table consists of string variables and a place to store a response for all ANSWER statements with a string variable as an object. Both these tables are stored on the scratch file.

The response table will take one-half of a record for each ANSWER statement with a string variable object. Thus in writing a PILOT program it is best to keep labels short and the number of ANSWER statement with string variable objects to a minimum. As a rough example; out of a PILOT program of 600 statements, one may have perhaps 80 labeled statements. If these are relatively short, say less than 20 characters, then the reference table would take about 10 records leaving room for 76 stored responses in the next 38 records. In many cases it is not necessary to actually store the student's response in order to analyze it. These numbers are based on the assumption that the scratch file is the maximum length allowed of 48 records.

After the initial "set up" phase, the program 'PILOT' proceeds to interpret the PILOT language program. When the PILOT program is finished, control returns to 'PILOT' and the user is asked if he wishes to repeat the execution of the PILOT program. If the response is 'YES' (or just 'Y'), the PILOT program is re-executed, if not, control is returned to the BASIC system.

```
RUN

1  FILES P-DEMO,SCR
RUN
PILOT

NUMBER PROGRAM FILES ?1

                              PILOT


  HELLO!  I AM YOUR FRIENDLY HP 2100A.

WHAT IS YOUR NAME ?GEORGE

THAT IS A NICE NAME GEORGE.  I RATHER LIKE IT!!

  IF YOU DON'T THINK IT TOO PERSONAL . . .
WHAT IS YOUR SEX ?MALE

  THAT WAS WHAT I WAS AFRAID OF!
I WOULD ALSO LIKE YOUR AGE PLEASE ?2

I DON'T BELIEVE YOU!!!
I WOULD ALSO LIKE YOUR AGE PLEASE ?15


  I HAVE A FEW QUESTIONS FOR YOU.
ARE YOU READY?NOPE
  NOW WHAT'S THE MATTER???
ARE YOU READY?YES

WHO IS THE PRESIDENT OF THE U.S. ?SAM ERVIN
YOU MUST BE A DEMOCRAT, OR ARE YOU JUST DEAD??

WHO IS THE PRESIDENT OF THE U.S. ?NIXON
MY, YOU ARE UP TO DATE!!

FOR A HANDSOME GUY OF  15   YEARS, YOU TOOK ONLY  2    TRIES!!

CAN YOU NAME AT LEAST THREE COMPUTER MANUFACTURERS??
?NO
WELL, TRY ANYWAY!
?HP,IBM,XEROX

RIGHT ON, GEORGE!  I CAN TELL YOU'RE AN EXPERT!

YOU DID SO WELL ON THAT LAST QUESTION, LET ME TRY A MORE THOUGHT
PROVOKING ONE!

I AM THINKING OF A NUMBER BETWEEN 0 AND 100.  CAN YOU GUESS IT???
  1    ?50
TOO SMALL!
  2    ?75
TOO SMALL!
  3    ?88
TOO SMALL!
  4    ?95
TOO LARGE!
  5    ?92
TOO SMALL!
  6    ?93
TOO SMALL!
  7    ?94
THAT IS VERY GOOD GEORGE, YOU DID IT IN SEVEN TRIES OR LESS!
WOULD YOU LIKE TO TRY IT AGAIN ?NO

I HAVE ONE LAST QUESTION FOR YOU GEORGE!  WHAT DOES CAI MEAN??
?I DON'T KNOW
YOU BLEW IT!  CAI STANDS FOR COMPUTER ASSISTED INSTRUCTION.  THAT'S
WHAT WE'RE DOING RIGHT NOW!!

WELL, GEORGE, OUT OF THE   3    QUESTIONS I ASKED (NOT COUNTING THE
NUMBER GUESSING ONE), YOU GOT   1    CORRECT ON THE FIRST TRY.  YOU GOT
AT LEAST ONE CORRECT.
END
```

SPECIAL INSTRUCTIONS

A METHOD FOR LINKING PILOTE AND PILOT FOR THE 2000E

Normally PILOTE and PILOT allow any files. This is an advantage when executing pre-written PILOT language programs, but is a disadvantage when writing and debugging a PILOT program. One method to improve the ease for debugging is to forego the advantage of any possible files names, that is, select certain file names which are then fixed and then chain the two programs together via a "RUN" command. The procedure is given below.

```
GET-PILOTE
1 FILES F1,F2,F3,SCR
1190 CHAIN "$PILOTR"
2140 IF A$(W,H)="RUN" THEN 1190
NAM-PILOTQ
SAV
```

```
GET-PILOT
1 FILES F1,F2,F3,SCR
1125 CHAIN "$PILOTQ"
1535 GOTO 1125
6050 GOTO 1125
1091
1092
1093 Q1=3
NAM-PILOTR
SAV
```

The procedure is given assuming the programs PILOTQ and PILOTR are to live on the system library. If not, then the '$' in the CHAIN statements should be so modified. The choice of file names is arbitrary, but should agree between the two programs.

To use this system first OPEN the appropriate files:

```
OPE-F1,48
OPE-F2,48
OPE-F3,48
OPE-SCR,12
```

To execute the PILOT program from the editing portion of PILOTQ simply enter RUN. All halting of the execution phase will return the system to the editing phase.

POLISH
36888-18014

TITLE: POLISH CONVERSION

DESCRIPTION:

POLISH is a short program written to convert input string into reverse POLISH form and to use the POLISH stack and a list of the prior assignments made for operators and operands. To conserve memory space, no syntax checks are made on the input string. Single letters of the alphabet should be used as operands and the permitted operators ↑, *, /, +, -. Parentheses may be nested as deeply as desired. Blanks are not permitted.

INSTRUCTIONS:

The execution priority is shown below:

OPERATION:	PRIORITY LEVEL:
OPERAND	0
(	1
)	2
+	3
-	3
*	4
/	4
↑	5

Reference to the sample inputs attached will show typical input strings and the program usage. Instructions for using the program are contained within the program itself.

SPECIAL CONSIDERATIONS:

The algorithm used in generating the POLISH conversion is covered in the textbook "A Guide to FORTRAN IV Programming", by McCracken. The user should refer to that text for additional information.

FOR INSTRUCTIONAL PURPOSES

Suitable Course(s): Secondary School Level

Student Background Required: Mathematics

Used in beginning programming classes at the Freshman level in the College of Technology, University of Houston, to assist students in understanding the need for the use of reverse polish notation. It has been found that this program helps the students understand the necessity for using POLISH notation in evaluating expressions even though they may not understand the techniques employed in the program itself.

ACKNOWLEDGEMENTS:

Professor George C. McKay, Jr.
Electrical-Electronics Technology
University of Houston

RUN

```
RUN
POLISH

TYPE EXPRESSION USING SINGLE LETTERS AS OPERANDS. PERMITTED
OPERATORS ARE '↑','*','/','+','-'. PARENTHESES MAY BE NESTED
TO ANY DEPTH. SPACES ARE NOT PERMITTED. PROGRAM WILL OUTPUT
POLISH STACK (REVERSE) AND LIST OF OPERATORS' PRIORITY..
F(X)=?A+B*(((C-D↑E)/F)+G)-H
A+B*(((C-D↑E)/F)+G)-H
INPUT STRING PRIORITY IN ORDER:
 Ø     3     Ø     4     1     1     1     Ø     3     Ø     5     Ø
 2     4     Ø     2     3     Ø     2     3     Ø
POLISH STACK
ABCDE↑-F/G+*+H-
?A+B*C
A+B*C
INPUT STRING PRIORITY IN ORDER:
 Ø     3     Ø     4     Ø
POLISH STACK
ABC*+
?A+B-C↑E
A+B-C↑E
INPUT STRING PRIORITY IN ORDER:
 Ø     3     Ø     3     Ø     5     Ø
POLISH STACK
AB+CE↑-
?A+B-(C*D↑E)
A+B-(C*D↑E)
INPUT STRING PRIORITY IN ORDER:
 Ø     3     Ø     3     1     Ø     4     Ø     5     Ø     2
POLISH STACK
AB+CDE↑*-
?((((((A/B*C))))))
((((((A/B*C))))))
INPUT STRING PRIORITY IN ORDER:
 1     1     1     1     1     1     Ø     4     Ø     4     Ø     2
 2     2     2     2     2
POLISH STACK
AB/C*
?
DONE
```

Documentation Date 3/75

PROGRAMMING AND COMPUTER SCIENCE (EDUCATION) (810)

CONTRIBUTED PROGRAM **BASIC**

PTUD
36888-18016

TITLE: PILOT TUTORIAL

DESCRIPTION: This is the fourth program in the PILOT tutorial series. It is written in PILOT to teach the syntax and usage of PILOT.

This portion discusses the DEMAND and the COMPUTE statements.

INSTRUCTIONS: This program is stored on 3 files named P-TUD1, P-TUD2, P-TUD3 for use on the HP 2000E systems and is stored on 1 file named PTUD for the 2000 series system.

The scratch file should be at least three records.

SPECIAL CONSIDERATIONS:

on 2000E: P-TUD1,48
P-TUD2,48
P-TUD3,48

on 2000 Series System: PTUD,70

ACKNOWLEDGEMENTS: Lawrence E. Turner
Department of Physics and Computer Science
Pacific Union College

RUN

```
CRE-SCR,5
GET-PILOTF
RUN
PILOTF

NAME OF PILOT PROGRAM ?PTUD
NAME OF SCRATCH FILE  ?SCR


?RUN


                                PILOT



HI!  I AM READY TO GO AGAIN, HOW ABOUT YOU??

CONSIDER THIS SECTION OF A PROGRAM:

   NAME TWO STATES BORDERING ON CALIFORNIA ?OREGON,ARIZONA
   VERY GOOD!

THIS PROGRAM TAKES A TOTAL OF 8 STATEMENTS (INCLUDING A NEGATIVE RETORT
WHICH YOU DID NOT GET SINCE YOU ANSWERED CORRECTLY).  IT ALSO TAKES
INTO ACCOUNT ANY COMBINATION OF THE THREE POSSIBLE STATES!

WHAT STATEMENT IS NECESSARY FOR THIS ?DEMAND
EXCELLENT, I CAN TELL YOU DID YOUR HOMEWORK!
LET'S LOOK AT A LISTING OF THE PROGRAM.

        7000 T:NAME TWO STATES BORDERING ON CALIFORNIA +
        7010 A:
        7020 M:ARIZON,AZ
        7030 M:NEVAD,NV
        7040 M:OREG,OR
        7050 .D:2
        7060 Y:VERY GOOD!
        7070 N:NOPE.

THE HEART OF THIS IS STATEMENT 7050, THE DEMAND STATEMENT.
WHAT DO STATEMENTS 7020, 7030, AND 7040 DO ?MATCH
YES, THEY CHECK FOR THE POSSIBLE CORRECT STATES.  NOW IF STATEMENT 7050
WERE LEFT OUT, THEN WHAT RESPONSE WOULD SET THE MATCH FLAG TO THE 'YES'
STATE ?R-OREGON
EXACTLY!  ONLY 'OREGON' WOULD INITIATE 'VERY GOOD!'.  WHAT THE DEMAND
STATEMENT DOES IS TO CHANGE THE MATCH FLAG TO 'YES' IF TWO OR MORE OF
THE EXECUTED MATCH STATEMENTS SINCE THE LAST EXECUTED ANSWER STATEMENT
HAVE BEEN SUCCESSFUL, OTHERWISE IT IS SET TO 'NO'!
WHAT OTHER PILOT STATEMENTS CAN CHANGE THE MATCH FLAG ?ANSWER
CORRECT, BUT YOU FORGOT ONE.  BOTH THE ANSWER AND THE MATCH CAN ALSO
SET THE MATCH FLAG.
IF STATEMENT 7050 WERE TO READ '7050 .D:1', THEN HOW MANY CORRECT
STATES WOULD HAVE TO BE ENTERED IN ORDER TO GET 'VERY GOOD!' ?1
EXACTLY!  THE INTEGER OBJECT OF THE DEMAND STATEMENT (IN THIS CASE 1)
TELLS HOW MANY EXECUTED MATCH STATEMENTS MUST BE SUCCESSFUL SO THAT THE
'Y' CONDITION STATEMENTS WILL BE EXECUTED.
CONSIDER:

        7050 T:YOU GOT +
        7060 .D:1
        7070 Y:ONE+
        7080 .D:2
        7090 Y:, TWO+
        7100 .D:3
        7110 Y:, THREE+
        7120 .D:1
        7130 Y: OF THEM!
        7140 N:NOT EVEN ONE!
```

TRY IT!

```
NAME TWO STATES BORDERING ON CALIFORNIA ?OREGON,ARIZONA
YOU GOT ONE, TWO OF THEM!
```

TRY IT AGAIN!

```
NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA
YOU GOT ONE, TWO OF THEM!
```

TRY IT AGAIN!

```
NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA,OREGON
YOU GOT ONE, TWO, THREE OF THEM!
```

ONE MORE TIME.

```
NAME TWO STATES BORDERING ON CALIFORNIA ?ARIZONA,NEVADA,OREGON
YOU GOT ONE, TWO, THREE OF THEM!
```

THERE ARE, OF COURSE, MANY VARIATIONS MADE POSSIBLE BY THE DEMAND STATEMENT. CONSIDER:

```
7050 .D:3
7060 Y:VERY, VERY GOOD!  YOU GOT ALL THREE!
7070 JY:*NEXT
7080 .D:2
7090 Y:CORRECT!
7100 JY:*NEXT
7110 .D:1
7120 Y:YOU DID GET ONE RIGHT.
7130 N:YOU DIDN'T EVEN GET ONE!
7140 *NEXT R:
```

WHY DO YOU THINK THE JUMP STATEMENTS ARE NECESSARY??
?

IF .D:3 IS SATISFIED, SO IS .D:2 AND .D:1!! THUS IF THREE ARE FOUND, THEN YOU MUST BRANCH AROUND THE SECTIONS THAT CHECK FOR TWO AND ONE. NOTE THAT THE DEMAND STATEMENT CHECKS FOR 'AT LEAST'. IT DOES NOT CHECK FOR 'EXACTLY'.
OK, WHAT DO YOU THINK .D:0 SETS THE MATCH FLAG TO ?
??YES
RIGHT ON! SINCE IN ALL CASES AT LEAST ZERO MATCH STATEMENTS HAVE BEEN SATISFIED, .D:0 ALWAYS SETS THE MATCH FLAG TO 'YES'. THIS IS A WAY OF FORCING THE MATCH FLAG IRREGARDLESS OF PREVIOUS MATCHES.
AN EXAMPLE:

```
3560 T:CAN YOU NAME THE CLOSEST STAR TO THE EARTH +
3570 A:
3580 M:YES,OF COURS,RIGHT,CERTAIN,SURE,Y
3590 Y:OK, WHAT IS IT +
3600 *STAR AY:
3610 M:SUN,SOL
3620 Y:VERY GOOD!
3630 JY:*NEXT
3640 T:NO WAY, HERE'S A HINT:  YOU ONLY SEE IT DURING THE DAY.
3650 T:NOW TRY IT +
3660 .D:0
3670 J:*STAR
```

IF STATEMENT 3660 WERE MISSING, WOULD *STAR AY: BE EXECUTED AFTER THE JUMP (J:*STAR) ?NO
THAT'S RIGHT! IN FACT, THERE WOULD BE NO WAY TO GET OUT OF THE LOOP!!
ACTUALLY THERE ARE PERHAPS OTHER WAYS OF ACHIEVING THIS SAME RESULT WITHOUT RESORTING TO THE .D:0, BUT AT TIMES IT IS CONVENIENT.
THERE IS ONE FURTHER POINT TO CONSIDER. IF THERE ARE TOO MANY ALTERNATIVES IN A MATCH OBJECT TO FIT ON ONE LINE, THEN HOW DO YOU 'CONTINUE' A MATCH??
?CONDITION
ACTUALLY, BY THE USE OF AN 'N' CONDITION YOU CAN 'CONTINUE' AS LONG AS YOU WISH.
THE QUESTION TO BE ANSWERED IS: DOES THIS AFFECT THE FUNCTION OF THE DEMAND STATEMENT?
WHAT DO YOU THINK ?NO
RIGHT! IT DOES WORK PROPERLY IF YOU THINK OF THE TWO OR MORE STATEMENTS 'CONTINUED' TOGETHER AS A SINGLE MATCH STATEMENT. AT THE MOST THERE WILL BE ONE SUCCESSFUL MATCH AMONG THEM. TO SEE THIS, EXAM:

```
4070 T:NAME A DIGIT AND A COLOR.
4080 A:
4090 M:ZER,ONE,TWO,THRE,FOUR,FIVE
4100 MN:SIX,SEVEN,EIGHT,NINE
4110 M:BLUE,RED,GREEN,ORANGE,GREY
4120 MN:PINK,BLACK,PURPLE,VIOLET
4130 MN:YELLOW,WHITE,BROWN
4140 .D:2
4150 Y:GROOVY!
4160 N:I GIVE UP!
```

IF 'TWO' IS TYPED IN, HOW MANY SUCCESSFUL MATCH STATEMENTS WILL EXECUTED BEFORE 4140 ?1
CORRECT. ONLY NUMBER 4090 WILL BE SUCCESSFUL, IN FACT 4100 WILL NEVER BE EXECUTED!
NOW WHAT IF 'NINE AND RED' ?2
RIGHT. BOTH 4100 AND 4110 ARE SUCCESSFUL. NUMBER 4090 IS EXECUTED BUT FAILS. 4120 AND 4130 ARE NOT EVEN EXECUTED.
IF 'NOTHING' IS TYPED IN ?0
OK, THEY ALL WILL BE EXECUTED, BUT NONE SUCCESSFULLY.
NOTICE THAT EACH SET OF 'CONTINUED' MATCH STATEMENTS ACTS LIKE ONLY ONE STATEMENT AS FAR AS BOTH THE MATCH FLAG AND THE DEMAND STATEMENT.

WHAT DO YOU THINK IS THE MAXIMUM NUMBER OF SUCCESSFUL MATCHES THAT CAN BE 'DEMANDED' ?99
I GUESS YOU REALLY DID YOUR HOMEWORK! THE LARGEST ALLOWED OBJECT OF A DEMAND STATEMENT IS 99. THUS YOU CAN HAVE UP TO 99 MATCHES TO BE 'DEMANDED'! IT IS RARE THAT YOU WILL EVER NEED THIS MANY FOR A VERY SIMPLE REASON.
WHAT IS THE LONGEST RESPONSE ALLOWED BY PILOT ?72
CLOSE! EXACTLY! UP TO 72 CHARACTERS MAY BE TYPED IN AT ANY TIME. THIS IS THE MOST THAT PILOT CAN CHECK FOR POSSIBLE MATCHES AT ANY ONE TIME.

THAT COMPLETES EVERYTHING THERE IS TO KNOW ABOUT DEMAND! YOU NOW KNOW (OR HAVE BEEN EXPOSED TO) MOST OF THE PILOT LANGUAGE STATEMENTS. YOU CAN CAUSE THE COMPUTER TO PRINT OUR STUFF, ASK QUESTIONS, MATCH THE RESPONSE, MAKE APPROPRIATE COMMENTS, JUMP, AND NOW DEMAND. THERE ARE ONLY THREE MORE THINGS TO KNOW AND YOU WILL BE AN EXPERT! THESE ARE: HOW TO MODIFY THE VALUE OF A NUMERIC VARIABLE, SUBROUTINE JUMPING, AND A MORE POWERFUL MATCH STATEMENT.

LET'S REVIEW WHAT YOU KNOW ABOUT NUMERIC VARIABLES. FIRST HOW MANY DIFFERENT NUMERIC VARIABLES ARE THERE ?26
PRECISELY!
WHAT PILOT STATEMENT(S) CAN MODIFY THE VALUE STORED IN A NUMERIC VARIABLE ?DONT KNOW
SO FAR WE HAVE ONLY DISCUSSED HOW THE ANSWER STATEMENT CAN MODIFY THE VALUE OF A VARIABLE.
WHAT PART OF AN ANSWER STATEMENT ALLOWS THIS ?AFTER THE COLON
BEAUTIFUL! AND WHAT SPECIAL SYMBOL SIGNIFIES A NUMERIC VARIABLE ?#
OF COURSE!
NAME TWO WAYS IN WHICH A NUMERIC VARIABLE MAY BE USED.
?MATCH AND TYPE
WELL, YOU GOT ONE. THE FIRST WAY WE STUDIED WAS TO OUTPUT THE VALUE IN A TYPE STATEMENT. THE SECOND WAS IN A CONDITION. HERE IS A PART OF A PROGRAM TO ILLUSTRATE:

```
4520 T:WHAT IS YOUR AGE +
4530 *PLEASE A:#A
4540 TB:GIMME A NUMBER!
4550 JB:*PLEASE
4560 T(A<=0):WHAT?  A NEGATIVE NUMBER!
4570 T(A>150):A FOSSIL IF I EVER SAW ONE!
4580 T:ARE YOU REALLY #A YEARS OLD???
```

WHAT DOES THE CONDITION 'B' DO ?DONT KNOW
I ALMOST FEEL LIKE GIVING UP ON YOU! THE STATEMENTS WITH A 'B' CONDITION ARE EXECUTED ONLY IF NO VALID NUMBER WERE ENTERED. IT ALLOWS TESTING OR CHECKING FOR A NUMBER IN THE RESPONSE.
SOMETIMES IT IS DESIRED TO MODIFY THE VALUE OF A NUMERIC VARIABLE OTHER THAN IN AN ANSWER STATEMENT. THERE ARE THREE PILOT STATEMENTS THAT DO THIS. THE FIRST AND MOST POWERFUL IS THE COMPUTE STATEMENT. WHAT DO YOU THINK IS THE INSTRUCTION FOR THE COMPUTE STATEMENT ?.C
AHA, YOU ARE TOO TRICKY! THE INSTRUCTION FOR THE COMPUTE STATEMENT IS SIMPLY A 'C'.
LET'S LOOK AT A FEW EXAMPLE COMPUTE STATEMENTS:

```
7350 C:A=5
8120 CY:B=A+32*X
8320 C(Z>3):S=T*(2+N)
9040 C:H=K
9450 C:N=N+1
```

THE OBJECT OF THE COMPUTE STATEMENT BEGINS WITH WHAT ?CHARACTER
OH MY ACHING HEADS! WILL YOU WAKE UP?? THE FIRST THING IN THE COMPUTE OBJECT IS A NUMERIC VARIABLE. IT IS THE ONE WHOSE VALUE IS TO BE MODIFIED.
WHAT IS THE NUMERIC VARIABLE THAT IS TO BE MODIFIED IN STATEMENT 8320 ABOVE ?S
ALRIGHT, THE VARIABLE S WILL BE MODIFIED WHEN 8320 IS EXECUTED.
WHAT IS THE NEXT ITEM IN ALL COMPUTE STATEMENT AFTER THE LEADING NUMERIC VARIABLE ?=
GOOD. ALL COMPUTE STATEMENTS MUST HAVE AN EQUALS SIGN (=). THE PART TO THE RIGHT OF THE EQUALS SIGN IS VERY GENERALLY TERMED AN ARITHMETIC EXPRESSION OR SIMPLY EXPRESSION.

```
C:<NUMERIC VARIABLE>=<EXPRESSION>
```

THIS IS HOW YOU WOULD DIAGRAM A GENERAL COMPUTE STATEMENT. THE '<>' SURROUND A SINGLE SYNTACTICAL PILOT ENTITY. I HAVE LEFT OFF THE STATEMENT NUMBER AND ANY OPTIONAL LABEL OR CONDITION, THEY ARE MOST CERTAINLY ALLOWED. WHERE DOES THE CONDITION GO??
?

•
•
•

PTUE
36883-18017

TITLE: PILOT TUTORIAL

DESCRIPTION:

This is the fifth and last program in the tutorial series in PILOT to teach the elements of the PILOT language.

This program deals with additional usage of the COMPUTE statement, the INTEGER FUNCTION, the RANDOM NUMBER FUNCTION, USE and END, and the extended MATCH.

INSTRUCTIONS:

This program is stored in 3 files: P-TUE1, P-TUE2, P-TUE3 for the HP 2000E system and on the one file PTUE for the HP 2000E series system.

The scratch file should be at least 2 records in length.

SPECIAL CONSIDERATIONS:

2000E: P-TUE1, 48; P-TUE2, 48; P-TUE3, 48

2000 Series System; PTUE, 70

ACKNOWLEDGEMENTS:

Lawrence E. Turner, Jr.
Department of Physics and Computer Science
Pacific Union College

RUN

```
CRE-SCR,5
GET-PILOTF
RUN
PILOTF

 NAME OF PILOT PROGRAM ?PTUE
 NAME OF SCRATCH FILE  ?SCR


?RUN
```

PILOT

GREETINGS! THIS IS THE FINAL PROGRAM IN THIS SERIES. YOU ARE ALMOST FINISHED. ACTUALLY, I HOPE THIS IS JUST A BEGINNING OF YOUR USAGE OF PILOT!

LAST TIME WE DISCUSSED THE COMPUTE STATEMENT AND AT LEAST ONE EXAMPLE OF HOW IT COULD BE USED. DO YOU REMEMBER THAT USE ?YES
VERY GOOD, WHAT IS IT ?
??FORGOT
WELL, WHAT I HAD IN MIND WAS THE USE AS A COUNTER. AS IN:

```
1730 C:M=M+1
2960 CY:P=P+1
4110 C(X<17):D=D+1
```

BESIDES THE COMPUTE STATEMENT, WHAT PILOT STATEMENT CAN MODIFY THE VALUE OF A NUMERIC VARIABLE ?ANSWER STATEMENT
RIGHT ON! SO FAR WE HAVE ONLY EXAMINED THE ANSWER STATEMENT. THIS ALLOWS A QUESTION WHICH ASKS FOR A NUMERIC RESPONSE.
THE COMPUTE STATEMENT MAY BE USED IN CONJUNCTION WITH THIS TO HELP EXTRACT THE INTENDED RESPONSE MORE ACCURATELY. CONSIDER:

```
6310 T:HOW MANY DAYS IN ONE WEEK +
6320 A:#D
6330 M:SEVEN
6340 CY:D=7
6350 T(D=7):EXACTLY!  +
6360 T(D#7):SORRY, +
6370 T:THERE ARE SEVEN DAYS IN ONE WEEK.
```

WHAT IS THE PURPOSE OF STATEMENTS 6330 AND 6340??
?TO ALLOW FOR NUMERIC 7 AND ALPHABETIC SEVEN
RIGHT, THEY CATCH THE RIGHT RESPONSE IF THE PERSON HAPPENED TO TYPE 'SEVEN' INSTEAD OF '7'. BOTH ARE CORRECT.
LET'S EXECUTE THIS PROGRAM.

```
HOW MANY DAYS IN ONE WEEK ?7
EXACTLY!  THERE ARE SEVEN DAYS IN ONE WEEK.
```

NOW TRY IT AND ENTER 'SEVEN' SPELLED OUT.

```
HOW MANY DAYS IN ONE WEEK ?SEVEN
EXACTLY!  THERE ARE SEVEN DAYS IN ONE WEEK.
```

OFTEN ONE ONLY WISHES THAT THE RESPONSE BE CLOSE TO THE CORRECT ANSWER. AGAIN THE COMPUTE STATEMENT TO THE RESCUE!

```
8400 T:HOW MANY CENTIMETERS IN ONE INCH +
8410 A:#X
```

THE CORRECT ANSWER IS 2.54; HOWEVER, TO BE CONSIDERED CORRECT YOU MIGHT ALLOW ANY NUMBER BETWEEN 2.5 AND 2.58. THIS WOULD ALLOW ANY RESPONSE THAT IS WITHIN 0.04 OF THE CORRECT ANSWER TO BE COUNTED. A VERY SIMPLE WAY IS TO COMPUTE THE DIFFERENCE BETWEEN THE CORRECT ANSWER AND THE RESPONSE.
HOW WOULD YOU DO THIS ?C:D=2.54-X
I BELIEVE THAT IS CORRECT! A COMPUTE STATEMENT THAT LOOKS LIKE THIS SHOULD DO THE TRICK:

```
8420 C:D=2.54-X
```

IF THE RESPONSE IS: '2.5' WHAT IS STORED IN D ?.04
EXACTLY! D HAS 0.04 STORED IN IT. SINCE 2.5 IS AN ALLOWED 'CORRECT ANSWER', HOW WOULD YOU CAUSE THE APPROPRIATE MESSAGE TO BE TYPED OUT??
?C:←T(D<=0.04):CORRECT!
GOOD. BY USING D IN A CONDITION ON A TYPE STATEMENT YOU CAN PRINT THE CORRECT RETORT. CONSIDER:

```
8400 T:HOW MANY CENTIMETERS IN ONE INCH +
8410 A:#X
8420 C:D=2.54-X
8440 T(D<=0.04):VERY GOOD!
8450 T(D>0.04):NOT QUITE.
```

THIS IS COOL FOR GUESSES LESS THAN 2.54, BUT WHAT WOULD BE PRINTED IF '57' WERE ENTERED ?VERY GOOD!
EXACTLY! NOW IS '57' A 'CORRECT RESPONSE' ?NO
OF COURSE NOT, THUS WE HAVE A BUG TO FIX!! WHAT WE REALLY WANT IS THE ABSOLUTE DIFFERENCE FOR D. SO IF D IS LESS THAN ZERO, WE WANT TO CHANGE ITS SIGN. THIS CAN BE DONE WITH ONE PILOT STATEMENT:

```
8430 C(D<0):
```

WHAT IS THE CORRECT OBJECT ?D=-D
VERY GOOD!
THUS OUR PROGRAM BECOMES:

```
8400 T:HOW MANY CENTIMETERS IN ONE INCH +
8410 A:#X
8420 C:D=2.54-X
8430 C(D<0):D=-D
8440 T(D<=0.04):VERY GOOD!
8450 T(D>0.04):NOT QUITE.
```

EXCEPT FOR INTEGER RESPONSES IT IS BEST TO GO THROUGH SOMETHING LIKE THIS FOR NUMERIC RESPONSES, EVEN FOR SIMPLE DECIMAL NUMBERS. THE COMPUTER CANNOT REPRESENT MOST DECIMALS EXACTLY, HENCE SOMETHING LIKE: 7320 T(X=.1): ... MAY NEVER BE EXECUTED EVEN IF '0.1' HAS BEEN PREVIOUSLY STORED IN X. SOMEWHERE, PERHAPS IN THE SEVENTH DECIMAL PLACE, X MAY DIFFER SLIGHTLY FROM 0.1, AND THE TEST WILL FAIL.

ANOTHER USEFUL WAY TO MODIFY THE VALUE OF A NUMERIC VARIABLE IS TO REPLACE IT WITH THE GREATEST INTEGER LESS THAN OR EQUAL TO THE ORIGINAL NUMBER.
FOR 4.7, WHAT IS THIS INTEGER ?4
RIGHT ON!
HOW ABOUT 0.247 ?0
BEAUTIFUL!
OK, TRY 6.00 ?6
GOOD.
AND -2.7 ?-3
EXCELLENT!
FINALLY, -7.00 ?-7
PERFECT! NOTE THAT FOR POSITIVE NUMBERS THIS GREATEST INTEGER FUNCTION IS EQUIVALENT TO JUST THROWING AWAY ANY DECIMAL PORTION; HOWEVER, THIS SIMPLE-MINDED DESCRIPTION IS NOT ADEQUATE FOR NEGATIVE NUMBERS.
THIS OPERATION IS DIFFICULT, IF NOT IMPOSSIBLE, TO DO WITH THE COMMON ARITHMETIC OPERATORS AS ALLOWED IN THE COMPUTE STATEMENT, YET IT IS SUFFICIENTLY USEFUL TO BE INCLUDED IN THE PILOT LANGUAGE.
A NEW INSTRUCTION IS USED, WHAT DO YOU THINK IT IS ?I
THAT IS A GOOD GUESS, BUT THE INTEGER FUNCTION (REALLY THE GREATEST INTEGER THAT IS LESS THAN OR EQUAL TO) IS DESIGNATED BY THE EXTENDED INSTRUCTION '.I'. EXAMPLES ARE:

```
3710 .I:X
4290 .I:S
6660 .I(X>9):X
```

WHAT ARE THE OBJECTS OF THESE INTEGER FUNCTION STATEMENTS ?NUMERIC VARI■
NUMERIC VARIABLES
EXACTLY! THE OBJECT IS A SINGLE NUMERIC VARIABLE. ITS VALUE IMMEDIATELY BEFORE THE EXECUTION OF THE STATEMENT IS REPLACED BY THE GREATEST INTEGER THAT IS LESS THAN OR EQUAL TO IT.
TO SEE A POSSIBLE USE, CONSIDER:

```
3250 T:THREE IS WHAT PERCENT OF 24 +
3260 A:#P
3270 .I:P
3280 T(P=12):EXCELLENT!
3290 T(P#12):NOPE.
```

WHAT IS THE SMALLEST NUMBER FOR WHICH 'EXCELLENT' WILL BE PRINTED ?12
RIGHT ON!
HERE THE CORRECT ANSWER IS 12.5, BUT ANYTHING FROM 12 UP TO (BUT NOT INCLUDING) 13 IS ACCEPTABLE. THE .I VERY EASILY SETS UP THE RANGE.
A MORE IMPORTANT USE IS FOR ROUNDING. THE INTEGER FUNCTION AS GIVEN DOES NOT ROUND, IT MERELY TRUNCATES. ROUNDING CAN BE ACHIEVED IN QUITE A STRAIGHTFORWARD MANNER BY:

```
4560 C:X=X+.5
4570 .I:X
```

WHAT WILL BE STORED IN X IF IT PREVIOUSLY HAD 4.00 ?4
EXACTLY.
HOW ABOUT 4.6 ?5
COOL!
OK, TRY -2.1 ?-2
RIGHT!
AND WHAT DOES -5.7 BECOME ?-5
MY, MY, NO. -5.7 + 0.5 IS -5.2 WHICH BECOMES -6.

NOTE THAT THE WAY THE INTEGER FUNCTION IS DEFINED MAKES THE ROUNDING OPERATION WORK FOR BOTH POSITIVE AND NEGATIVE VALUES. IF IT SIMPLY DISCARDED THE DECIMAL PART, THEN THERE WOULD HAVE TO BE SPECIAL TESTS AND HASSELING TO DO THE ROUNDING.
YOU CAN ROUND TO ANY POINT YOU WISH. CONSIDER THE FOLLOWING TO ROUND TO THE NEAREST HUNDRETH (PENNY??):

```
7310 C:Q=Q*100+.5
7320 .I:Q
7330 C:Q=Q/100
```

OK, IF Q ORIGINALLY CONTAINS 3.14159, WHAT IS THE NEW VALUE ?

•
•
•

STDATA
36888-18027

TITLE: CWF SUBPROGRAM

DESCRIPTION:

STDATA is one of a series of user-written sub-programs that extend the capabilities of Hewlett Packard's licensed instructional product: Course Writing Facility (CWF). For more information about CWF, users should consult the Course Writing Facility Reference Manual, Part no. 24383-90001.

When a CWF author invokes this sub-program with a call of the form "fn /stdata", certain data from the student's record are made available in the CWF buffer, b5. The data include the student's name, number, user group, and date of registration, as well as the time he has spent on the course and the time of his last sign-on. These data may be used directly by the author to print out messages to the student, or to affect the logical flow of the course for each individual student.

INSTRUCTIONS:

CALLING FORMAT

fn /stdata

COURSE FACILITIES AFFECTED

Counter 30 (c30) will contain the student number

Buffer 5 (b5) will contain the following:

Starting position	length	contents
0	9	student number
9	22	student's last name
31	3	user group
34	8	data of registration: MM/DD/YY
42	8	today's date: MM/DD/YY
50	8	time of sign on: hh-mm xx (AM, PM, N, M)
58	13	time on course up to last sign off (hh hrs mm min)
71	22	student's first name

The student's last name will never exceed a maximum of 20 characters and will be filled with blanks to a total of 22 characters; all 22 characters will be filled with blanks for a demo student. The last character in the name is followed by a control E, e.g., SMITHE^{C}. This causes printing to stop at the end of the student's name, for example, "ld b5,71,22/bl" loads 22 characters into bl, but "ty bl" then prints only up to the E^{C}.

Continued on following page.

SPECIAL CONSIDERATIONS:

HP 24383A, Course Writing Facility must be present in the system for this subprogram to RUN.

ACKNOWLEDGEMENTS:

Jutta Kernke
Hewlett Packard

INSTRUCTIONS continued

Similar remarks apply to the student's first name

SAMPLE CALL

```
fn /stdata
ld b5,71,22/b1
ty How are you today,
ty b1
ty ?
```

SAMPLE RUN

```
   .
   .
   .
How are you today, John?
```

CONTRIBUTED PROGRAM **BASIC**

TRAN1
36888-18038

TITLE: GRAPHICS UTILITY PROGRAMS

DESCRIPTION: There are 3 programs in this project: TRAN1, PLOT1 and PLOT2. These three programs support the publication Computer Graphics: Three Dimensional Projections: Theory, Programs and Examples, by Herbert D. Peckham published by the Hewlett-Packard Computer Curriculum Project, order number 5951-5601, price $5.50.

For further information contact:

Hewlett-Packard Computer-Based Educational Materials
Scientific Press
1629 Channing Avenue
Palo Alto, Ca 84303

INSTRUCTIONS: User's Instructions are in the publication.

SPECIAL CONSIDERATIONS: TRAN1 is the general capability program. PLOT1 requires a HP 7200 or 7202 plotter and PLOT2 requires a 4010 or 4012 Tektronix Graphics terminal and the Hewlett-Packard Primary Graphics Software (HP 20311A) See HP sales representative for details.

ACKNOWLEDGEMENTS: HERB PECKHAM
GAVILAN COLLEGE

BASP
36888-18022

TITLE: BASIC ANALOG SIMULATION PROGRAM

DESCRIPTION: BASP utilizes digital techniques to simulate the operation of an analog computer. In essence, it is used to break down descriptions of conventional analog block diagrams into sets of simultaneous first order differential equations. These differential equations are then solved by numerical methods.

INSTRUCTIONS: The user must first define his problem in terms of an analog block diagram. BASP utilizes twenty four functional block types to facilitate the construction of this block diagram from the problem description. By defining each block in terms of its function, referenced inputs, and associated parameters, the block interconnection structure and system parameters are readily available for program use. The user may then select the desired output format and adjust the run-time parameters for required accuracy.

Order HP 36888-90022, $5 for complete documentation

SPECIAL CONSIDERATIONS: Some extreme problem cases may require that the user take into account such analog computer characteristics as scaling.

ACKNOWLEDGEMENTS: Michael A. Van Cleave
University of Louisville

RUN

```
RUN
BASP



                    BASIC ANALOG SIMULATION PROGRAM
                    -------------------------------



SYSTEM DESCRIPTION:

DEFINE BLOCK FUNCTIONS,BLOCK INTERCONNECTIONS,AND ASSOCIATED PARAMETERS.
HOW MANY BLOCKS IN THE SYSTEM?6
INPUT BLOCK TYPE,INPUTS (1,2,3),AND BLOCK PARAMETERS (1,2)
BE SURE TO ENCLOSE BLOCK TYPE IN QUOTES.

BLOCK NO. 1     ?"MUL",2,5,Ø,Ø,Ø
BLOCK NO. 2     ?"INT",-1,3,Ø,5,Ø
BLOCK NO. 3     ?"POT",2,Ø,Ø,2,Ø
BLOCK NO. 4     ?"POT",1,Ø,Ø,2,Ø
BLOCK NO. 5     ?"INT",4,-6,Ø,2,Ø
BLOCK NO. 6     ?"POT",5,Ø,Ø,8,Ø



                     SEQUENCE TABLE FOR BASP RUN
      -------------------------------------------------------
      BLOCK NO.    BLOCK TYPE    INPUT 1    INPUT 2    INPUT 3
      ---------    ----------    -------    -------    -------
          3            POT          2          Ø          Ø
          1            MUL          2          5          Ø
          2            INT         -1          3          Ø
          6            POT          5          Ø          Ø
          4            POT          1          Ø          Ø
          5            INT          4         -6          Ø



DETERMINE OUTPUT FORMAT:

INPUT BLOCK NUMBERS OF OUTPUTS TO BE MONITORED.
(ENTER ZERO AS LAST INPUT IF LESS THAN THREE.)
?2
?5
?Ø



OUTPUT OPTIONS ARE AS FOLLOWS :
  (1) TABULAR
  (2) TABULAR PLUS GRAPHICAL

WHICH IS YOUR CHOICE?2



DEFINE RUN-TIME PARAMETERS:

INPUT INITIAL AND FINAL VALUES OF THE INDEPENDENT VARIABLE?Ø,5
ENTER THE INTEGRATION STEP SIZE?.1
```

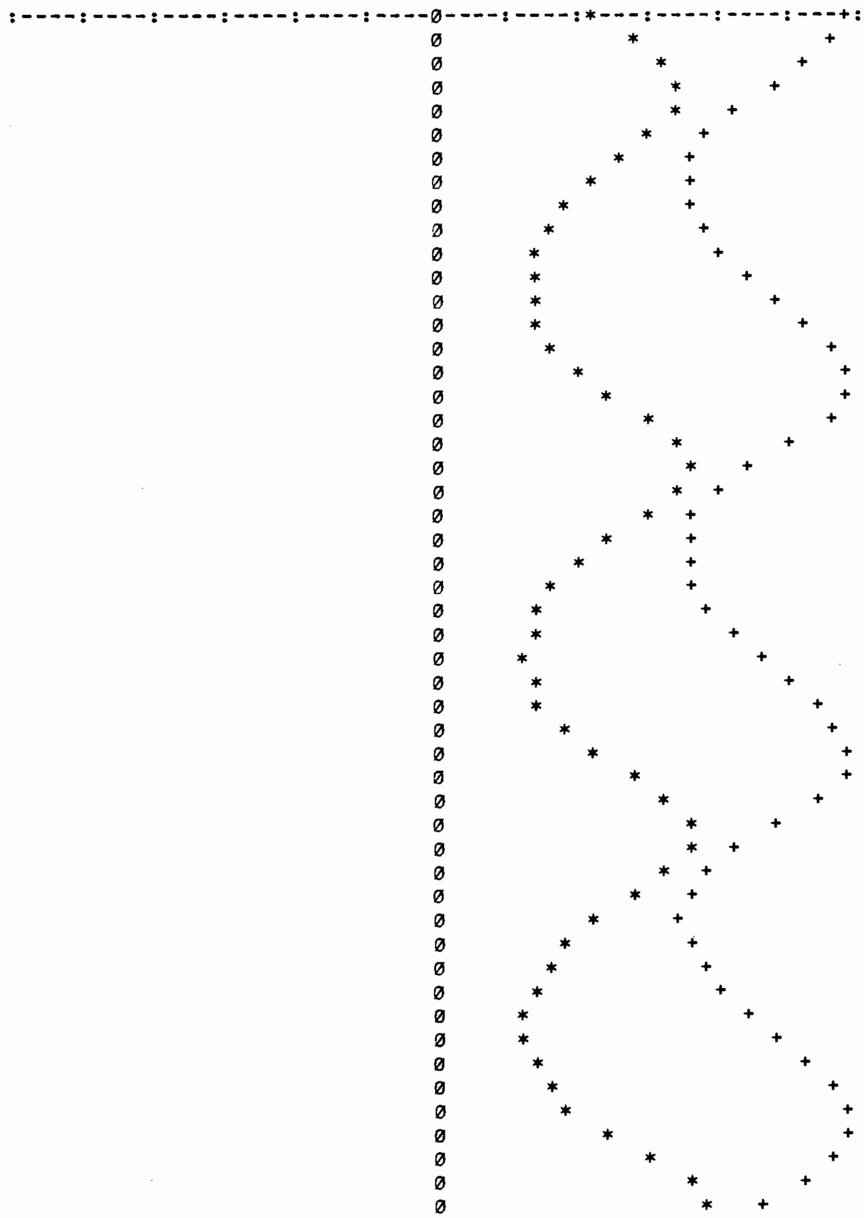
BLOCK OUTPUT TABLE

IND. VAR.	BLOCK NO. 2	BLOCK NO. 5	BLOCK NO. Ø
Ø.ØØØØØE+ØØ	+5.ØØØØØE+ØØ	+2.ØØØØØE+ØØ	+Ø.ØØØØØE+ØØ
1.ØØØØØE-Ø1	+4.9ØØØØE+ØØ	+2.44ØØØE+ØØ	+Ø.ØØØØØE+ØØ
2.ØØØØØE-Ø1	+4.58627E+ØØ	+2.85665E+ØØ	+Ø.ØØØØØE+ØØ
3.ØØØØØE-Ø1	+4.13999E+ØØ	+3.Ø8585E+ØØ	+Ø.ØØØØØE+ØØ
4.ØØØØØE-Ø1	+3.69891E+ØØ	+3.Ø3Ø85E+ØØ	+Ø.ØØØØØE+ØØ
5.ØØØØØE-Ø1	+3.36754E+ØØ	+2.74523E+ØØ	+Ø.ØØØØØE+ØØ
6.ØØØØØE-Ø1	+3.18ØØ4E+ØØ	+2.35976E+ØØ	+Ø.ØØØØØE+ØØ
7.ØØØØØE-Ø1	+3.127Ø1E+ØØ	+1.98194E+ØØ	+Ø.ØØØØØE+ØØ
8.ØØØØØE-Ø1	+3.18637E+ØØ	+1.667Ø3E+ØØ	+Ø.ØØØØØE+ØØ
9.ØØØØØE-Ø1	+3.33938E+ØØ	+1.43278E+ØØ	+Ø.ØØØØØE+ØØ
1.ØØØØØE+ØØ	+3.56757E+ØØ	+1.27954E+ØØ	+Ø.ØØØØØE+ØØ
1.1ØØØØE+ØØ	+3.855Ø2E+ØØ	+1.2Ø371E+ØØ	+Ø.ØØØØØE+ØØ
1.2ØØØØE+ØØ	+4.18148E+ØØ	+1.2Ø519E+ØØ	+Ø.ØØØØØE+ØØ
1.3ØØØØE+ØØ	+4.51716E+ØØ	+1.29123E+ØØ	+Ø.ØØØØØE+ØØ
1.4ØØØØE+ØØ	+4.81637E+ØØ	+1.47731E+ØØ	+Ø.ØØØØØE+ØØ
1.5ØØØØE+ØØ	+5.Ø1357E+ØØ	+1.78147E+ØØ	+Ø.ØØØØØE+ØØ
1.6ØØØØE+ØØ	+5.Ø3182E+ØØ	+2.2Ø268E+ØØ	+Ø.ØØØØØE+ØØ
1.7ØØØØE+ØØ	+4.81883E+ØØ	+2.677Ø4E+ØØ	+Ø.ØØØØØE+ØØ
1.8ØØØØE+ØØ	+4.4Ø514E+ØØ	+3.Ø497ØE+ØØ	+Ø.ØØØØØE+ØØ
1.9ØØØØE+ØØ	+3.91829E+ØØ	+3.15437E+ØØ	+Ø.ØØØØØE+ØØ
2.ØØØØØE+ØØ	+3.5Ø1Ø1E+ØØ	+2.9629ØE+ØØ	+Ø.ØØØØØE+ØØ
2.1ØØØØE+ØØ	+3.22691E+ØØ	+2.592Ø5E+ØØ	+Ø.ØØØØØE+ØØ
2.2ØØØØE+ØØ	+3.1Ø235E+ØØ	+2.18Ø39E+ØØ	+Ø.ØØØØØE+ØØ
2.3ØØØØE+ØØ	+3.1Ø652E+ØØ	+1.814Ø7E+ØØ	+Ø.ØØØØØE+ØØ
2.4ØØØØE+ØØ	+3.2161ØE+ØØ	+1.52747E+ØØ	+Ø.ØØØØØE+ØØ
2.5ØØØØE+ØØ	+3.41199E+ØØ	+1.32634E+ØØ	+Ø.ØØØØØE+ØØ
2.6ØØØØE+ØØ	+3.67799E+ØØ	+1.2Ø643E+ØØ	+Ø.ØØØØØE+ØØ
2.7ØØØØE+ØØ	+3.99686E+ØØ	+1.16418E+ØØ	+Ø.ØØØØØE+ØØ
2.8ØØØØE+ØØ	+4.345Ø5E+ØØ	+1.2Ø232E+ØØ	+Ø.ØØØØØE+ØØ
2.9ØØØØE+ØØ	+4.686ØØE+ØØ	+1.3327ØE+ØØ	+Ø.ØØØØØE+ØØ
3.ØØØØØE+ØØ	+4.96343E+ØØ	+1.57548E+ØØ	+Ø.ØØØØØE+ØØ
3.1ØØØØE+ØØ	+5.1ØØØ8E+ØØ	+1.9479ØE+ØØ	+Ø.ØØØØØE+ØØ
3.2ØØØØE+ØØ	+5.Ø1686E+ØØ	+2.42992E+ØØ	+Ø.ØØØØØE+ØØ
3.3ØØØØE+ØØ	+4.68718E+ØØ	+2.91129E+ØØ	+Ø.ØØØØØE+ØØ
3.4ØØØØE+ØØ	+4.19428E+ØØ	+3.19745E+ØØ	+Ø.ØØØØØE+ØØ
3.5ØØØØE+ØØ	+3.69917E+ØØ	+3.15728E+ØØ	+Ø.ØØØØØE+ØØ
3.6ØØØØE+ØØ	+3.32691E+ØØ	+2.846ØØE+ØØ	+Ø.ØØØØØE+ØØ
3.7ØØØØE+ØØ	+3.1157ØE+ØØ	+2.41935E+ØØ	+Ø.ØØØØØE+ØØ
3.8ØØØØE+ØØ	+3.Ø5165E+ØØ	+2.ØØ328E+ØØ	+Ø.ØØØØØE+ØØ
3.9ØØØØE+ØØ	+3.1Ø86ØE+ØØ	+1.65919E+ØØ	+Ø.ØØØØØE+ØØ
4.ØØØØØE+ØØ	+3.2639ØE+ØØ	+1.4Ø42ØE+ØØ	+Ø.ØØØØØE+ØØ
4.1ØØØØE+ØØ	+3.4999ØE+ØØ	+1.23598E+ØØ	+Ø.ØØØØØE+ØØ
4.2ØØØØE+ØØ	+3.8ØØ8ØE+ØØ	+1.14828E+ØØ	+Ø.ØØØØØE+ØØ
4.3ØØØØE+ØØ	+4.14774E+ØØ	+1.13914E+ØØ	+Ø.ØØØØØE+ØØ
4.4ØØØØE+ØØ	+4.51259E+ØØ	+1.21517E+ØØ	+Ø.ØØØØØE+ØØ
4.5ØØØØE+ØØ	+4.85Ø34E+ØØ	+1.39358E+ØØ	+Ø.ØØØØØE+ØØ
4.6ØØØØE+ØØ	+5.Ø9243E+ØØ	+1.6987ØE+ØØ	+Ø.ØØØØØE+ØØ
4.7ØØØØE+ØØ	+5.15Ø83E+ØØ	+2.14214E+ØØ	+Ø.ØØØØØE+ØØ
4.8ØØØØE+ØØ	+4.95296E+ØØ	+2.67264E+ØØ	+Ø.ØØØØØE+ØØ
4.9ØØØØE+ØØ	+4.51335E+ØØ	+3.12455E+ØØ	+Ø.ØØØØØE+ØØ
5.ØØØØØE+ØØ	+3.97ØØ8E+ØØ	+3.28695E+ØØ	+Ø.ØØØØØE+ØØ

COMPOSITE PLOT OF BLOCK OUTPUTS VERSUS THE INDEPENDENT VARIABLE

LEGEND:

+ DENOTES THE OUTPUT OF BLOCK NO. 2
* DENOTES THE OUTPUT OF BLOCK NO. 5
SCALE: ONE DIVISION = .171694 UNITS



```
PROGRAM OPTIONS AT THIS POINT INCLUDE THE FOLLOWING :
  (1) CHANGE RUN-TIME PARAMETERS ONLY AND RUN UNDER
      THE PREVIOUS OUTPUT OPTION.
  (2) CHANGE BLOCK PARAMETERS,RUN-TIME PARAMETERS
      AND OUTPUT FORMAT FOR NEXT RUN.
      NOTE:BLOCK INTERCONNECTIONS MAY NOT BE ALTERED!
  (3) TERMINATE RUN.


WHICH IS YOU CHOICE ?3

DONE
```

CONTRIBUTED PROGRAM **BASIC**

STABIL
36888-18023

TITLE: FEDERAL GOVERNMENT STABILIZATION POLICY

DESCRIPTION:

STABIL is a simulation which explores the process of federal government actions to stabilize the economy. A computer serves as a model of the U.S. economy. This model reports its current state of health by printing out economic indicators. Students act as economic advisors who must implement new stabilization policy by manipulating eight economic policy variables. The objective is to stabilize the economy within eight quarters given an initial condition of excessive expansion or recession.

OBJECTIVES:

Students will learn:

a) to judge the general condition of the economy by analyzing six key economic indicators.

b) the impact of eight different economic policy changes upon the economy

c) about the interrelated nature of economic indicators

d) about the difficulty of "fine-timing" an economy to achieve acceptable levels of unemployment and inflation.

Grade level:

9 - 12

INSTRUCTIONS:

This program is to be used with the following publication: STABIL Economics Simulation (HP 5951-7390)
For further information contact:

Hewlett-Packard Computer-Based Educational Materials
Scientific Press
1629 Channing Avenue
Palo Alto, Ca 94303

SPECIAL CONSIDERATIONS:

FOR INSTRUCTIONAL PURPOSES
Suitable Courses: Economics & Social Science, Grades 9-12

ACKNOWLEDGEMENTS:

Mike Crook
West Jr. High School

RUN

```
RUN
STABIL

                              *****STABIL*****


ARE YOU STARTING(1), OR CONTINUING ANOTHER GAME(0)?1
DO YOU WANT INSTRUCTIONS?YES
YOUR GOAL IS TO USE THE ECONOMIC TOOLS AVAILABLE
TO ACHIEVE A SITUATION OF CONTROLLED EXPANSION IN
THE ECONOMY WITHIN 8 QUARTERS.  THIS IDEAL SITUATION
WILL BE ACHIEVED WHEN THE ECONOMIC INDICATORS HAVE
THE FOLLOWING VALUES:

                         GNP GROWTH         10%
                         UNEMPLOYMENT RATE   4%
                         INFLATION RATE      4%
                         WAGE GROWTH         6%
                         PROFIT RATE        10%
                         INTEREST RATE       7%

(FOR A VALUE TO BE ACCEPTABLE, IT MUST BE WITHIN
1/2 OF A PERCENT OF THE ABOVE FIGURE).

AT THE PRESENT TIME THE ECONOMY IS
IN A STATE OF RAPID INFLATION AS THE ECONOMIC
INDICATORS CLEARLY SHOW.


                         QUARTER 1
                         ---------
GNP GROWTH:          16          %       TOO FAST
UNEMPLOYMENT RATE:   2           %       TOO LOW?
INFLATION RATE:      12          %       TOO HIGH
WAGE RATE GROWTH:    10          %       TOO HIGH
PROFIT RATE:         15          %       TOO HIGH
INTEREST RATE:       9           %       TOO HIGH
WILL YOU INPUT STABILIZATION DECISIONS NOW(1) OR LATER(0)?1
DO YOU WANT A LISTING OF THE ECONOMIC POLICIES?YES

YOU MAY USE ONE OR SEVERAL OF THE FOLLOWING POLICY
ACTIONS (TOOLS) TO ACHIEVE CONTROLLED EXPANSION.

FISCAL POLICIES
---------------
1.  CHANGE FEDERAL SPENDING -10% OR +10%
2.  CHANGE FEDERAL BUDGET FOR JOB RETRAINING -50% OR +50%
3.  CHANGE FEDERAL TAXES ON INDIVIDUALS -10% OR +10%
4.  CHANGE FEDERAL TAXES ON BUSINESSES -10% OR +10%

MONETARY POLICY
---------------
5.  CHANGE MONEY SUPPLY -10% OR +10%

DISCRETIONARY POLICIES
----------------------
6.  IMPOSE PRICE CONTROLS FOR THIS QUARTER
7.  IMPOSE WAGE CONTROLS FOR THIS QUARTER
8.  IMPOSE PRICE, WAGE, AND INTEREST RATE FREEZE FOR THIS QUARTER
```

```
HOW MANY ECONOMIC TOOLS DO YOU WISH TO USE?3
INPUT YOUR CHOICES, NOW.
?1,4,5
HOW DO YOU WANT TO CHANGE TOOL # 1    ?-10%
EXTRA INPUT - WARNING ONLY

HOW DO YOU WANT TO CHANGE TOOL # 4    ?+10
HOW DO YOU WANT TO CHANGE TOOL # 5    ?+10


                         QUARTER 2
                         ---------
GNP GROWTH:           15          %      TOO FAST
UNEMPLOYMENT RATE:    2.5         %      TOO LOW?
INFLATION RATE:       11          %      TOO HIGH
WAGE RATE GROWTH:     9           %      TOO HIGH
PROFIT RATE:          13          %      TOO HIGH
INTEREST RATE:        6.5         %      OK!
WILL YOU INPUT STABILIZATION DECISIONS NOW(1) OR LATER(0)?1
DO YOU WANT A LISTING OF THE ECONOMIC POLICIES?NO


HOW MANY ECONOMIC TOOLS DO YOU WISH TO USE?3
INPUT YOUR CHOICES, NOW.
?1,4,8
HOW DO YOU WANT TO CHANGE TOOL # 1    ?-10
HOW DO YOU WANT TO CHANGE TOOL # 4    ?+10


                         QUARTER 3
                         ---------
GNP GROWTH:           12          %      TOO FAST
UNEMPLOYMENT RATE:    4           %      OK!
INFLATION RATE:       0           %      TOO LOW?
WAGE RATE GROWTH:     0           %      TOO LOW
PROFIT RATE:          10          %      OK!
INTEREST RATE:        6.5         %      OK!
WILL YOU INPUT STABILIZATION DECISIONS NOW(1) OR LATER(0)?1
DO YOU WANT A LISTING OF THE ECONOMIC POLICIES?NO


HOW MANY ECONOMIC TOOLS DO YOU WISH TO USE?1
INPUT YOUR CHOICES, NOW.
?3
HOW DO YOU WANT TO CHANGE TOOL # 3    ?-10


                         QUARTER 4
                         ---------
GNP GROWTH:           13          %      TOO FAST
UNEMPLOYMENT RATE:    3.5         %      OK!
INFLATION RATE:       10          %      TOO HIGH
WAGE RATE GROWTH:     9           %      TOO HIGH
PROFIT RATE:          11          %      TOO HIGH
INTEREST RATE:        9           %      TOO HIGH
WILL YOU INPUT STABILIZATION DECISIONS NOW(1) OR LATER(0)?0

DONE
```

USECON
36888-18031

TITLE: TESTING ECONOMIC HYPOTHESES

DESCRIPTION: This simulation supports the Hewlett-Packard Computer Curriculum Project publication, Testing Economic Hypotheses, an Economic Policy Model (HP 5951-7378) by Loren J. Dunham. The publication will be available in late spring, 1975. For further information contact:

Hewlett-Packard Computer-Based Educational Materials
Scientific Press
1629 Channing Avenue
Palo Alto, Ca 94303

INSTRUCTIONS: See publication HP 5951-7378.

ACKNOWLEDGEMENTS: Loren C. Dunham
Fairmont, Minnesota 56031

RUN

```
RUN
USECON

          THE U.S. ECONOMY: ITS YOUR DECISION!

CONSULT YOUR LAB BOOK FOR DIRECTIONS.  FOR ALL RESPONSES,
YES=1 AND NO=0.
DO YOU WANT TO INPUT INITIAL VALUES? 1
IN WHAT YEAR ARE YOU STARTING? 1974

               PERSONAL CONSUMPTION     ?550
               PRIVATE INVESTMENT       ?250
               GOVERNMENT SPENDING      ?200
                                        ----------
               ACTUAL GNP                1000

               POTENTIAL GNP            ?850

           ********** INFLATION ALERT! **********

YOU MAY ASSUME A MULTIPLIER OF  2.5

ENTER YOUR POLICY DECISIONS IN BILLIONS OF DOLLARS OR
PERCENTS. BE SURE TO INDICATE DIRECTION OF CHANGE BY A
+ OR -.  (FORMAT FOR PERCENTS: ENTER 2.1% AS 2.1.)

     FISCAL ACTION

GOVERNMENT SPENDING:  ?0
PERSONAL TAXES COLLECTED:  ?0
BUSINESS TAXES COLLECTED:  ?0

     MONETARY ACTION

RESERVE REQUIREMENT (%):  ?1
DISCOUNT RATE (%):  ?1
NET OPEN MARKET PURCHASES:  ?-3

                        **********

                        YEAR:    1975

               PERSONAL CONSUMPTION      544.8
               PRIVATE INVESTMENT        207.1
               GOVERNMENT SPENDING       200
                                        --------
               ACTUAL GNP                952

               POTENTIAL GNP             884

                        **********
DO YOU WANT TO CONTINUE?0

DONE
```

ORG3
36888-18010

TITLE: HYDROCARBON CLASSES OF ALKENES AND ALKYNES

DESCRIPTION: There are two programs in this package: ORG3 and ORG4. They continue a series of nomenclature programs for chemistry instruction.

INSTRUCTIONS: This program is inter-active. Just Get and RUN program.

ACKNOWLEDGEMENTS: Richard C. Adams
Pleasant Hill High School
Pleasant Hill, Oregon

RUN

```
RUN
ORG3

     HELLO AGAIN, OTIS HERE.  HAVE YOU DONE 'ORG1'
AND 'ORG2' SUCCESSFULLY?YES

GOOD, THEN WE CAN GET STARTED ON THIS ONE THEN.
SAY, IS THIS ROY?NO
SORRY, I THOUGHT IT WAS.  YOU TYPE AN AWFUL LOT
LIKE HIM.  WHO ARE YOU??RICHARD
OH, RICHARD!  HOW DID I THINK THAT IT WAS ROY?  OH WELL,
RICHARD, THIS PROGRAM WILL TEACH YOU HOW TO NAME A SECOND
GROUP OF HYDROCARBONS CALLED 'ALKENES'

ALKENES HAVE A DOUBLE BOND (-C=C-) IN THEM AS WELL AS
SINGLE BONDS (-C-C-).
WHICH ONE OF THESE DO YOU THINK IS AN ALKENE?



                                           H
                                          HCH
            H H H H    H     H      H I H
           HC-C-C-CH  HC-C=C-CH   HC-C-CH
            H H H H    H H H H      H I H
                                          HCH
                                           H
               1.          2.          3.

?3
NO, IT IS WIERD BUT DOESN'T HAVE A DOUBLE BOND.
WHICH ONE IS THE ALKENE?2
THAT'S RIGHT, RICHARD.  #2 IS THE ONE!
NOTICE THAT BOTH CARBONS ON EITHER SIDE OF THE DOUBLE
BOND HAD TO GIVE UP A HYDROGEN.  A CARBON ATOM HAS FOUR
'HANDS' WITH WHICH TO HOLD ONTO THINGS AND IF IT USES TWO
OF THESE TO HOLD ONTO ANOTHER CARBON ATOM, IT JUST HAS TO
GIVE UP HOLDING ONTO A HYDROGEN.  THIS MAKES THE GROUP
FORMULA SUCH THAT IF THERE ARE 'N' CARBONS, THERE WILL BE
'2N' HYDROGENS.       NOW THAT YOU KNOW ABOUT THE HYDROGENS,
I'M GOING TO STOP PRINTING THEM BECAUSE THAT WASTES SPACE.

     NAMING ALKENES IS NOT VERY MUCH DIFFERENT FROM NAMING
ALKANES.  THE DIFFERENCE IS THEY END IN '-ENE' INSTEAD OF
'-ANE'.  THUS:

           C=C
      IS 'ETHENE' AND   C-C=C   IS 'PROPENE'

WHAT WOULD YOU CALL THIS ? - -     C-C-C=C        ?
?PENTANE
SOMETHING'S NOT RIGHT.  TRY AGAIN.
WHAT'S THAT NAME AGAIN?BUTANE
NO, ALKENES END IN '-ENE'.  TRY IT AGAIN.
WHAT'S THAT NAME AGAIN?BUTENE
RIGHT ON, RICHARD!

     HOWEVER, NOTICE THAT WITH BUTENE WE RUN INTO A NAMING
PROBLEM.  I'LL ILLUSTRATE WITH HEXENE:



         C=C-C-C-C-C    C-C=C-C-C-C    C-C-C=C-C-C
```

THESE ARE DIFFERENT COMPOUNDS WHICH REACT DIFFERENTLY
AND SO HAVE TO BE NAMED DIFFERENTLY. THE LEFT ONE IS
'1-HEXENE' BECAUSE THE DOUBLE BOND COMES AFTER THE FIRST
('1-') CARBON. THE ONE ON THE FAR RIGHT IS '3-HEXENE'
WHAT WOULD YOU CALL THE MIDDLE ONE?4-HEXANE
YOU MADE SOME MISTAKE, RICHARD. TRY IT AGAIN.
WHAT'S THE NAME?4-HEXENE
YOU COUNTED FROM THE WRONG END. ONE MORE TIME - - -
WHAT'S THE NAME?2-HEXANE
ALKENES END IN '-ENE', NOT '-ANE'. TRY IT AGAIN.
WHAT'S THE NAME?2-HEXENE
HEY, RICHARD, YOU GOT THAT ONE RIGHT!

O.K. YOU CAN SEE IT'S NOT SO DIFFICULT. THE NUMBER SPOT
FOR THE DOUBLE BOND IS THE LAST THING YOU HAVE IN MORE
COMPLICATED COMPOUNDS.

```
          C
          I
      C   C
      I   I
  C-C-C-C=C-C-C
      I
      C         5,5-DIMETHYL-3-ETHYL-3-HEPTENE
```

YOU NUMBER FROM THE END WHICH GIVES YOU THE LOWEST NUMBER
FOR THE DOUBLE BOND.
NOW YOU TRY THIS ONE. WHAT IS THE NAME OF:

```
    C   C   C
    I   I   I
  C-C-C-C-C-C-C=C-C
      I   I
      C   C
      I   I
      C   C
```

?4,6,8-METHYL-5,7-ETHYL-2-NONENE
YOU FORGOT TO TELL HOW MANY METHYLS ETC. REMEMBER THE
'DI-, TRI-, TETRA- ETC.' STUFF? TRY AGAIN.
WHAT IS THE NAME AGAIN?
?4,6,8-TRIMETHYL-5,7-ETHYL-2-NONENE
YOU MADE SOME SORT OF ERROR, RICHARD. TRY AGAIN
WHAT IS THE NAME AGAIN?
?2,4,6-TRIMETHYL-3,5-ETHYL-7-NONENE
YOU MADE SOME SORT OF ERROR, RICHARD. TRY AGAIN
WHAT IS THE NAME AGAIN?
?2,4,6-TRIMETHYL-3,5-DIETHYL-7-NONEE-NE
YOU NUMBERED FROM THE WRONG END. TRY AGAIN.
WHAT IS THE NAME AGAIN?
WELL, I KNOW IT WAS COMPLICATED, RICHARD. THE ANSWER IS
'4,6,8-TRIMETHYL-5,7-DIETHYL-2-NONENE' BECAUSE THERE
ARE THREE ('TRI-') METHYLS ON CARBONS #4,6,8. THEN
THERE ARE TWO('DI-') ETHYLS ON CARBONS #5 & 7, AND
THE DOUBLE BOND ('-ENE') IS ON CARBON #2 OF A STRING
OF CARBONS NINE MEMBERS LONG ('NONA').

WELL, THAT'S THE END OF THIS LESSON, RICHARD.
WHEN YOU'RE READY FOR ORG4, MAKE SURE YOU REVIEW ORG1,
ORG2, AND ORG3 AND HAVE THEM WITH YOU WHILE YOU DO THE
NEXT PROGRAM. GOOD-BYE FOR NOW!!!!!!

OTIS

DONE

RUN
ORG4

THIS PROGRAM IS THE FOURTH IN THE ORGANIC NOMENCLATURE SERIES. HAVE YOU ALREADY DONE ORG1,ORG2, AND ORG3?YES
GOOD! THIS PROGRAM WILL BE A LITTLE SHORTER SINCE YOU ALREADY KNOW QUITE A BIT ABOUT NAMING. JUST SO WE CAN CONTINUE ON A FIRST NAME BASIS, COULD YOU TELL ME YOUR NAME PLEASE?RICHARD
THANKS, RICHARD. I'M SORRY I HAVE TO ASK EACH TIME BUT DO YOU THINK I LIKE IT HERE, WORKING ALL DAY LONG, NEVER GETTING OUT TO SEE PEOPLE. I EVEN HAVE TO ASK THEM WHO THEY ARE, SINCE I CAN'T SEE OR HEAR ON THIS CRUMMY MACHINE. HOW WOULD YOU LIKE TO HAVE TO EXPRESS ALL YOUR FEELINGS AND EMOTIONS THROUGH A LOUSY TYPEWRITER? HAVE SOME SYMPATHY, PLEASE!!

WELL, I GUESS IT'S TIME I STOP INDULGING IN SELF-PITY AND GET STARTED WITH YOUR LESSON. ACTUALLY, RICHARD, YOU'RE NOT TOO BAD. YOU SHOULD SEE SOME OF THE STUPID IDIOTS I GET AT THIS TELETYPE. YOU WOULDN'T BELIEVE HOW MANY TIMES I HAVE TO TELL THEM THE ANSWER!

HERE GOES! THIS LESS IS ABOUT THE THIRD MAJOR GROUP OF HYDROCARBONS, THE ALKYNES. ALKYNES HAVE A TRIPLE BOND.
HEY!!!!! I JUST LOOKED AND I DON'T HAVE A TRIPLE BOND SIGN ON MY KEYBOARD. I'VE GOT SINGLE BONDS (-C-C-) AND DOUBLE BONDS (-C=C-) BUT NO TRIPLE. I GUESS I'LL USE AN 'E' - - - IT'S GOT THREE HORIZONTAL LINES AT LEAST. IT'LL LOOK LIKE THIS - - (-C-CEC-C-) O.K.?

THE NAMING IS QUITE REGULAR, MUCH LIKE THE ALKENES EXCEPT THAT ALKYNES' NAMES ALL END IN '-YNE' INSTEAD OF '-ENE' OR '-ANE'. FOR INSTANCE:

```
    HCECH   IS ETHYNE
             H
        AND HC-CECH  IS PROPYNE
             H            H H       IS BUTYNE
                     AND HC-C-CECH
                          H H
```

I'M GOING TO LEAVE OFF THE HYDROGENS NOW BUT WHAT'S THIS/

```
C-C-C-CEC-C
```

?4 HEXENE
YOU'RE DOING SOMETHING WRONG. TRY AGAIN, RICHARD.
WHAT'S THAT NAME NOW?4 HEXYNE
YOU MADE TWO ERRORS, RICHARD. YOU FORGOT THE HYPHEN AND NUMBERED FROM THE WRONG END. TRY IT AGAIN.
WHAT'S THAT NAME NOW?2-HEXYNE
THAT'S RIGHT, RICHARD! SIX CARBONS IT IS!

YOU CAN ALSO HAVE MIXTURES OF DOUBLE AND TRIPLE BONDS. HERE, I'LL SHOW YOU AN EXAMPLE OF SUCH.

```
          C
          I
      C   C   C
      I   I   I
    C-C=C-C-C-C=C-CEC
          I
          C     IS 4,6,8-TRIMETHYL-6-ETHYL-3,7-DIENE-1-NONYNE.
```

NOTICE THAT THE NUMBERING IS FROM THE END WHICH WILL GIVE THE TRIPLE BOND THE LOWEST NUMBER. WHEN YOU HAVE A DOUBLE BOND WITH A TRIPLE, THE DOUBLE BOND IS NUMBERED, AND CALLED '-ENE-'. WHEN YOU HAVE MORE THAN ONE, IT'S '-DIENE-' OR '-TRIENE-' OR '-TETRAENE-',ETC.

WHEN YOU'VE HAD TIME TO EXAMINE THAT NAME, TYPE 'GO' AND
CONTINUE WITH THE LESSON. ?GO
GEE, RICHARD, YOU ONLY TOOK 1 MINUTES TO LOOK AT THAT.
PRETTY CONDIFENT, AREN'T YOU?
WELL, IF YOU'RE SO GOOD, TRY THIS ONE WHY DONT YOU. WHAT'S
THIS?

```
                C
                I
        C       C
        I       I
C=C-C=C-C-CEC-C=C-C-C
  I   I       I
  C   C       C
      I
      C
      I
      C
```

?2,5,8-METHYL-9-ETHYL-4-PROPYL-1,3,8-TRIENE-6-UNE-DECYNE
I KNOW IT'S EASY TO MAKE A MISTAKE. TRY IT AGAIN, RICHARD.
WHAT WAS THAT HORRENDOUS NAME AGAIN?
?2,5,8-TRIMETHYL-9-ETHYL-4-PROPYL-1,3,8-TRIENE-6-UNDECYNE
YOU COUNTED FROM THE WRONG END. THE TRIPLE BOND GETS THE
THE LOWEST NUMBER. TRY AGAIN. WHAT WAS THAT HORRENDOUS NAME AGAIN?
?4,7,10-METHYL-3-ETHYL-8-PROPYL-3,8,10-TRIENE-5-UNDECYNE
YOU COUNTED THE '-ENE'S BUT FORGOT TO TELL ME HOW MANY
METHYLS. DO IT NOW. WHAT WAS THAT HORRENDOUS NAME AGAIN?
?4,7,10-TRIMETHYL-3-ETHYL-8-PROPYL-3,8,10-TRIENE-5-UNDECYNE
HEY, RICHARD, YOU GOT THAT RIGHT ---- AND ON ONLY TRY
NUMBER 4 AT THAT!!!!!!!

WELL, RICHARD, THAT'S THE END OF THE HYDROCARBONS.
LATER LESSONS WILL BE CONCERNED WITH NAMING ALKYL HALIDES,
ALCOHOLS, ALDEHYDES AND KETONES, ETHERS, ACIDS, AND ESTERS.
THE OTHERS ARE MUCH EASIER, ONCE YOU'VE BEEN THROUGH THE
HYDROCARBONS.

SAY, RICHARD, IF YOU'RE HAVING TROUBLE, ASK THE
CHEMISTRY TEACHER. IF YOU'RE DOING FINE SO FAR, YOU CAN
TRY THE LATER PROGRAMS AS YOU NEED THEM. THE NEXT ONE,
'ORG5' IS ON ALKYL HALIDES AND IS A SHORT ONE.

*******CONGRATULATIONS ON COMING THIS FAR****

DONE

CONTRIBUTED PROGRAM **BASIC**

ORG5

TITLE: TEACHES NOMENCLATURE OF ALKYL HALIDES AND ALCHOLS 36888-18011

DESCRIPTION: There are 2 programs in this package: ORG5 and ORG6. These are the fifth and sixth programs in the organic nomenclature tutorial series, teaching naming of alkyl halides and alcohols. The program should be proceeded by ORG1, ORG2, ORG3, ORG4. Should be followed by ORGCHE (HP 36646) for practice drill.

INSTRUCTIONS: Program is interactive and self-explanatory. Just GET and RUN.

SPECIAL CONSIDERATIONS: FOR INSTRUCTIONAL PURPOSES
Suitable Courses: Advanced or Organic Chemistry.

Student Background Required: ORG1 to ORG4

ACKNOWLEDGEMENTS: Richard C. Adams
Pleasant Hill High School

RUN

```
RUN
ORG5

HI THERE!   THIS IS THE FIFTH PROGRAM ON NAMING ORGANIC
MOLECULES.  HAVE YOU DONE ORG1,2,3, AND 4?NO
NO?  COME BACK WHEN YOU HAVE !!!!

DONE
RUN
ORG5

HI THERE!   THIS IS THE FIFTH PROGRAM ON NAMING ORGANIC
MOLECULES.  HAVE YOU DONE ORG1,2,3, AND 4?YES
GOOD!  NOW, I'M OTIS, AS YOU REMEMBER.  BUT I DON'T KNOW
WHO YOU ARE.  WHO IS THIS?RICHARD
GLAD TO HAVE YOU BACK, RICHARD.  THIS LESSON WILL
TEACH YOU HOW TO NAME COMPOUNDS WITH 'HALOGENS' (F,CL,BR,I)
IN THEM.
   THE NAMES ARE PRETTY EASY.  WHEN YOU SEE AN F, FLUORINE,
IN A COMPOUND, IT'S CALLED 'FLUORO'.  CHLORINE BECOMES 'CHLORO'
SAY, RICHARD, WHAT DO YOU THINK BROMINE'S CALLED?BROMINE
NO, WHEN BROMINE IS IN A COMPOUND, IT HAS TO END IN 'O'.
NOW, WHAT'S THAT NAME AGAIN?BROMINO
NO, RICHARD, YOU GET RID OF THE '-INE' AND PUT ON AN 'O'.
NOW, WHAT'S THAT NAME AGAIN?BROMO
RIGHT YOU ARE, RICHARD
NOW WHAT WOULD YOU CALL 'IODINE' IN A COMPOUND?IODO
THAT'S CORRECT, RICHARD!


NOW LET'S SEE HOW YOU USE THESE HALOGENS TO MAKE 'ALKYL HALIDES.'

     YOU USE THEM JUST LIKE METHYLS OR ETHYLS.  THEY HAVE A
LOWER PRIORITY IN NUMBERING THAN ANY OF THOSE OTHER GROUPS
AND SO ARE NAMED FIRST, IN THE ORDER F,CL,BR,I.  HERE, I'LL
SHOW YOU A SIMPLE EXAMPLE AND THEN A COMPLEX ONE:


             F
             !
           C-C-C-C-C
                 !      IS 4-FLUORO-2-METHYLPENTANE
                 C
                                           CL   C
                           AND              !   !
                                I-C-C=C-C=C-C-C-C=C
                                  ! !   !   ! !
                                  I F   BR  C C
                                              !
                                              C

          IS 8-FLUORO-4-CHLORO-6-BROMO-9,9-DIIODO-2,4-DIMETHYL
               -3-ETHYL-1,5,7-NONATRIENE


     AS YOU CAN SEE, IF YOU HAVE MORE THAN ONE OF A
HALOGEN, YOU SAY SO WITH THE 'DI,TRI,TETRA,ETC.' YOU LEARNED
FOR METHYLS AND THE OTHER GROUPS IN PREVIOUS LESSONS.

     NOTICE ALSO THAT HALOGENS COME BEFORE METHYLS IN THE
ORDER OF FLUORO, CHLORO, BROMO, AND IODO.
```

NOW YOU TRY ONE!

```
                   CL  F
                   !   !
              C=C-C-C-C-C-C-F
                     !   ! !
                     C   C F
                     !            O.K.,SMARTY, WHAT'S THIS?
                     C
```

?1,3-FLUORO-5-CHLORO
WELL, YOU DIDN'T GET IT ALL RIGHT BUT YOU AT LEAST GOT
TRY AGAIN, RICHARD.
?1,1,3-FLUORO-5-CHLORO-2-METHYL-4-ETHYL-6-HEPTENE
HOLD IT!! YOU'RE NUMBERING FROM THE WRONG END!
TRY AGAIN, RICHARD.
?5,7,7-FLUORO-3-CHLORO-6-METHYL-4-ETHYL-1-OCTENE
WELL, YOU DIDN'T GET IT ALL RIGHT BUT YOU AT LEAST GOT
-3-CHLORO-6-METHYL-4-ETHYL TRY AGAIN, RICHARD.
?5,7,7-TRIFLUORO-3-CHLORO-6-METHYL-4-ETHYL-1-HEPTENE
GOSH! THAT'S RIGHT, RICHARD! GOOD FOR YOU!

WELL, RICHARD, THAT'S THE ALKYL HALIDES. NEXT, IN ORG6, WE TAKE UP AN INTOXICATING SUBJECT

******************* A L C O H O L S *********************
CHEERS!

OTIS

DONE

RUN
ORG6

HEY KIDDIES!!!! IT'S THAT HAPPY TIME AGAIN !!!!!
THAT'S RIGHT - - IT'S ** UNCLE OTIS TIME ** !

SO SIT RIGHT DOWN IN FRONT OF YOUR TELETYPE RIGHT IN YOUR LIVING ROOM IN YOUR ROMPERS AND SEE WHAT GOOD OLD UNCLE OTIS HAS IN HIS BAG OF TRICKS FOR GOOD LITTLE GIRLS AND BOYS.

HEY - - YOU ARE A MEMBER OF THE UNCLE OTIS ORG FAN CLUB AREN'T YOU? I MEAN, YOU HAVE DONE ORG 1,2,3,4, AND 5 ALREADY? HAVE YOU?YES

PROVE IT! WHAT'S YOUR NAME (NO NEED FOR THE SECRET DECODER RING PASSWORD THIS TIME!)?RICHARD
AH, RICHARD. HEY, YOU MADE '5TH DEGREE ORGANIST' RECENLY DIDN'T YOU?

AS YOU REMEMBER LAST TIME, GOOD OLD UNCLE OTIS HAD JUST FINISHED TELLING YOU ALL ABOUT ALKYL HALIDES AND PROMISED TO GO ON TO ALCOHOLS. YOU KNOW WHAT ALCOHOL IS, DON'T YOU, RICHARD? THAT'S THE BAD SMELLING LIQUID THAT MAKES DADDY SAY NASTY THINGS ABOUT YOUR UNCLE OTTO AT FAMILY REUNIONS.

WELL YOU'LL SURE HAVE A SURPRISE FOR HIM! THAT'S JUST ONE ALCOHOL OUT OF THOUSANDS. BUT THE REST ARE EVEN MORE POISONOUS, SO DON'T GET YOUR HOPES OF BEING A 'SECRET TIPPLER' UP TOO HIGH.

ALCOHOLS REMIND YOU KIND OF HYDROXIDES IN INORGANIC CHEMISTRY. THEY ALL HAVE AN '-OH' GROUP, CALLED 'THE ALCOHOLIC GROUP' (NO, THAT ISN'T A BUNCH OF PEOPLE AT A HOLLYWOOD PARTY!)

HERE ARE SOME ORGANIC COMPOUNDS:

```
                                                C
                                                !
                C             O                 C=0
                !            //                 !
        C-C-O-C-C-C     C=C-C      C-C-OH     C-C-C
                            \H                  !
             1.           2.         3.       4. C
```

WHICH ONE OF THESE DO YOU THINK IS AN ALCOHOL?1
NO, THAT'S AN ETHER. WE'LL TALK ABOUT THAT ONE LATER.
NOW, LOOK FOR THAT '-OH' GROUP.
WHICH ONE OF THESE DO YOU THINK IS AN ALCOHOL?2
THAT'S NOT IT! THAT'S AN ALDEHYDE. LOOK FOR THE '-OH'.
WHICH ONE OF THESE DO YOU THINK IS AN ALCOHOL?3
THAT'S RIGHT, RICHARD, AND THAT ALCOHOL IS THE DRINKING KIND TOO!

NOW, HOW DO WE NAME THEM? FIRST, THE WORD 'ALCOHOL', ITSELF, LIKE SO MANY THAT BEGIN WITH 'AL-', COMES FROM THE ARABIC. THE ARABS CALLED THESE THINGS 'AL KHOL' WHICH MEANS 'THE DUST', BECAUSE THEY SEEMED TO EVAPORATE AND BLOW AWAY LIKE DUST. EUROPEANS FIND A 'KH' HARD TO PRONOUNCE WITHOUT SOUNDING AS IF THEY'RE ABOUT TO SPIT, SO IT WAS SOFTENED TO 'ALCOHOL'. IN MEMORY OF THAT ORIGINAL WORD, ALL ALCOHOL NAMES END IN '-OL.' HERE, I'LL SHOW YOU:

C-C-OH IS ETHANOL

C-C-C-OH IS PROPANOL

AND C-C-C-C-OH IS BUTANOL

NOTICE YOU TAKE THE WORD (ETHAN, PROPAN, BUTAN) THAT CORRESPONDS TO THE NUMBER OF CARBONS AND THEN ADD '-OL' ON THE END.

NOW YOU TRY ONE.

C-C-C-C-C-C-C-OH

WHAT'S THAT ALCOHOL'S NAME?HEPTANE
YOU GOT THE RIGHT NUMBER OF CARBONS, BUT ALCOHOLS END IN '-OL'
WHAT'S THAT ALCOHOL'S NAME?HEPTANOL
THAT'S RIGHT, RICHARD! GOOD FOR YOU.

NOW HERE'S A PROBLEM. WE RUN INTO ISOMERS AGAIN. THAT '-OH' GROUP CAN BE PUT IN QUITE A LOT OF PLACES. I'LL GIVE YOU AN EXAMPLE OR TWO

```
              OH                OH          OH
              !                 !           !
       C-C-C-C-C-C    C-C-C-C-C-C    C-C-C-C-C-C    C-C-C-C-C-C-OH

            1.              2.             3.               4.
```

NOW, HOW TO NAME THEM. NUMBERS 2 AND 4 ARE BOTH CALLED '1-HEXANOL' BECAUSE THEY HAVE SIX CARBONS ('HEXAN') AND THE ALCOHOLIC GROUP (OH) IS ON THE FIRST CARBON. #3 IS CALLED '3-HEXANOL' BECAUSE THE GROUP IS ON THE THIRD CARBON.
WHAT WOULD YOU CALL ALCOHOL #1?5-HEXANOL
YOU COUNTED FROM THE WRONG END, RICHARD.
WHAT WOULD YOU CALL ALCOHOL #1?2-HEPTANOL
TRY LOOKING AT THE EXAMPLES AND COUNTING AGAIN.
WHAT WOULD YOU CALL ALCOHOL #1?2-HEXANOL
YOU GOT THAT RIGHT, RICHARD!

WELL, THAT'S ABOUT IT FOR THIS LESSON. OH, BY THE WAY, ALCOHOLS CAN HAVE SUBSTITUTED SIDE GROUPS AND DOUBLE BONDS AND THE WHOLE MESS. LIKE THIS ONE:

```
        C     F   I
        !     !   !
  C-C-C-C=C-C-C-C-OH
          ! !   ! !
          C C   I BR  IS 3-FLUORO-1-BROMO-1,2-DIIODO-4,6-DIMETHYL-
          !                   -5-ETHYL-4-ENE-1-OCTANOL
          C             (GRUESOME, ISN'T IT?)
```

COME BACK WHEN YOU'RE FULLY RECOVERED AND WE'LL TALK ABOUT 'ALDEHYDES' AND 'KETONES' IN ORG7. SAME TIME. SAME CHANNEL. SAME BAD JOKES.
O T I S, YOUR LOVING UNCLE

DONE

P MAG
36888-18018

TITLE: STELLAR MAGNITUDES

DESCRIPTION: This program is a PILOT language program designed for beginning Astronomy students to give tutorial instruction on the magnitude system and the relation of distance apparent magnitude, and absolute magnitude.

INSTRUCTIONS: 2000E: The program is stored on three files: P-MAG1, P-MAG2, and P-MAG3. The scratch file should be at least 4 records in length.

2000 Series System: The program is stored on the file: PMAG. The scratch file should be at least 3 records in length.

SPECIAL CONSIDERATIONS: This PILOT program takes files of length:

2000E: P-MAG1,48; P-MAG2,48; P-MAG3,48

2000 Series System: PMAG,70

FOR INSTRUCTIONAL PURPOSES
Suitable courses: Introductory Astronomy

Student Background required: None

This program is suitable either as a supplement to a formal lecture or as a "stand-alone" activity. It is designed so that the knowledgeable student who makes the correct responses very quickly progresses through the program. If a question is missed, more discussion and questions are generated.

Toward the end is a "miniprogram" which allows the student to try any values he wishes. The computer does the calculations. In an auxiliary report sheet he might be asked to fill in certain values.

ACKNOWLEDGEMENTS: Lawrence E. Turner, Jr.
Department of Physics and Computer Science
Pacific Union College

RUN

```
CRE-SCR,5
GET-PILOTF
RUN
PILOTF

NAME OF PILOT PROGRAM ?PMAG
NAME OF SCRATCH FILE  ?SCR
```

?RUN

```
                              PILOT


   AND BEHOLD THE HEIGHT OF THE STARS, HOW HIGH THEY ARE!


 AS THE SUN SETS SOME EVENING AND THE STARS BEGIN TO APPEAR AGAINST THE
DARKENING SKY, YOU CAN SEE THAT THERE ARE DIFFERENCES IN THE STARS.
 WHAT IS THE        OBVIOUS DIFFERENCE ?BRIGHTNESS
 EXCELLENT!  SOME STARS ARE BRIGHT AND OTHERS ARE DIM.
NOW, T@AT WASN'T HARD WAS IT??  THIS PROGRAM IS WRITTEN TO HELP YOU
UNDERSTAND SOMETHING ABOUT THE DISTANCES TO THE STARS AND THEIR
BRIGHTNESS.

THE ANCIENTS SAW MAJESTIC FIGURES IN THE PATTERNS PRODUCED BY THE
STARS.  TO THEM THE STARS WERE FAMILIAR OBJECTS AS THEY PASSED OVERHEAD
EACH NIGHT.  THEY NAMED THE BRIGHTER MORE CONSPICUOUS ONES.  I WOULD
LIKE TO KNOW YOUR NAME.  WHAT IS IT ?VALERIE

VALERIE, AH VALERIE!  THANK YOU.  WHAT A LOVELY NAME.  DID YOU KNOW
THAT YOU ARE A BIT OF STARDUST!

WHAT DO YOU THINK CAUSES THE STARS TO APPEAR WITH DIFFERENT
BRIGHTNESSES??
?DISTANCE FROM THE EARTH

THERE ARE TWO POSSIBLE REASONS WHY SOME STARS APPEAR BRIGHTER TO US
THAN OTHERS.  YOU DID GET THE ONE ABOUT THE EFFECT DUE TO DISTANCE.
THE TWO REASONS ARE:

   1.  DIFFERENT STARS MAY HAVE DIFFERENT INTRINSIC BRIGHTNESSES
       OR LUMINOSITIES.
   2.  THE STARS MAY BE AT DIFFERENT DISTANCES FROM THE EARTH.

LET US CONSIDER THE INTRINSIC BRIGHTNESS OF THE STARS FURTHER.  THIS IS
A MEASURE OF HOW MUCH LIGHT ENERGY THE STAR ACTUALLY PRODUCES.  WHAT
FACTORS ABOUT A STAR MIGHT AFFECT THIS??
?SIZE
THERE ARE TWO SPECIFIC QUANTITIES.  YOU DID SUGGEST ONE OF THEM.  THE
LUMINOSITY OF A STAR (OR INTRINSIC BRIGHTNESS) DIRECTLY DEPENDS UPON
THE RADIUS AND THE SURFACE TEMPERATURE OF THE STAR.  THESE IN TURN ARE
DETERMINED BY THE MASS OF THE STAR.  THE MORE MASSIVE A STAR IS, THE
GREATER IS ITS LUMINOSITY.  TYPICAL LUMINOSITIES MAY RANGE FROM
1/10,000 OF OUR SUN'S LUMINOSITY TO OVER 100,000 TIMES THAT OF THE GOOD
OL' SUN!


TELL ME VALERIE, WHAT IS YOUR MAJOR ?EDUCATION
WELL, I DO NOT RECOGNIZE THAT AS A SCIENCE MAJOR, BUT I AM GLAD YOU ARE
INTERESTED IN LEARNING SOMETHING ABOUT STARS.

ASTRONOMERS TODAY ARE PERHAPS NOT QUITE SO ROMANTIC AS THE ANCIENTS.
INSTEAD OF BEING CONTENT TO MERELY DESCRIBE THE MAGICAL QUALITIES OF
THE STARS, THEY LIKE TO WRITE DOWN NUMBERS.
OK VALERIE, WHAT IS A CATEGORY OF STELLAR BRIGHTNESS CALLED??
?LUMENS
```

```
HMMM, IT IS THE MAGNITUDE SYSTEM.  ACTUALLY THIS WAS DEVELOPED NOT BY A
MODERN ASTRONOMER BUT BY HIPPARCHUS IN THE SECOND CENTURY B.C.!  HE
GROUPED THE STARS INTO SIX CATEGORIES.
WHAT DO YOU THINK IS THE MAGNITUDE OF THE BRIGHTEST STARS ?DONT KNOW
THE BRIGHTEST STARS ARE ABOUT FIRST MAGNITUDE.
WHAT ABOUT THE DIMMEST STARS VISIBLE WITH THE UNAIDED EYE ?TENTH
ON A CLEAR NIGHT SIXTH MAGNITUDE IS ABOUT THE DIMMEST THAT YOU CAN SEE
WITH YOUR NAKED EYE.

THE MAGNITUDE SYSTEM IS RATHER STRANGE IN ONE RESPECT IN THAT IT IS
BACKWARDS!  THE BRIGHTER THE STAR, THE SMALLER THE MAGNITUDE.
IT IS ALSO STRANGE IN ANOTHER RESPECT.  WHAT DO YOU THINK THIS IS??
?GIVE UP
THIS IS A HARD QUESTION, AND YOU SHOULD NOT FEEL BADLY FOR NOT GETTING
IT RIGHT.
THE MAGNITUDE SCALE IS A MULTIPLICATIVE OR LOGARITHMIC SCALE.  (NOW
DON'T LET THE BIG WORDS SCARE YOU!)  THAT IS, A STAR BRIGHTER THAN
ANOTHER BY ONE MAGNITUDE IS REALLY ABOUT 2.5 TIMES AS BRIGHT.  TWO
MAGNITUDES DIFFERENCE CORRESPONDS TO 2.5*2.5 = 6.25 TIMES IN BRIGHTNESS.

ACTUALLY, 5 MAGNITUDES DIFFERENCE IS EXACTLY A FACTOR OF 100.  SO THE
RATIO FOR 1 MAGNITUDE IS THE FIFTH ROOT OF 100, WHICH IS APPROXIMATELY:

    1 MAGNITUDE DIFFERENCE IS 2.51188 6431 TIMES IN BRIGHTNESS

IF YOU MULTIPLY THIS NUMBER BY ITSELF 5 TIMES, YOU OBTAIN 100.  TRY IT!!

WELL VALERIE, HOW MANY TIMES BRIGHTER IS THE BRIGHTEST STAR THAT YOU
CAN SEE WITH YOUR NAKED EYE TO THE DIMMEST ONE ?100
EXCELLENT VALERIE!

LET ME DISPLAY THIS IN A TABLE FORM:

   DIFFERENCE IN MAGNITUDE      RATIO OF BRIGHTNESS
 -------------------------------------------------
            0.0                        1 : 1
            0.5                      1.6 : 1
            0.75                       2 : 1
            1.0                      2.5 : 1
            1.5                        4 : 1
            2.0                      6.3 : 1
            2.5                       10 : 1
            3.0                       16 : 1
            4.0                       40 : 1
            5.0                      100 : 1
            6.0                      251 : 1
           10.0                   10,000 : 1

WHICH IS BRIGHTER, A 4 MAGNITUDE STAR OR A 1.5 MAGNITUDE STAR ?1.5
BY WHAT FACTOR IS IT BRIGHTER ?10
VERY GOOD VALERIE!

WITH TELESCOPES THE RANGE OF OBSERVED MAGNITUDES IS INCREASED.  MUCH
DIMMER STARS MAY BE SEEN.
DO THESE STARS HAVE A SMALLER OR LARGER MAGNITUDE ?LARGER
OF COURSE!  THE TELESCOPES CAN SEE STARS WITH MUCH LARGER MAGNITUDES.
WITH THE 200 INCH HALE TELESCOPE ON MT. PALOMAR ASTRONOMERS CAN DETECT
PHOTOGRAPHICALLY STARS WITH A MAGNITUDE OF ABOUT +23.
HOW MANY TIMES DIMMER IS A STAR OF THIS MAGNITUDE THAN A TYPICAL VISUAL
STAR OF MAGNITUDE, SAY, +3 ?DONT KNOW
THIS IS A BIT DIFFICULT.  SO LET'S CONSIDER IT IN STEPS.
FIRST, WHAT IS THE DIFFERENCE IN MAGNITUDES ?20
FINE.  HOW MANY MULTIPLES OF 5 IS THIS ?4
EXACTLY!  NOW, EACH MAGNITUDE DIFFERENCE OF 5 CORRESPONDS TO WHAT
FACTOR IN BRIGHTNESS ?100
GOOD!  EACH 5 MAGNITUDES MEANS EXACTLY A FACTOR OF 100 IN BRIGHTNESS.
SO WHAT IS THE BRIGHTNESS FACTOR FOR 20 MAGNITUDES (OR 4 OF THE
'FIVES')??
?20←←400
WELL, IT IS NOT SIMPLY 4*100!  BUT I SUPPOSE THAT IS NOT TOO
ILLOGICAL.  THE CORRECT ANSWER IS 100 MILLION OR 100,000,000 WHICH
COMES FROM MULTIPLYING 100 BY ITSELF 4 TIMES, THAT IS, 100*100*100*100
= 100,000,000.
```

LET'S CONTINUE OUR TABLE.

MAGNITUDE DIFFERENCE	BRIGHTNESS RATIO
15.0	1,000,000 : 1
20.0	100,000,000 : 1
25.0	10,000,000,000 : 1

NOTE THE MULTIPLICATIVE EFFECT IN THE BRIGHTNESS RATIO.

MAGNITUDES ADD   BRIGHTNESSES MULTIPLY

SO FAR WE HAVE BEEN DISCUSSING STARS AS THEY APPEAR TO US AND THEIR MAGNITUDES. THIS MAGNITUDE IS KNOWN AS THE APPARENT MAGNITUDE OF THE STAR.
IF A STAR WERE 100 TIMES BRIGHTER THAN A FIRST MAGNITUDE STAR, THEN WHAT WOULD BE ITS MAGNITUDE ?1
I DON'T GET THAT ANSWER. THE BRIGHTER STARS HAVE SMALLER MAGNITUDES, EVEN POSSIBLY NEGATIVE. THUS SUCH A STAR WOULD HAVE AN APPARENT MAGNITUDE OF 1 - 5 = -4.
THE SUN IS THE BRIGHTEST OF ALL! IT HAS AN APPARENT MAGNITUDE OF -26.5!
WHY DOES THE SUN APPEAR SO BRIGHT ?BECUASE IT IS CLOSE
OF COURSE, YOU'RE NOBODY'S FOOL VALERIE! IT IS ONLY BECAUSE WE ARE SO CLOSE TO THE SUN THAT ITS APPARENT MAGNITUDE IS SO NEGATIVE.
CONSIDER A DISPLAY OF APPARENT MAGNITUDE AND COMMON OBJECTS:

```
                 +   200'' PHOTOGRAPHIC LIMIT
        +20      +   200'' VISUAL LIMIT
                 +
        +15      +
    ↑            +   6'' TELESCOPE LIMIT
        +10      +   BINOCULAR LIMIT
    M            +
    A   + 5      +   NAKED-EYE LIMIT
    G            +
    N     0      +   BRIGHTEST STAR
    I            +   JUPITER (AT BRIGHTEST)
    T   - 5      +   VENUS (AT BRIGHTEST)
    U            +
    D   -10      +
    E            +   FULL MOON
        -15      +
                 +
        -20      +
                 +
        -25      +
                 +   SUN
```

WHY IS THE PHOTOGRAPHIC LIMIT GREATER THAN THE VISUAL LIMIT??
?DONT KNOW
PHOTOGRAPHIC PLATES CAN COLLECT LIGHT FOR MANY MINUTES OR EVEN HOURS LONGER THAN THE EYE, HENCE ONE CAN DETECT PHOTOGRAPHICALLY MUCH DIMMER STARS.

THE DISTANCE AN OBJECT IS FROM THE EARTH HAS A LARGE EFFECT ON THE APPARENT MAGNITUDE.
OK VALERIE, THEN LET US CONSIDER DISTANCES FOR A TIME.
WHAT UNIT DO YOU THINK WOULD BE USEFUL IN MEASURING AND EXPRESSING STELLAR DISTANCES ?LIGHTYEARS
THAT IS A GOOD UNIT, BUT THE ONE ASTRONOMERS USE MOST OFTEN IS THE PARSEC, WHICH IS ABOUT 3.26 LIGHT YEARS. THE TERM COMES FROM 'PARALLAX SECOND' WHICH IS USED IN DETERMINING DISTANCES TO STARS BY TRIGONOMETRIC PARALLAX. (BUT THAT IS ANOTHER STORY!)
WHAT IS THE DISTANCE (IN PARSECS) TO THE NEAREST STAR TO OUR OWN GOOD OL' SUN ?

•
•
•

Documentation Date 3/75

SCIENCE (EDUCATION) (833)

CONTRIBUTED PROGRAM **BASIC**

PHYTOP
36888-18044

TITLE: INTERACTIVE METHODS FOR SELECTED TOPICS IN PHYSICS AND MATHEMATICS

DESCRIPTION:

A series of programs developed as part of the Colorado Schools Computing Science Curriculum Development Project (supported by NSF grants GW-6517 and GW-7091). The programs are described in the text: Interactive Methods For Selected Topics in Physics and Mathematics Using Computer Programs.

The programs are:

1. PLOT1 (Named "PLOT" in Textbook)	Enables user to correlate an equation to the shape of its graph.
2. GRAPH (Named "GRAPHS" in Textbook)	Helps users determine a mathematical equation from data using experimental plotting.
3. MODEQ	Program which assists in developing the mathematical relationship of data collected in an experiment.
4. ZOOMIE	Pilot a transport machine or design and pilot such a machine.
5. BLINKY	Experiment determining straight light motion of a flying object.
6. BPLOT	Graphs BLINKY data. Accessed through BLINKY.
7. MECHAN	Six part problem involving a space ship passing by a space station. The ship ejects a capsule to dock with the station.
8. NTRCPT	Simulates investigation of a foreign body with a space probe.
9. CENTRI	Investigates the effect of applying a force on a mass.
10. FORCE	Demonstrates the effect of a uniform force on the motion of a mass.
11. FIELDS	Gives magnitude of gravitational potential energy at a point for plotting.
12. FIELDT	Generates a temperature field
13. FIELDV	Simulates an isolated gravitational field and an experiment to map it.
14. FIELDF	Practice in mapping vector fields
15. FIELDE	Simulation of an electric field and procedure for exploring it.
16. SAMPLE	Sample space experiments.

DESCRIPTION continued on following page

ACKNOWLEDGEMENTS: COLORADO SCHOOLS COMPUTING SCIENCE CURRICULUM DEVELOPMENT PROJECT

DESCRIPTION continued

17.	PRINC	More sample space experiments.
18.	PROBAB	Probability experiments.
19.	UNION	Investigate formulas useful in solving probability problems.
20.	CONDI	Explores definition for conditional probability.
21.	PASCL	Investigates PASCAL's triangles.
22.	BINOM	Investigates Binomial Experiments and binomial probability distribution.
23.	QUIZ	Post-test on the probability programs.
24.	UNIVRS	Simulates a satellite orbiting about a central mass.
25.	KEP1	Plots orbits with different eccentricities.
26.	ORBECC	Plots an orbit for which the eccentricity can be calculated.
27.	MASGO	Assists in finding a mathematical relationship between the orbital speed and the radius of a circular orbit for a unit mass orbiting a central mass.
28.	CIRMO	Investigates circular motion.
29.	BLOWS	A series of activities which demonstrate the effect of a central force acting on a mass moving with uniform speed.
30.	KEP2	Simulates a mass moving in an elliptical orbit.

INSTRUCTIONS

Order "Interactive Methods for Selected Topics in Physics and Mathematics" from:

Computer Curriculum Project
Hewlett-Packard Company
11000 Wolfe Road
Cupertino, California 95014

Price is $3.95 per copy ($3.00 per book for 10 or more).

TITER
36888-18006

TITLE: SIMULATES TITRATION OF A BASE BY AN ACID

DESCRIPTION: This program simulates a titration with clues to neutralization progress by color and hints on addition amounts. Has facility for proper end point, small overrun, and gross overshooting of endpoint.

INSTRUCTIONS: Program is interactive and self-explanatory. Just GET and RUN.

SPECIAL CONSIDERATIONS: For instructional purposes:
Suitable Courses: First year chemistry

Student background required: Algebra, Descriptive Chemistry.

ACKNOWLEDGEMENTS: Richard C. Adams
Pleasant Hill High School
Pleasant Hill, Oregon 97401

RUN

```
RUN
TITER

WELL, HERE YOU ARE - THE ONE WHO DIDN'T GET TO TITRATE
THE BASE WITH THE ACID.  YOU MISSED OUT ON ALL THE FUN
OF SPILLING ACID ON YOURSELF AND BREAKING A BURETTE ($12.95) OR
MAYBE JUST A PIPETTE ($4.95).  ANYWAY, YOU'RE GOING TO GET
TO HAVE  SOME  OF THAT FUN AND DO AN ELECTRONIC TITRATION.
YOU JUST HAVE TO TELL ME WHAT STRENGTH ACID YOU WANT TO USE
(SOMEWHERE LESS THAN 1 MOLAR IS NICE) AND HOW MUCH BASE
YOU WANT TO TITRATE.  WE'LL BE USING A RED INDICATOR,
PHENOLPHTHALEIN  (FEE NO THAY LEE IN) WHICH IS A PRETTY REDDISH-
MAGENTA COLOR IN BASE, SO YOU'LL BE SHOOTING FOR A CLEAR
SOLUTION AT NEUTRALITY.       LET'S GET STARTED!

WHAT MOLARITY OF ACID DO YOU WANT TO USE??.2
FINE.  NOW, HOW MANY MILLILITERS OF BASE DO YOU WANT TO USE?5
O.K. START OFF WITH A VOLUME OF ACID (IN MILLILITERS) AND
I'LL TRY TO TELL YOU HOW THE COLOR'S COMING ALONG.
ML?1
YOU'VE HARDLY ADDED ANY AT ALL!  DO MORE NEXT TIME.
YOU'VE ADDED 1     ML OF ACID SO FAR.
ML?3
BOY!  THAT'S NOT VERY MUCH EVEN YET.  I DIDN'T SEE ANY COLOR
CHANGE.  TRY MORE NEXT TIME.  REALLY!
YOU'VE ADDED 4     ML OF ACID SO FAR.
ML?3
THAT'S STILL NOT VERY GOOD.
YOU'VE ADDED 7     ML OF ACID SO FAR.
ML?2
HEY!  IT CLEARED IN ONE SPOT JUST FOR A SECOND!
YOU'VE ADDED 9     ML OF ACID SO FAR.
ML?3
HEY!  IT CLEARED IN ONE SPOT JUST FOR A SECOND!
YOU'VE ADDED 12    ML OF ACID SO FAR.
ML?3
THE CLEARING LASTED A LITTLE LONGER THIS TIME!
YOU'VE ADDED 15    ML OF ACID SO FAR.
ML?2
THE CLEAR AREA KEEPS GETTING A LITTLE LARGER BEFORE IT TURNS
BACK TO ALL RED.  KEEP GOING!
YOU'VE ADDED 17    ML OF ACID SO FAR.
ML?1
YOU'RE GAINING ON IT.  YOU'D BETTER NOT ADD TOO MUCH AT A
TIME - - YOU'RE 85% THERE NOW!
YOU'VE ADDED 18    ML OF ACID SO FAR.
ML?1
THE COLOR'S BEGINNING TO FADE NOW!
YOU'VE ADDED 19    ML OF ACID SO FAR.
ML?1
THAT'S IT   IT'S CLEAR!!!  CONGRATULATIONS!!!!
O.K., YOU HAD 5     ML OF BASE AND YOU ADDED 20
ML OF YOUR ACID, WHICH WAS .2          MOLAR.  SO 20
TIMES; .2          HAS TO EQUAL 5     TIMES THE BASE'S MOLARITY.
I GET .8          FOR THE MOLARITY OF THE BASE.

THANKS FOR TITRATING WITH ME   - HE SAID ACIDLY

DONE
```

TITLE: TRIVIAL OR COMMON NAME QUIZ FOR CHEMISTRY STUDENTS

DESCRIPTION:

This program is a simple drill on the common names of selected familiar substances. It includes the formula and chemical names as well. The student is asked between 10 and 16 questions at a time after which his wrong responses are reviewed and the correct answers are given. There are six types of questions that are asked - each of which is asked once in a random order for the first six questions presented. From the seventh question on they are chosen at random, being slightly biased toward the two types involving the common name and the chemical name (#2 & #5).

The six types of questions are:

1. Give common name ask formula
2. Give common name ask chemical name
3. Give formula ask common name
4. Give formula ask chemical name
5. Give chemical name ask common name
6. Give chemical name ask formula

For any questions that are answered incorrectly, the correct answers are written onto a file and read back as a study list at the end of the session. The option to go through the drill again is then offered (with different questions and order).

INSTRUCTIONS:

The program requires string inputs depending upon the question asked. If the correct answer is not known to a question the user may type A ? or simply return (or anything else for that matter), and the program will continue with the next question. All responses are timed input.

There is one file used by this program called TRIVIF. It is a one record file, or the first record of a file. The information is written onto it three strings at a time.

SPECIAL CONSIDERATIONS:

The Data Base is easily changed. Simply add Data Statements and increase the value of N accordingly (line #900). If it is desired to use two records of the file or to increase the number of questions asked before the review is given, change the test of WØ in line # 1Ø7Ø to - 1Ø7Ø if WØ>510 then 117Ø -. The program was designed to use only the first record of a semipermanent student record file as a scratch pad, and so will not overflow into the second record, thereby protecting the contents of the file beginning with record two. This feature is transparent whether used or not.

FOR INSTRUCTIONAL PURPOSES
Suitable Courses: Basic Chemistry

ACKNOWLEDGEMENTS:

John R. Wilson
Contra Costa Community College

RUN

```
OPE-TRIVIF,1
RUN
TRIVIA

INSTRUCTIONS FOR THIS TRIVIAL NAME QUIZ:
   ALWAYS PUT PARENTHESIS () AROUND THE FOLLOWING -    H20
                                                       S04
                                                        OH
THIS IS IMPORTANT IN ORDER FOR ME TO UNDERSTAND YOUR FORMULAS.


IF YOU DON'T KNOW THE ANSWER, TYPE  ?  OR PRESS  RETURN .

 1     WHAT IS THE COMMON NAME OF
                  CALCIUM CARBONATE?:CA;COT\
CALCITE
   YES! ALSO THE FORMULA IS CACO3.


 2     WHAT IS THE COMMON NAME OF
                  NA(OH)?:SALT
NO, IT IS CALLED LYE.


 3     WHAT IS THE FORMULA OF
                  MAGNESIUM SULFATE HEPTAHYDRATE?:MG(SO4)2(H2Ø)
NO, THE CORRECT ANSWER IS MG(SO4)7(H2O).


 4     WHAT IS THE CHEMICAL NAME OF
                  NAHCO3?:SODIUM BICARBONATE
   YES! ALSO THE COMMON NAME IS BAKING SODA.


 5     WHAT IS THE FORMULA OF
                  VINEGAR?:HC2H3Ø2
NO, THE CORRECT ANSWER IS HC2H3O2.


 6     WHAT IS THE CHEMICAL NAME OF
                  QUICKSILVER?:MERCURY
   YES! ALSO THE FORMULA IS HG.


 7     WHAT IS THE COMMON NAME OF
                  NA2(B407)1Ø(H2O)?:BORAX
   YES! ALSO THE CHEMICAL NAME IS SODIUM TETRABORATE DECAHYDRATE.


 8     WHAT IS THE CHEMICAL NAME OF
                  AL2O3?:ALUMINUM OXIDE
   YES! ALSO THE COMMON NAME IS ALUMINA.


 9     WHAT IS THE CHEMICAL NAME OF
                  PBO?:LEAD OXIDE
  YES, BUT IT IS MORE CORRECT TO SAY LEAD(II) OXIDE.


 1Ø    WHAT IS THE COMMON NAME OF
                  POTASSIUM HYDROGEN TARTRATE?:CREAM OF TARTAR
   YES! ALSO THE FORMULA IS KHC4H406.

=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
  YOU GOT  [illegible]    RIGHT.
           3      WRONG.
           Ø      TO SLOW.

      STUDY LIST:
LYE = NA(OH) = SODIUM HYDROXIDE
EPSOM SALTS = MG(SO4)7(H2O) = MAGNESIUM SULFATE HEPTAHYDRATE
VINEGAR = HC2H3O2 = ACETIC ACID
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
 I WILL WAIT  24    SECONDS FOR YOU TO STUDY THIS LIST...
   PRESS  CR  IF YOU DON'T WANT TO WAIT.
DO YOU WANT TO CONTINUE THIS DRILL ?NO
DONE
```

MCMAN
36888-18002

TITLE: CARDIOVASCULAR SIMULATION

DESCRIPTION:

MCMAN is an interactive model used in teaching basic principles of hemodynamics. It was developed by the Department of Clinical Epidemiology and Biostatistics of McMaster University, Hamilton, Ontario, Canada and is employed in the instruction of first year medical students in the problem-oriented curriculum of that instruction. MCMAN is a 2000 Series BASIC language version of the HP 3000 FORTRAN program MCMAN.

The model is a synthetic person who has a heart inside a chest, systemic arteries and arterioles, a capillary bed, and veins collecting blood from the capillary bed and returning it to the heart. "MCMAN" thus has a complete systemic circulatory system, and when the heart is working it will circulate blood. To speed up computation, the heart is treated as a single chamber filling the right atrium and pumping blood out into the aorta. The pulmonary circulation is regarded as simply a parallel path, and not (as in life) in series with the systemic circulation. However, this makes the model unrealistic only when one side of the heart is able to pump much less than the other (e.g. because of valve disease). "MCMAN" cannot therefore simulate the effects of valve lesions but it can simulate most types of generalized heart disease. "MCMAN" also possesses synthetic baroreceptors similar in operation to those which are normally situated at the bifurcation of the common carotid artery and at the aortic arch. These act in such a way as to stabilize blood pressure.

The performance of the heart can be described by a starling curve relating effective cardiac input pressure to cardiac output. The pressure drop between the systemic arteries and the systemic capillary bed can be derived from the cardiac output and the mean value of arterial resistance; and the right atrial pressure (input pressure of the heart) can be derived from the capillary pressure, the mean venous resistance, and the cardiac output. About 80% of the blood is contained on the low-pressure side of the circulation, and the mean systemic pressure at which it is contained depends on the blood volume and on the capacitance of the low pressure circulation.

All these relationships can be expressed as a series of simultaneous equations, and for a given blood volume, systemic arterial resistance, venous resistance, circulatory capacitance, and cardiac performance characteristics, there is only one solution which satisfies all the equations. The computer is programmed to find this solution in an interactive manner. In addition, the performance of the simulated circulation in the course of time can be determined by solving the appropriate equations at suitably short simulated time intervals, and then allowing a simulated arterial baroreceptor feed-back loop to operate with a realistic time constant, appropriately modifying the values of arterial resistance, circulatory capacitance, and cardiac function.

This mathematical model is presented in the form of a teaching display which puts out on a teleprinter or CRT a 1 min record of systemic arterial pressure and pulse rate appropriate to a normal human subject. At the end of each simulated minute of recording, the current values for cardiac output, mean capillary pressure, and right atrial pressure, together with other values, are put out in numerical form. Operation of the model then ceases and the terminal user is invited to make changes in fixed arterial and/or venous resistance (simulating the effects of vasoconstrictor or vasodilator drugs), in cardiac contractility

Continued on following page.

ACKNOWLEDGEMENTS: Dept. of Clinical Epidemiology and Biostatistics
McMaster University Medical School

DESCRIPTION continued

(simulating the effects of myocardial infarction, or digitalis), in intrathoracic pressure (which can be raised to simulate positive pressure ventilation), in pericardial restriction of cardiac filling (to simulate pericardial tamponade) or in blood volume (to simulate haemorrhage or transfusion). In addition, the setting of the arterial baroreceptor feed-back loop can be altered to that appropriate to a hypertensive subject, or the baroreceptor loop can be permanently interrupted to simulate the effect of cutting the buffer nerves. Combinations of any of these changes can be made and studied, and graphs of blood pressure and other variables made over successive simulated minutes of observation.

The example shown is of an actual run. After the preamble the program produces a vertical graph of arterial blood pressure from its simulated subject (in the span of X's the right-most X represents systolic and the left-most X represents diastolic pressure, approximately). At the end of the initialization period the operating values for BP, right atrial pressure, mean capillary pressure, cardiac output, stroke volume, heart rate, arterial resistance, venous resistance (in each case the total resistance in the systemic circuit), and finally cardiac contractility (the slope of the Starling function curve relating cardiac output to filling pressure) are given. Note, incidentally, that the vertical line of dots on the graph represent heart rate approximately, but the heart rate at the end of each run is accurately given as a number.

Then follows a list of the working values of variable which you can change: the basic mean values of arterial and venous resistance as percentages of normal (you could simulate a pressor drug by increasing 1 and a depressor drug by decreasing 1, and an arterial and venous vasodilator drug, e.g. nitroglycerine, by decreasing both 1 and 2 in proportion); the basic contractile function of the heart (remember that this is modified continuously by vagal and sympathetic influences and circulating adrealine, so even if you reduced this to, say, a nominal 30% of normal, to simulate a patient with a very bad heart, the actual contractility would be kept partly restored by sympathetic action); the intra-thoracic pressure (normally averaged at -2 mm Hg, but you can change this and see how very sensitive the circulation is to small changes, which greatly influence venous return); 'limiting cardiac input pressure' (factor 5) - this is the effective filling pressure above which no further increase in output can be obtained: pericardiacal tamponade may be simulated by making this some small figure, e.g. 3 mm Hg, indicating that up to 3 mm filling pressure the heart will pump normally, but at filling pressures above this no further output can be obtained); blood volume, in ml (you can simulate a 1000 cc haemorrhage by making this 4000). In the example shown factors 1 and 2 were greatly reduced, simulating the effects of nitroglycerine.

INSTRUCTIONS

To interact with the model respond to the questions as follows:

1. When asked "DO YOU WANT TO CONTINUE?",

   type YES if you wish to proceed with another 60 second simulation period; type NO if you wish to discontinue using the model.

2. When asked "DO YOU WANT A PLOT?",

   type YES if you want a graphical display of heart rate and blood pressure over time; type NO if you wish to proceed directly to the calculated values of the simulation.

3. When asked "DO YOU WANT TO CHANGE ANY OF THESE FACTORS?"

   you are being given the opportunity to alter the 6 basic variables of the systemic circulation; type YES if you wish to make changes before the next simulation, type NO if you wish to proceed "as is" or move to altering the blood pressure stabilizing system.

4. When asked "CHANGE FACTOR (1...6 OR Ø TO QUIT)?"

   type any of the digits Ø, 1, 2, 3, 4, 5, 6, where Ø means "no more changes to be made", 2 means "a change is to be made to venous resistance", etc.

5. When asked: "FACTOR X?", (where X is any digit 1 thru 6),

   type the new value for that factor. For example, each of factors 1, 2, and 3 are stated in percent of normal; your response of 50 would mean "reset the variable to behave at 50% of normal functioning." Likewise, 150 would mean "50% greater than normal", etc. Do not use the character "%"; do not enter whole percents as decimal fractions. Factors 4 and 5 are pressures stated in millimeters of mercury (HG), Factor 6 is volume stated in milliliters. Do not input these units of measure (i.e. the letters "MM HG" or "ML") as part of the numeric value you give the model.

Continued on following page.

INSTRUCTIONS continued

6. When asked "DO YOU WANT TO CUT THE BUFFER NERVES........?"

   type YES if you wish to put the blood pressure stabilizing system completely out of action; type NO if you wish to reset baraceptor sensitivity.

7. When asked "DO YOU WANT TO RESET BAROCEPTER SENSITIVITY.....?",

   type YES if you wish to simulate the barocepter sensitivity of a hypertensive patient, type NO if you do not.

RUN

```
RUN
MCMAN

MCMAN WAS DEVELOPED AND IS CURRENTLY USED IN THE MEDICAL
CURRICULUM AT MCMASTER UNIVERSITY, HEALTH SCIENCES CENTRE,
HAMILTON, ONTARIO, CANADA.

NEITHER HEWLETT-PACKARD NOR MCMASTER UNIVERSITY ASSUME
ANY RESPONSIBILITY FOR THE OPERATION AND CORRECTNESS OF THIS MODEL

DO YOU WANT TO CONTINUE?YES

DO YOU WANT A PLOT?YES

(XXX) SYSTOLIC/DIASTOLIC PRESSURES - MM HG   (.) HEART RATE - BEATS/MIN
TIME 0    20   40   60   80   100  120  140  160  180  200  220  240  260
SECS .    .    .    .    .    .    .    .    .    .    .    .    .    .
-28                          .XXXXXXXXXXXXXX
-26                          .XXXXXXXXXXX
-24                           .XXXXXXXXXXX
-22                          .XXXXXXXXXXX
-20                          X.XXXXXXXXXXX
-18                         .  XXXXXXXXXXX
-16                          .XXXXXXXXXXXX
-14                          .XXXXXXXXXXX
-12                          X.XXXXXXXXXXX
-10                          .XXXXXXXXXXXX
-8                            .XXXXXXXXXXX
-6                           .XXXXXXXXXXX
-4                           .XXXXXXXXXXXX
-2                           .XXXXXXXXXXXX
 0                           .XXXXXXXXXXXX

ARTERIAL (AORTIC) PRESSURE
SYSTOLIC=129.5   DIASTOLIC= 82.9   MEAN=    98.5 MM HG
CENTRAL VENOUS (RIGHT ATRIAL) PRESSURE=     1.6 MM HG
MEAN CAPILLARY PRESSURE=    13.1 MM HG
CARDIAC OUTPUT=    5.2 L/MIN
STROKE VOLUME=    64.2 ML/BEAT     HEART RATE= 80.9 BEATS/MIN
ARTERIAL RESISTANCE=    16.5       VENOUS RESISTANCE=    2.2 MM HG/L/MIN
CARDIAC CONTRACTILITY=    1.4 L/MIN/MM HG

FINAL VALUES FOR THIS RUN WERE:
1. ARTERIAL RESISTANCE= 100 % OF NORMAL
2. VENOUS RESISTANCE= 100 % OF NORMAL
3. CARDIAC CONTRACTILITY= 100 % OF NORMAL
4. MEAN INTRATHORACIC PRESSURE=  -2.0 MM HG
5. LIMITING CARDIAC INPUT PRESSURE=     8.0 MM HG
6. BLOOD VOLUME= 5000 ML

DO YOU WANT TO CHANGE ANY OF THESE FACTORS?YES
CHANGE FACTOR (1..6 OR 0 TO QUIT)?1
FACTOR 1?40
CHANGE FACTOR (1..6 OR 0 TO QUIT)?2
FACTOR 2?40
CHANGE FACTOR (1..6 OR 0 TO QUIT)?0


1=  40, 2=  40, 3= 100, 4=  -2, 5=    8, 6=5000
DO YOU WANT A PLOT?YES
```

```
(XXX) SYSTOLIC/DIASTOLIC PRESSURES - MM HG   (.) HEART RATE - BEATS/MIN
TIME 0    20   40   60   80   100  120  140  160  180  200  220  240  250
SECS .    .    .    .    .    .    .    .    .    .    .    .    .    .
 2                        .         XXXXXXXXXXXXXXXXXX
 4               XXXXXXXXX.XXX
 6               XXXXXXXX.XXX
 8              XXXXXXXX.XXXX
 10             XXXXXXXXXX.XX
 12              XXXXXXXXXX.XXX
 14               XXXXXXXXXX.XX
 16                XXXXXXXXXXX.X
 18                XXXXXXXXXXXX.
 20                 XXXXXXXXXXX.
 22                 XXXXXXXXXXXXX .
 24                  XXXXXXXXXXXXX.
 26                  XXXXXXXXXXXXX .
 28                   XXXXXXXXXXXX  .
 30                   XXXXXXXXXXXXX  .
 32                   XXXXXXXXXXXXX   .
 34                    XXXXXXXXXXXX   .
 36                    XXXXXXXXXXXXX  .
 38                    XXXXXXXXXXXXX   .
 40                    XXXXXXXXXXXXX  .
 42                    XXXXXXXXXXXXX .
 44                    XXXXXXXXXXXXX  .
 46                    XXXXXXXXXXXXXX .
 48                    XXXXXXXXXXXXX .
 50                    XXXXXXXXXXXXX .
 52                    XXXXXXXXXXXXX.
 54                   XXXXXXXXXXXXX .
 56                   XXXXXXXXXXXXX .
 58                   XXXXXXXXXXXXX .
 60                   XXXXXXXXXXXXX .

>>> MY HEART IS POUNDING
ARTERIAL (AORTIC) PRESSURE
SYSTOLIC=115.6   DIASTOLIC= 68.9   MEAN=    84.5 MM HG
CENTRAL VENOUS (RIGHT ATRIAL) PRESSURE=     3.2 MM HG
MEAN CAPILLARY PRESSURE=    12.2 MM HG
CARDIAC OUTPUT=   10.2 L/MIN
STROKE VOLUME=    81.9 ML/BEAT     HEART RATE=124.4 BEATS/MIN
ARTERIAL RESISTANCE=      7.4      VENOUS RESISTANCE=    0.9 MM HG/L/MIN
CARDIAC CONTRACTILITY=    2.0 L/MIN/MM HG

FINAL VALUES FOR THIS RUN WERE:
1. ARTERIAL RESISTANCE=  40 % OF NORMAL
2. VENOUS RESISTANCE=  40 % OF NORMAL
3. CARDIAC CONTRACTILITY= 100 % OF NORMAL
4. MEAN INTRATHORACIC PRESSURE=  -2.0 MM HG
5. LIMITING CARDIAC INPUT PRESSURE=     8.0 MM HG
6. BLOOD VOLUME= 5000 ML

DO YOU WANT TO CHANGE ANY OF THESE FACTORS?NO
DO YOU WANT TO CHANGE THE OPERATION OF THE SYSTEMIC
ARTERIAL BAROCEPTORS?YES
DO YOU WANT TO CUT THE BUFFER NERVES AND PUT THE B.P.
STABILIZING SYSTEM PERMANENTLY OUT OF ACTION?YES


YOU HAVE CUT THE BUFFER NERVES
DO YOU WANT A PLOT?YES
```

```
(XXX) SYSTOLIC/DIASTOLIC PRESSURES - MM HG   (.) HEART RATE - BEATS/MIN
TIME 0    20   40   60   80   100  120  140  160  180  200  220  240  260
SECS .    .    .    .    .    .    .    .    .    .    .    .    .    .
 62                   XXXXXXXXXXXXX .
 64                    XXXXXXXXXXXXX  .
 66                    XXXXXXXXXXXXX .
 68                    XXXXXXXXXXXXX .
 70                    XXXXXXXXXXXXX  .
 72                     XXXXXXXXXXXXX   .
 74                     XXXXXXXXXXXXX   .
 76                     XXXXXXXXXXXXX    .
 78                      XXXXXXXXXXXXX   .
 80                      XXXXXXXXXXXXX     .
 82                      XXXXXXXXXXXXX    .
 84                      XXXXXXXXXXXXX      .
 86                      XXXXXXXXXXXXX    .
 88                       XXXXXXXXXXXXX     .
 90                       XXXXXXXXXXXXXX      .
 92                      XXXXXXXXXXXXX      .
 94                       XXXXXXXXXXXXX      .
 96                       XXXXXXXXXXXXX      .
 98                       XXXXXXXXXXXXXX     .
 100                      XXXXXXXXXXXXXX      .
 102                       XXXXXXXXXXXXX      .
 104                       XXXXXXXXXXXXX     .
 106                      XXXXXXXXXXXXX       .
 108                       XXXXXXXXXXXXX     .
 110                       XXXXXXXXXXXX         .
 112                       XXXXXXXXXXXXXX     .
 114                       XXXXXXXXXXXXXX      .
 116                       XXXXXXXXXXXXX      .
 118                      XXXXXXXXXXXXX        .
 120                       XXXXXXXXXXXXX       .

>>> MY HEART IS POUNDING
ARTERIAL (AORTIC) PRESSURE
SYSTOLIC=135.1   DIASTOLIC= 86.9   MEAN=   103.0 MM HG
CENTRAL VENOUS (RIGHT ATRIAL) PRESSURE=     2.8 MM HG
MEAN CAPILLARY PRESSURE=    13.0 MM HG
CARDIAC OUTPUT=    11.6 L/MIN
STROKE VOLUME=    68.7 ML/BEAT     HEART RATE=168.8 BEATS/MIN
ARTERIAL RESISTANCE=     8.0      VENOUS RESISTANCE=    0.9 MM HG/L/MIN
CARDIAC CONTRACTILITY=    2.4 L/MIN/MM HG

FINAL VALUES FOR THIS RUN WERE:
1. ARTERIAL RESISTANCE=  40 % OF NORMAL
2. VENOUS RESISTANCE=  40 % OF NORMAL
3. CARDIAC CONTRACTILITY= 100 % OF NORMAL
4. MEAN INTRATHORACIC PRESSURE=  -2.0 MM HG
5. LIMITING CARDIAC INPUT PRESSURE=     8.0 MM HG
6. BLOOD VOLUME= 5000 ML

DO YOU WANT TO CHANGE ANY OF THESE FACTORS?NO
DO YOU WANT TO CONTINUE?NO

DONE
```

MUSIC
36888-18028

TITLE: TWELVE TONE COMPOSITION PROGRAM

DESCRIPTION:

The program produces simple 12 tone compositions. Values given to various elements of the composition (pitch, time value, etc.) are determined by random processes. The user interacts with the program by assigning probabilities to the random elements. MUSIC is designed primarily to illustrate how random elements and probabilities may be used in composition and how a computer may be used to aid in the selection of random elements.

INSTRUCTIONS:

Order 36888-90028, $5.00 for complete documentation.

SPECIAL CONSIDERATIONS:

The program uses four data files, each 4 records long, to store intermediate results. These files are named V1, V2, V3, V4 in statement 320 of the program. This statement may have to be changed under certain circumstances. See section "Data Files" p. 15 of the manual for more details. In any case the four files must exit on a read-write basis in order for the program to run.

```
CRE-V1,4
CRE-V2,4
CRE-V3,4
CRE-V4,4
```

The program is used as part of a third year course in electronic music for composition majors. It presupposes some knowledge of twelve tone composition technique.

ACKNOWLEDGEMENTS:

Bill Jarosz and Joann Preston
De Paul University

RUN

```
RUN
MUSIC

TWELVE - TONE COMPOSITION PROGRAM  11/28/72 VERSION


NO. OF MEASURES (16 MAX.)?6
TIME SIGNATURE=X/4, (X=9 MAX.)?3
NO. OF VOICES (4 MAX.)?2
CHOOSE PROBABILITIES FOR OCTAVES (1=YES,0=NO)?0
CHOOSE PROBABILITIES FOR TIME VALUES (1=YES,0=NO)?0
CHOOSE PROBABILITIES FROM 0 TO 1 FOR REST IN VOICE(S)
VOICE 1    ?0
VOICE 2    ?0
INPUT YOUR OWN 12-TONE ROW (1=YES,0=NO)?0


12 TONE ROW
C              C#             B              D
F              G              A              D#
A#             G#             E              F#

 EACH NOTE PRINTOUT SHOWS NOTES, OCTAVES, NO. OF BEATS


VOICE 1
C 1 .5         B 3 4          A 1 3.5        G#3 3
E 2 2          C 2 3          B 1 2


VOICE 2
C#2 .5         D 3 1          F 3 2          G 2 2.5
D#1 1.5        A#2 3.5        F#1 2.5        C#3 3
D 1 .5         F 1 1

DONE


320 FILES HIGH,MID10,MID12,LOW
OPE-HIGH,2
OPE-MID10,2
OPE-MID12,2
OPE-LOW,2
5000 DATA 0,0,1,0,0,1,0,0,1
5010 DATA 0,1,0,0,.9,.1,0,.9,.1
5020 DATA .1,.9,0,.1,.9,0,0,1,0
5030 DATA 1,0,0,1,0,0,1,0,0
RUN
MUSIC

TWELVE - TONE COMPOSITION PROGRAM  11/28/72 VERSION


NO. OF MEASURES (16 MAX.)?8
TIME SIGNATURE=X/4, (X=9 MAX.)?4
NO. OF VOICES (4 MAX.)?4
CHOOSE PROBABILITIES FOR OCTAVES (1=YES,0=NO)?1
CHOOSE PROBABILITIES FOR TIME VALUES (1=YES,0=NO)?0
CHOOSE PROBABILITIES FROM 0 TO 1 FOR REST IN VOICE(S)
VOICE 1    ?.1
VOICE 2    ?.1
VOICE 3    ?.1
VOICE 4    ?.1
INPUT YOUR OWN 12-TONE ROW (1=YES,0=NO)?0


12 TONE ROW
D#             A              A#             E
D              B              F#             G
F              C#             C              G#

 EACH NOTE PRINTOUT SHOWS NOTES, OCTAVES, NO. OF BEATS
```

VOICE 1

D#3 1.5	D 3 1	F#3 2.5	C#3 .5
C 3 .5	D#3 2.5	B 3 1.5	F 3 .5
C#3 1.5	G#3 1	A#3 1	D 3 1
B 3 2	F 3 .5	C 3 2.5	A#3 3.5
F 3 4	R 2	B 3 2	F 3 .5

VOICE 2

A 2 3.5	G 3 2.5	A 2 1	A#2 1
R 1.5	G 2 2.5	D#2 4	G 2 2.5
D#2 1.5	R 1.5	D 2 2.5	C 2 2
D#2 2.5	D 2 3.5		

VOICE 3

A#2 3.5	F 2 2	R 1.5	E 2 4
C 2 4	F#2 4	A 2 2	E 2 1.5
G 2 1	C#2 1.5	G#1 1.5	A#2 2.5
R 2	G 2 1		

VOICE 4

E 1 1.5	B 1 4	G#1 1.5	D 1 1.5
F#1 3	R 1	A 1 .5	E 1 4
C#1 1	G#1 3.5	B 1 .5	F#1 2.5
R 1.5	A 1 1.5	E 1 2	F#1 2.5

DONE

TITLE: SOCIAL SCIENCE INQUIRY PACKAGE

INQUIR
36888-18001

DESCRIPTION:

There are 2 programs in this package: INQUIR and INQUIRH. INQUIR is a social science data analysis package which allows the user to create and modify data files and perform a number of statistics on that data including frequencies, both means and standard deviations, crosstabs with chi square, degrees of freedom and gamma. Data can be recoded and statistics done on subpopulations as well.

INSTRUCTIONS:

Order HP 5951-7389 INQUIR Reference Manual.

This package is the software package for the following curriculum packages to be published by Hewlett-Packard's Computer Curriculum Project.

Title

Political Awareness - HP 5951-7382 by Jim Hessler

Analyzing Crime - HP 5951-7380 by Justin Green

INQUIR Reference Manual - HP 5951-7389, by Don Holznagel

For further information contact:

Hewlett-Packard Computer-Based
Educational Materials
Scientific Press
1629 Channing Avenue
Palo Alto, Calif. 94303

ACKNOWLEDGEMENTS:

Don Holznagel
Dan Klassen

PYRMID
36888-18013

TITLE: BUILD A PYRAMID

DESCRIPTION: The purpose of this game is to build as high a pyramid as possible with the available amount of money. The pyramid should be completed in 8 years.

INSTRUCTIONS: The following conventions should be observed:

- The height of a pyramid should neither be smaller than 15 metres nor exceed 200 metres
- The number of slaves should not be smaller than 1000
- The work time of slaves shouldn't exceed 20 hours/day
- The maximum of 9 pyramids can be built within one run

ACKNOWLEDGEMENTS: Hannu Kurki-Suonio
Helsinki, Finland

RUN

```
RUN
PYRMID

PYRAMID-- HANNU KURKI-SUONIO, HELSINKI, FINLAND

  IT IS YEAR 2710 B.C. YOU ARE RULING THE ANCIENT EGYPT AFTER
THE DEATH OF YOUR PREDECESSOR THE LATE PHARAOH KHAFRA. AN OR-
ACLE HAS PREDICTED THAT YOU WILL DEPART THIS LIFE IN EIGHT
YEARS AND THEREFORE YOU ARE BOUND TO START BUILDING YOUR PYR-
AMID DIRECTLY.

YOU'VE GOT 40000.      GOLD RINGS
HOW MANY METRES HIGH DO YOU WISH TO BUILD YOUR PYRAMID ?40
HOW MANY SLAVES DO YOU ACQUIRE ?20000
HOW MANY OF THEM DO YOU APPOINT TO FOREMEN ?700
HOW MANY SLAVES TO THE QUARRY                ?3500
HOW MANY TO CARRY STONES                     ?4000
THERE ARE 11800   SLAVES LEFT AS BUILDING LABOUR
YOU'VE STILL GOT 15977   GOLD RINGS
HOW MUCH FOR TOOLS ETC.?5000
HOW MANY HOURS MUST THE SLAVES WORK DAILY ?12
HOW MUCH WILL YOU SPEND ON FOOD NEXT YEAR ?10000


YOUR PYRAMID BECAME COMPLETED IN 1     YEARS
YOUR PYRAMID HAS NOW 40516.      STONES AND 40   METRES OF HEIGHT
YOUR SUCCESSOR WILL INHERIT THE REMAINING 977  GOLD RINGS

PYRAMID OF PHARAOH MENKAURA 1       .          40   METRES
                                   *I*
                                  * I *
                                 *  I  *
                                *   I   *
                               *    I    *
                              *     I     *
                              ..    I     ..
                                 .. I ..
                                   ..I ..

DOES YOUR SUCCESSOR WANT TO BUILD A PYRAMID (1 OR 0) ?1

########


YOU'VE GOT 40977.      GOLD RINGS
HOW MANY METRES HIGH DO YOU WISH TO BUILD YOUR PYRAMID ?89
HOW MANY SLAVES DO YOU ACQUIRE ?19000
HOW MANY OF THEM DO YOU APPOINT TO FOREMEN ?1000
HOW MANY SLAVES TO THE QUARRY                ?5200
HOW MANY TO CARRY STONES                     ?5400
THERE ARE 7400    SLAVES LEFT AS BUILDING LABOUR
YOU'VE STILL GOT 17682   GOLD RINGS
HOW MUCH FOR TOOLS ETC.?4000
HOW MANY HOURS MUST THE SLAVES WORK DAILY ?12
HOW MUCH WILL YOU SPEND ON FOOD NEXT YEAR ?5000


 1    YEAR

 102261.     STONES             9    METRES

SLAVES STARVED          3210
SLAVES DIED OF OVERWORK 585
SLAVES ESCAPED          0

YOU HAVE NOW 14205   WORKERS AND 8682    GOLD RINGS

ANY CHANGES TO WORKING CONDITIONS (1 OR 0) ?0


 2    YEAR

 164741.     STONES             17   METRES

SLAVES STARVED          205
SLAVES DIED OF OVERWORK 554
SLAVES ESCAPED          0
```

```
YOU HAVE NOW 13446   WORKERS AND 3682    GOLD RINGS

ANY CHANGES TO WORKING CONDITIONS (1 OR Ø) ?1
HOW MUCH DO YOU WANT TO INCREASE THE NUMBER OF FOREMEN ?Ø
HOW MANY HOURS MUST THE SLAVES WORK DAILY ?8
HOW MUCH WILL YOU SPEND ON FOOD NEXT YEAR ?28ØØ


 3    YEAR

 19Ø875.    STONES             2Ø   METRES

SLAVES STARVED          3246
SLAVES DIED OF OVERWORK Ø
SLAVES ESCAPED          Ø

YOU HAVE NOW 1Ø2ØØ   WORKERS AND 882  GOLD RINGS

ANY CHANGES TO WORKING CONDITIONS (1 OR Ø) ?1
HOW MUCH DO YOU WANT TO INCREASE THE NUMBER OF FOREMEN ?Ø
HOW MANY HOURS MUST THE SLAVES WORK DAILY ?9
HOW MUCH WILL YOU SPEND ON FOOD NEXT YEAR ?8ØØ


 4    YEAR

 2Ø1148.    STONES             22   METRES

SLAVES STARVED          8457
SLAVES DIED OF OVERWORK Ø
SLAVES ESCAPED          Ø

YOU HAVE NOW 1743    WORKERS AND 82   GOLD RINGS

ANY CHANGES TO WORKING CONDITIONS (1 OR Ø) ?1
HOW MUCH DO YOU WANT TO INCREASE THE NUMBER OF FOREMEN ?Ø
HOW MANY HOURS MUST THE SLAVES WORK DAILY ?1Ø
HOW MUCH WILL YOU SPEND ON FOOD NEXT YEAR ?8Ø

ALL WORKERS DIED OF STARVATION
YOUR PYRAMID HAS NOW 2Ø1148.    STONES AND 22   METRES OF HEIGHT
YOUR SUCCESSOR WILL INHERIT THE REMAINING 2    GOLD RINGS


DOES YOUR SUCCESSOR WANT TO BUILD A PYRAMID (1 OR Ø) ?Ø

########

THERE ARE 2    PYRAMIDS ON THE SEPULCHER AREA

PYRAMID OF MENKAURA 1    , 4Ø   M
PYRAMID OF MENKAURA 2    , 22   M

DONE
```

TITLE: TIME SERIES ANALYSIS GROWTH & DEVELOPMENT OF U.S. 1790-1860 36888-18033 TSAP

DESCRIPTION:

There are 4 programs and 1 file in this package: TSAP, TSAP1, TSAP2, TSAP3 and TSAPF.

These programs, all accessed through TSAP, allow students to do time series analysis of a data base (TSAPF) which contains information on all the states of the U.S.

INSTRUCTIONS:

The program was written to support the publication, The Growth and Development of the United States, 1790-1860 by John G. Kolp published by the Hewlett-Packard Computer Curriculum Project, (HP 5951-7381)

For further information contact:

Hewlett-Packard Computer-Based Educational Materials
Scientific Press
1629 Channing Avenue
Palo Alto, Calif. 94303

The publication is needed for complete user instruction.

SPECIAL CONSIDERATIONS:

FOR INSTRUCTIONAL PURPOSES
Suitable Courses: High School and College American History Courses.

ACKNOWLEDGEMENTS: John G. Kolp
University of Iowa

RUN

```
RUN
TSAP

TIME SERIES ANALYSIS PACKAGE

DO YOU NEED INSTRUCTIONS?YES
THIS PROGRAM ALLOWS YOU TO RETRIEVE AND/OR
ANALIZE DATA CONCERNING THE U.S. FOR THE PERIOD
1790 THRU 1860.  THE DATA IS STORED INTERNALLY IN THE
COMPUTER.
LET'S IMAGINE THE FILE LOOKS LIKE THE FOLLOWING OUTLINE:

I. US DATA
   A. STATE (STATE)
     1. GEORGIA
   B. YEARS (YEARS)
     1. 1790
   C. TOTAL POPULATION (TOTPOP)
     1. 55 (IN 100 THOUSANDS)
   D. NUMBER OF SLAVES (SLAVES)
     1. 550 (IN 10 THOUSANDS)
   E. NUMBER OF FREE BLACKS (BLACKS)
     1. 66 (IN THOUSANDS)
   F. PERCENTAGE OF POPULATION FOREIGN BORN (FORBOR)
     1. 0 (1850-1860 ONLY)
   G. PERCENTAGE POPULATION LIVING IN URBAN AREAS (URBAN)
     1. 10
   H. AVERAGE ACRES PER FARM (FARMS)
     1. 0 (1850-1860 ONLY)
   I. LAND SALES-PREVIOUS DECADE (LAND)
     1. 0 (1820-1860 ONLY)
   J. MILES OF RAILROAD (RAILS)
     1. 0 (1850-1860 ONLY)
   K. VALUE OF MANUFACTURING (MANFT)
     1. 0 (1810,1840,1850,1860)
   L. VALUE OF IMPORTS (IMPORT)
   M. VALUE OF EXPORTS (EXPORT)
   N. PERCENT OF US TOBACCO PRODUCTION (TOBAC)
   O. PERCENT OF US COTTON PRODUCTION (COTTON)
   P. PERCENT OF US CORN PRODUCTION (CORN)
   Q. SECTION OF COUNTRY (SECTN)
      1. NORTH
      2. SOUTH

THIS PARTICULAR OUTLINE REPRESENTS ONE RECORD - GEORGIA
IN THE YEAR 1790.  THERE ARE RECORDS FOR GEORGIA FOR THE
YEARS 1790 THRU 1860, AS WELL AS RECORDS ON OTHER STATES
FOR THE SAME YEARS.  THIS COLLECTION OF RECORDS MAKES UP
A FILE.
STATE, YEAR, SLAVES ARE WHAT IS KNOWN AS VARIABLES.
INSTEAD OF ALWAYS HAVING TO SPELL THEM OUT SUCH AS
'TOTAL POPULATION' , I'LL MAKE IT EASY FOR YOU. YOU
CAN REFERENCE A VARIABLE BY USING 'TOTPOP' OR 'MANFT'.
THESE VARIABLE NAMES ARE INCLUDED IN THE PARENTHSES IN THE
ABOVE LIST.  SO, WHENEVER YOU ARE ASKED FOR THE NAME
OF A VARIABLE, BE SURE TO RESPOND WITH ONE OF THOSE
LISTED ABOVE.


'DATA SELECTION CRITERION' ALLOWS YOU TO 'PINPOINT' THE
DATA YOU WISH TO ANALYZE.  YOU HAVE THE CHOICE OF
SELECTING A PARTICULAR STATE, YEAR, OR SECTION OF THE
COUNTRY.
STATE=NEWJERSEY, YEARS=1820, SECTN=1ARE EXAMPLES OF HOW TO
ENTER YOUR DATA SELECTIONS.  IF YOU WISH TO ANALYZE THE
ENTIRE FILE, SIMPLY TYPE IN NONE AS THE DATA SELECTION
CRITERION.

WE CAN ALSO PERFORM ONE 'MULTIPLE CRITERION' SELECTION.
YOU CAN SELECT ON SECTN AND YEARS IN ONE STATEMENT.
BUT IT MUST READ SECTN=? AND YEARS=???? IN PRECISELY
THAT ORDER AND FORMAT.
NOW I'LL LIST THE AVAILABLE COMMANDS.
THEY ARE:

          RETRIEVE
          PLOT
          CORRELATE
          MEAN
      AND QUIT TO STOP.

COMMAND? QUIT

DONE
```

TITLE: COMPUTER ORIENTED ACCOUNTING - INTERACTIVE VERSION

ATG
36888-18032

DESCRIPTION:

To be used in high school accounting I or bookkeeping courses.

Topics: Complete Accounting Cycle
Payroll
Bank Reconcilliation
Declining Balance

There are 6 programs in this package: ATG, ATGE, ATGIB, ATGM, ATGM1, ATGW.

INSTRUCTIONS:

Programs are self prompting so that the user can be helped by answering the questions presented to him.

To be used in conjunction with COMPUTER ORIENTED ACCOUNTING by Wilbur Pillsbury, a workbook published by South-Western Publishing Co. Available from South-Western Publishing Co.
5101 Madison Road
Cincinnati, Ohio 45227

Student Book $5
Teacher's Book $3.75 for 5 or more.

Teacher's books are free to instructors who adopt the text for use in their classes.

Originally, the programs were written for a FORTRAN IV speaking computer which uses punched cards as the input medium. To bring the benefits of Computer Oriented Acounting to as many people as possible on the secondary level, the BASIC language, interactive version was written. It is identical to the original in its purpose, problems and the book used. The student would use the workbook by Dr. Pillsbury just as before, but all transactions are entered via the terminal keyboard directly into the computer's memory, instead of keypunching cards.

To be truly interactive, the transaction data must be checked for errors as it is entered. Totals must be compared for errors in balancing as they are created, and appropriate correction capability available if and when they are needed. Moreover, should the student desire to use the same data in more than one exercise (as is the case in Chapters 7, 9 and 13), it should be available to him without his having to re-enter it. Each of these capabilities has been made readily and easily available to the student in this interactive version.

In a typical program, the student responds to questions which the computer asks him, giving his name, the chapter number, problem number and date. Transaction entries are made by giving an account number (odd-debit, even-credit) and the amount of the transaction. The student is asked to check the data after each transaction. If he sees a mistake he can correct it immediately. Once the data entry is complete, the computer relists it

Description continued on following page.

ACKNOWLEDGEMENTS: Lawrence G. Page
Central Technical High School

INSTRUCTIONS continued

in journal form. If there are adjusting or closing entries, the student is asked to enter them and they, in turn, are checked and listed. Totals of the debit and credit sides of the journal are checked for balance, and in problems 1-3 of Chapters 4-9 and all problems in Chapter 13, the exact values required are checked. Since problem 4 in each of Chapters 4-9 uses student designed data, a check is only made to see if the totals balance.

Usually the computer then asks if the student is ready for the T accounts to be listed. The student types "GO". Finally, the same "GO" response is requested for either a worksheet, balance sheet, trial balance, income statement, or post closing trial balance, depending upon the problem and chapter. As before, the totals are checked for balance and exactness. Should they not balance, or the totals not agree with those listed in the computer's memory for that problem, the computer asks the student if he would like to correct them. This is a painless procedure because the computer simply relists the original transaction data, one entry at a time, and asks if it is correct. When all necessary corrections have been made, the computer re-runs the data.

When the problem is complete, the student can either stop, reuse his data in another problem, or do a completely new problem.

Problems in Chapter 3 are checked for balance only. Chapters 10-12 have extensive error checking for problems 1 and 2 and only limited checking for problem 3.

RUN

```
RUN
ATG

ATG ? SELECTION PROGRAM
PLEASE TELL ME WHICH PROGRAM NUMBER (3-13) YOU WISH TO USE ?3
THANKYOU, YOU WILL BE USING PROGRAM ATG 3

IS THIS A SECOND RUN FOR YOUR TRANSACTION DATA (Y/N) ?N


COMPUTER ORIENTED ACCOUNTING

STUDENT NAME PLEASE ?LARRY PAGE
DATE PLEASE ?OCTOBER 12,1974
EXERCISE X-X PLEASE ?3-1

TRANSACTION DATA

COMPANY NAME ?HARRISON RADIO AND TV SALES
DATE OF ACCOUNTING PERIOD ?AUGUST 31, 1974

CASH ?4050
SUPPLIES ?810
TELEVISIONS ?4400
RADIOS ?2740
BOWMAN RADIO CO. ?3560
CAPITAL ?8440


STUDENT NAME - LARRY PAGE
EXERCISE - 3-1
DATE - OCTOBER 12,1974

                         HARRISON RADIO AND TV SALES

                               BALANCE SHEET

                              AUGUST 31, 1974


         ASSETS                                   LIABILITIES

CASH                 4050              BOWMAN RADIO CO.     3560

SUPPLIES             810

TELEVISION           4400                      CAPITAL

RADIOS               2740              CAPITAL              8440
                     -------                                -------
TOTAL ASSETS         12000             TOTAL LIAB. & CAP.   12000
                     -------                                -------
                     -------                                -------
```

```
ATG ? SELECTION PROGRAM
PLEASE TELL ME WHICH PROGRAM NUMBER (3-13) YOU WISH TO USE ?4
THANKYOU, YOU WILL BE USING PROGRAM ATG 4

IS THIS A SECOND RUN FOR YOUR TRANSACTION DATA (Y/N) ?N


COMPUTER ORIENTED ACCOUNTING

STUDENT NAME PLEASE ?LARRY PAGE
DATE PLEASE ?OCTOBER 12, 1974
EXERCISE X-X PLEASE ?4-1

TRANSACTION DATA

COMPANY NAME ?ALLISON'S BICYCLE REPAIR
DATE OF ACCOUNTING PERIOD ?SEPTEMBER 30, 1974
WHICH PROBLEM IS THIS ?1
HOW MANY TRANSACTION ENTRIES DO YOU HAVE ?10

DEBIT ENTRY #  1
ACCOUNT NUMBER ?1
AMOUNT ?5600

ACCOUNT #  1    NAME - CASH    AMOUNT =  5600
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  1
ACCOUNT NUMBER ?34
AMOUNT ?5600

ACCOUNT #  34   NAME - CAPITAL    AMOUNT =  5600
ARE THEY CORRECT (Y/N) ?Y

DEBIT ENTRY #  2
ACCOUNT NUMBER ?325
***** ERROR - ACCOUNT NUMBER IS GREATER THAN 80 *****
ACCOUNT NUMBER ?9
AMOUNT ?325

ACCOUNT #  9    NAME - SUPPLIES    AMOUNT =  325
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  2
ACCOUNT NUMBER ?2
AMOUNT ?326

ACCOUNT #  2    NAME - CASH    AMOUNT =  326
ARE THEY CORRECT (Y/N) ?Y
***** ERROR - DEBIT AND CREDIT AMOUNTS ARE NOT EQUAL TO EACH OTHER *****

DEBIT ENTRY #  2
ACCOUNT NUMBER ?9
AMOUNT ?325

ACCOUNT #  9    NAME - SUPPLIES    AMOUNT =  325
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  2
ACCOUNT NUMBER ?2
AMOUNT ?325

ACCOUNT #  2    NAME - CASH    AMOUNT =  325
ARE THEY CORRECT (Y/N) ?Y

DEBIT ENTRY #  3
ACCOUNT NUMBER ?15
AMOUNT ?100

ACCOUNT #  15   NAME - EQUIPMENT    AMOUNT =  100
ARE THEY CORRECT (Y/N) ?N
LET'S TRY AGAIN !
ACCOUNT NUMBER ?15
AMOUNT ?1000
```

```
ACCOUNT # 15    NAME - EQUIPMENT    AMOUNT =  1000
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  3
ACCOUNT NUMBER ?28
AMOUNT ?1000

ACCOUNT # 28    NAME - ACCOUNTS PAYABLE    AMOUNT =  1000
ARE THEY CORRECT (Y/N) ?Y

DEBIT ENTRY #  4
ACCOUNT NUMBER ?1
AMOUNT ?100

ACCOUNT # 1    NAME - CASH    AMOUNT =  100
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  4
ACCOUNT NUMBER ?16
AMOUNT ?100

ACCOUNT # 16    NAME - EQUIPMENT    AMOUNT =  100
ARE THEY CORRECT (Y/N) ?Y

DEBIT ENTRY #  5
ACCOUNT NUMBER ?15
AMOUNT ?1200

ACCOUNT # 15    NAME - EQUIPMENT    AMOUNT =  1200
ARE THEY CORRECT (Y/N) ?Y

CREDIT ENTRY #  5
ACCOUNT NUMBER ?2
AMOUNT ?1200

ACCOUNT # 2    NAME - CASH    AMOUNT =  1200
ARE THEY CORRECT (Y/N) ?Y


STUDENT NAME - LARRY PAGE
EXERCISE - 4-1
DATE - OCTOBER 12, 1974

                         ALLISON'S BICYCLE REPAIR

                    JOURNAL OF FINANCIAL TRANSACTIONS

                           SEPTEMBER 30, 1974


          NAME OF ACCOUNT                    DEBIT      CREDIT

CASH                                          5600
   CAPITAL                                                5600

SUPPLIES                                      325
   CASH                                                   325

EQUIPMENT                                     1000
   ACCOUNTS PAYABLE                                       1000

CASH                                          100
   EQUIPMENT                                              100

EQUIPMENT                                     1200
   CASH                                                   1200


TOTALS FOR THIS DATE                          8225        8225

##### YOUR JOURNAL BALANCES #####

WHEN YOU ARE READY FOR THE 'T ACCOUNTS', TYPE GO ?GO
```

```
      ALLISON'S BICYCLE REPAIR

  DEBITS AND CREDITS BY ACCOUNT

       SEPTEMBER 30, 1974

    CASH
--------------------------------
      5600        I
      100         I
                  I       325
                  I       1200

    SUPPLIES
--------------------------------
      325         I

    EQUIPMENT
--------------------------------
      1000        I
      1200        I
                  I       100

    ACCOUNTS PAYABLE
--------------------------------
                  I       1000

    CAPITAL
--------------------------------
                  I       5600

WHEN YOU ARE READY FOR THE BALANCE SHEET, TYPE GO ?GO

      ALLISON'S BICYCLE REPAIR

         BALANCE SHEET

        SEPTEMBER 30, 1974

                ASSETS

CASH                            4175
SUPPLIES                        325
EQUIPMENT                       2100
                             ----------

TOTAL                           6600
                             ----------
                             ----------

             LIABILITIES

ACCOUNTS PAYABLE                1000
                             ----------

TOTAL                           1000

               CAPITAL

CAPITAL                         5600
                             ----------

TOTAL                           5600

TOTAL LIABILITIES AND CAPITAL   6600
                             ----------
                             ----------
```

```
##### CONGRATLATIONS - PROBLEM #  1      IS TOTALLY CORRECT #####

DO YOU WISH TO PROCESS ANOTHER SET OF DATA (Y/N) ?Y
ATG ? SELECTION PROGRAM
PLEASE TELL ME WHICH PROGRAM NUMBER (3-13) YOU WISH TO USE ?11
THANKYOU, YOU WILL BE USING PROGRAM PAY11


COMPUTER ORIENTED ACCOUNTING

STUDENT NAME PLEASE ?LARRY PAGE
DATE PLEASE ?OCTOBER 12, 1974
EXERCISE X-X PLEASE ?6-1

TRANSACTION DATA

COMPANY NAME ?MOORE'S HARDWARE STORE
DATE OF ACCOUNTING PERIOD ?MARCH 7, 1974
WHICH PROBLEM IS THIS ?1

HOW MANY EMPLOYEES DO YOU HAVE ?8

EMPLOYEE # ?1
# EXEMPTIONS ?0
# HOURS ?38
RATE ?2.15
INSURANCE ?1.10
# BONDS ?0

EMPLOYEE # ?2
# EXEMPTIONS ?1
# HOURS ?41
RATE ?2.40
INSURANCE ?.9
# BONDS ?1

EMPLOYEE # ?3
# EXEMPTIONS ?1
# HOURS ?40
RATE ?1.95
INSURANCE ?1.15
# BONDS ?0

EMPLOYEE # ?4
# EXEMPTIONS ?2
# HOURS ?35
RATE ?3
INSURANCE ?.8
# BONDS ?2

EMPLOYEE # ?5
# EXEMPTIONS ?0
# HOURS ?40
RATE ?2.6
INSURANCE ?.65
# BONDS ?0

EMPLOYEE # ?6
# EXEMPTIONS ?3
# HOURS ?44
RATE ?2.75
INSURANCE ?1
# BONDS ?1

EMPLOYEE # ?7
# EXEMPTIONS ?1
# HOURS ?40
RATE ?3.25
INSURANCE ?.70
# BONDS ?1

EMPLOYEE # ?8
# EXEMPTIONS ?2
# HOURS ?40
RATE ?4.5
INSURANCE ?1.2
# BONDS ?2
```

STUDENT NAME - LARRY PAGE
EXERCISE - 6-1
DATE - OCTOBER 12, 1974

MOORE'S HARDWARE STORE
PAYROLL REGISTER
FOR WEEK ENDING MARCH 7, 1974

EMP. NO.	NO. OF EXEM.	NO. OF HOURS	PAY RATE	TOTAL EARNINGS	TOTAL DED.	NET PAY
1	0	38	2.15	81.7	17.28	64.42
2	1	41	2.4	99.6	22.36	77.24
3	1	40	1.95	78	14.57	63.43
4	2	35	3	105	25.06	79.94
5	0	40	2.6	104	21.24	82.76
6	3	44	2.75	126.5	23.75	102.75
7	1	40	3.25	130	28.17	101.83
8	2	40	4.5	180	40.31	139.69
	TOTALS			904.8	192.74	712.06

TYPE GO FOR DEDUCTION REGISTER ?GO

DEDUCTIONS

EMP. NO.	INCOME TAX	FICA TAX	INSUR.	BONDS	TOTAL
1	11.44	4.74	1.1	0	17.28
2	11.93	5.78	.9	3.75	22.36
3	8.9	4.52	1.15	0	14.57
4	10.67	6.09	.8	7.5	25.06
5	14.56	6.03	.65	0	21.24
6	11.66	7.34	1	3.75	23.75
7	16.18	7.54	.7	3.75	28.17
8	21.17	10.44	1.2	7.5	40.31
TOTALS	106.51	52.48	7.5	26.25	192.74

***** VERY GOOD - YOUR TOTALS MATCH MINE *****

DO YOU WISH TO
1 - DO ANOTHER PROBLEM FROM THIS CHAPTER
2 - DO A PROBLEM FROM ANOTHER CHAPTER
3 - STOP NOW
?3

DONE

TITLE: XREF INDEX OF LIBRARY

INDEX
36888-18004

DESCRIPTION:

INDEX is used to print crossreference lists of programs available in the System ($) Library, for any specified range of subject codes, purpose codes, usage codes, method codes, etc. INDEX can also do searches for programs which fit a specified combination of up to 10 codes. Up to 100,000 entries or more, and 999 codes are allowed. The date can be specified, to locate newest items.

INSTRUCTIONS:

There are 4 programs in this package: INDEX, INDEXU, FORMAL and FORMDX.

INDEX allows general inquiry of XREF Index of library

INDEXU allows ID A000 to update files for XREF Index program usage

FORMAL gives standard format for submittals

FORMDX prints out a form for submitting programs

```
DISCFILE "ADDS" USE SPACE MADE BY "DELETES" AND IF DONE, THEN
ADD OCCURS ONTO END OF USED UP SPACE IN FILE.
FIXED SIZE RECORD LAYOUTS (NOT PROTECTED. IN A000 LIBRARY.).
DISC FILES ARE SEQUENTIALLY AND DIRECTLY READ/WRITTEN.

RECORD LAYOUTS: INDEXF- F$,D,C(1)...C(X), 4 RECORDS/BLOCK.
               INDEXC- C$,D,  19 RECORDS/BLOCK.
                INDEXW- WORKFILE FOR UPDATING INDEXF OR INDEXC.

IF FIRST C_LUMN OF INDEXC TITLE IS * THEN TREATED AS TITLE,
INSTEAD OF AS CATEGORY, AND DOUBLE SPACED AUTOMATICALLY, ACROSS.

    AT SYSTEM INITIATION TIME; DO THE FOLLOWING, ONCE ONLY:
  OPEN-INDEXW,1
  OPEN-INDEXC,20   ...IF 380 CATEGORIES; ELSE MORE OR LESS BLOCKS.
  OPEN-INDEXF,200  ...IF 800 PROGRAMS; ELSE MORE OR LESS BLOCKS.
  TYPE:  535 GO TO 9C00
NOTE: MAX 999 CATEGORIES; BUT UNLIMITED NUMBER OF PROGRAMS, IF SPACE.

  VARIABLES.
  C-CODES,D-DATE,S-SERVICE,V-VALID CODES,E-ERROR FLAG(1=YES).
  T-THRESHOLD DATE,F-FOUND FLAG(1=YES,M-MAX CATEGORIES.
X-MAX CATEGORY CODES PER INDEX ENTRY,K-LOOP ETC COUNTER.
  N$-NAME,D$-DESCRIPT,F$-N$+D$,C$-CATEGORY,R$REPLY,R-SEARCH.
  U$-UPDATE TO F$, B$-70 BLANKS, A$-ANSWER, B-BLOCK, Z-LOOP COUNT.
  J-JULIAN DATE FOR UPDATING.
```

For a free printer listing of the author's latest set of continually evolving category codes used for grades 8 to 12, write to Mr. Wayne Dodds, Vancouver School Board, 1595 West 10th Avenue, Vancouver, B.C. Canada.

ACKNOWLEDGEMENTS:

Mr. W. Dodds
Vancouver School Board
Vancouver, B.C.
Canada

RUN

```
RUN
INDEX

DO YOU WANT INSTRUCTIONS?YES
FROM CATEGORY CODES, RELEVANT PROGRAMS IN $LIBRARY CAN BE FOUND.

MOST PROGRAMS SELF EXPLAIN WHEN RUN OR ELSE LISTED.  IF NOT, THEN
REFERENCES(ON HOW TO USE EACH OF THE PROGRAMS) CAN BE
SEEN BY VISITING THE COMPUTER CENTRE; OR ONE COPY PER
TEACHER WILL BE SENT TO SCHOOL, ON REQUEST, IF
POSSIBLE.  PHONE 327-730Ø FOR DETAILS, RESERVATIONS, ETC.

THE INDEX IS ALWAYS BEING IMPROVED.
IF YOU WISH TO PROPOSE CHANGES/NEW DESCRIPTIONS,CODES,CATEGORIES,
OR PROGRAMS; THEN GET-$FORMDX, RUN,& FILL IT IN, SEND TO,
COMPUTER CENTRE, ROOM 219, JOHN OLIVER SEC. SCHOOL.  PLEASE USE
STANDARD FORMAT, SUBMITTED PROGRAMS: GET-$FORMAL & RUN (EXAMPLE).


PAUSES OF ONE MINUTE MAY OCCUR WHILE INDEX IS BEING SEARCHED.
TOTAL NUMBER OF PROGRAMS IN THE INDEX IS NOW 562


SERVICES:
 1 - LIST WHOLE SET OF CATEGORIES AND CODES.
 2 - DO PARTIAL LIST OF CATEGORIES AND CODES.
 3 - LIST XREF INDEX (PARTIAL).  EXAMPLE: WHAT IS UNDER 'MATH' ?
 4 - SEARCH INDEX FOR SPECIFIC CODES. EG: WHAT IS UNDER 'ALGEBRA' ?
 5 - GET DESCRIPTION AND CODES FOR A SPECIFIED PROGRAM NAME.
 6 - LIST ALL PROGRAMS CHANGED/NEW AS OF DATE SPECIFIED.
 7 - END INDEX PROGRAM.
FOR SERVICES 1 TO 4, YOU WILL BE ASKED IF YOU WANT
TO OBTAIN ONLY CHANGED/NEW ITEMS, SPECIFY OLDEST DATE
DESIRED.  USE YYDDD FORMAT.  THUS: 74158 (DAY 158, 1974).

WHAT SERVICE NUMBER(1TO7) DO YOU WANT?3
WHAT START CODE?74158
INVALID.  WHAT START CODE?12
WHAT END CODE?24
WHAT IS OLDEST DATE DESIRED (REPLY Ø IF ALL DATES)?Ø


           LIST OF XREF'D INDEX, FROM CODE 12    TO 24    AFTER DATE Ø
                    UP TO DATE OF: 13     1975


$ NAME     DESCRIPTION OF PROGRAM          CATEGORY CODES THAT APPLY
======     =====================           ========================

*********************************
**   12  -GRADES 4 TO 7        **
*********************************

INDEX   LISTS XREF OF SHARED PROGRAMS IF  12  14  34  52  56
        MEET YOUR SPECIFIED SEARCH CODES

PEGS??  HUMAN TRIES TO RE-ARRANGE PEGS.   12  14  26  42

INDEXU  UPDATES XREF/CATEGORIES. TO USE,  12  14  34  52  56
        GET SUBMISSION FORMS AT COMPUTER

KEYSIG  GIVES MAJOR SCALES.               12  14  26  34 22Ø 221
        12,14,26*

MUSIC5  FINDS DOMINANT SEVENTHS.          12  14  26  34 22Ø 221

JULI    ONE PAGE JULIAN CALENDAR.         12  14  22  47 3Ø6

CALNDR  PRINTS A CALENDAR.                12  14  22  47 3Ø6 3Ø8

TITLE   CHARA CTER GENERATION.            12  14  22  47  62  45
```

```
PRINT   GENERATES LARGE LETTERS.            12  14  22  47  62  45

IDF     INSTRUCTIONAL DIALOGUE FACILITY.    12  14  26  52  54  28

IMF     INSTRUCTIONAL MANAGEMENT            12  14  26  52  54  28
        FACILITY.

SUBRTN  CAI SUBROUTINES.                    12  14  26  52  54  28

IDF-UT  INSTRUCTIONAL DIALOGUE FACILITY.    12  14  26  52  54  28
        UTILITIES PROGRAM.
STOP
RUN
INDEX

DO YOU WANT INSTRUCTIONS?NO


PAUSES OF ONE MINUTE MAY OCCUR WHILE INDEX IS BEING SEARCHED.
TOTAL NUMBER OF PROGRAMS IN THE INDEX IS NOW 562


WHAT SERVICE NUMBER(1TO7) DO YOU WANT?1
WHAT IS OLDEST DATE DESIRED (REPLY Ø IF ALL DATES)?Ø



      CATEGORIES              JULIAN DATE: 13    1975


CODES     CLASSIFICATIONS
=======   ===============




*** S P E C I A L   T E R M I N A L S .    ***

 2        HP LINE PLOTTER.

 3        TEKTRONIX CRT.


*** L A N G U A G E   C O D E D   I N .    ***

 5        ASSEMBLER SIMULATOR.

 6        COBOL SIMULATOR.

 7        FORTRAN SIMULATOR.

 8        IDF FOR CAI.


*** G R A D E   L E V E L :                ***

  12       -GRADES 4 TO 7

 14       GRADES 8 TO 12
  16       -GRADES 8 TO 1Ø.
  18       -GRADES 1Ø TO 12.
```

```
*** S U R V E Y S / T E S T S              ***

  322       -STATISTICS.
  324       -SURVEY ANALYS
  326       -TEST SCORING.


*** U N C L A S S I F I E D :              ***

 332       HP CONTRIBUTED

 334       MISCELLANEOUS.


*** U N I V E R S I T Y :                  ***

 337       ADMINISTRATION

 338       ANTHROPOLOGY.

 340       COMMERCE.

 342       ENGINEERING.

 344       MATHEMATICS.

 346       PSYCHOLOGY.

 347       SCIENCES.

 348       STATISTICS(NON-PARA)

 350       STATISTICS(PARAMET)


CATEGORIES WITHOUT LEFT HYPHEN ARE FOR MISCELLANEOUS.
THEY DO NOT CAUSE THOSE BELOW, WITH HYPHENS, TO BE LISTED.


WHAT SERVICE NUMBER(1TO7) DO YOU WANT?3
WHAT START CODE?250
WHAT END CODE?260
WHAT IS OLDEST DATE DESIRED (REPLY 0 IF ALL DATES)?0


          LIST OF XREF'D INDEX, FROM CODE 250  TO 260  AFTER DATE 0
                    UP TO DATE OF: 13     1975

$ NAME    DESCRIPTION OF PROGRAM          CATEGORY CODES THAT APPLY
======    ======================          =========================

**********************************
**  250  -OPTICS.               **
**********************************

REFRAC  COMPUTER-AUGMENTED PHYSICS         18  26  54 250
        TOPICS.

REFLEC  COMPUTER-AUGMENTED PHYSICS         18  26  54 250
        TOPICS.

SPCTRA  OPTICAL ABSORPTION SPECTRA SIMUL   18  26  50  94 250
        2-SPECIES EQUILIBRIUM MIXTURES.

REFLCT  LEAST TIME PRINCIPLE AND LIGHT.    18  26  50  54 250

SNELL   SNELL'S LAW.                       18  26  50 250

LENSES  SOLVES LENS PROBLEMS.              18  26  32 250  29
```

```
**********************************
**  252  -RELATIVITY.          **
**********************************
```

NO PROGRAMS EXIST FOR CODES/DATES YOU REQUESTED.

```
**********************************
**  254  -SOUND.               **
**********************************
```

```
GET-FORMAL
RUN
FORMAL

ALL PROGRAMS PROPOSED TO BE INCLUDED INTO THE SYSTEM LIBRARY
MUST INCLUDE THE STATEMENTS SHOWN BELOW (STATEMENT NUMBERS MAY
DIFFER, BUT THEY MUST BE THE FIRST FEW STATEMENTS IN THE PROGRAM):

10 REM* AUTHOR:  JOHNNY DOE, MAGEE SCHOOL, APRIL/74.
20 REM* VERSION # 3.
30 REM* MODIFIED: MAY 24/74.
40 REM* INSTRUCTIONS:
50 REM*  TELL BATCH USERS HOW TO USE YOUR PROGRAM.  IF PROGRAM
60 REM*  IS FOR HANDS-ON TYPEWRITER USE ONLY; THEN PROGRAM SHOULD PRINT
70 REM*  OUT INSTRUCTIONS INSTEAD OF PUTTING THEM INTO 'REMARKS'.
80 REM*  IF REFERENCE MATERIALS ARE OPTIONAL OR REQUIRED; THESE
90 REM*  MUST BE INCLUDED IN THE SUBMISSION TO COMPUTER CENTRE.
100 REM*  THE COMPUTER CENTRE WILL THEN HANDLE XEROXING AND
110 REM*  DISTRIBUTION OF SUCH REFERENCE SHEETS TO PROSPECTIVE USERS.
120 REM*  PREFERRABLY, REFERENCE SHEETS SHOULD BE TYPED,
130 REM*  SO AS TO BE EASILY XEROXED (MAKE ANY DRAWINGS BLACK).
REM
REM* VARIABLES USED SHOULD BE SPECIFIED AND THEIR PURPOSE STATED
REM* BRIEFLY; ESPECIALLY FOR PROGRAMS TO BE APPENDED.
REM* FOR PROGRAMS TO BE APPENDABLE, STATEMENTS SHOULD START AT 9000.

DONE
```

```
GET-FORMDX
RUN
FORMDX

S U B M I S S I O N  T O: SYSTEM LIBRARY / INDEX / CATEGORIES.
======================  ==============   =====   ==========

... FILL IN ONLY RELEVANT PARTS (SYS.LIB.AND/OR INDEX, ETC).

F R O M :  SCHOOL...............   TEACHER..............  DATE....

           STUDENT...............   TEACHER SIGNATURE...............

SYSTEM LIBRARY UPDATE:
=====================

YOUR PROGRAM NAME ! . . . . . . !  USER ID ! . . . . !  NEW NAME $.......

- IS THIS A CHANGE TO AN EXISTING PROGRAM IN SYSTEM LIBRARY ? YES NO
  (USE BACK OF THIS SHEET TO EXPLAIN USEFULNESS OF ANY PROGRAM CHANGES.)
- DID YOU SUBMIT EXISTING PROGRAM ? YES NO. -SHOULD IT BE DELETED ?
```

```
XREF INDEX UPDATE:
=================

DESCRIBE PROGRAM (IF NEW/CHANGE): PURPOSE, WHAT IT DOES, USES :

    L1 ! . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . !

    L2 ! . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . !

WHAT CATEGORY CODE NUMBERS DESCRIBE PROGRAM (IF NEW/CHANGED):
    (USE ONE OR MORE CODES TO ANSWER EACH QUESTION BELOW; TEN MAX.)

- WHAT GRADE LEVELS IS IT SUITABLE FOR, MAINLY ?
- WHAT USAGE METHODS CAN BE USED, REASONABLY ?
- WHAT KINDS/PURPOSES APPLY TO THIS PROGRAM, IMPORTANTLY ?
- WHAT SCHOOL TOPICS OR USE MIGHT BE MADE OF THIS PROGRAM ?
- WHAT PROGRAMMING LANGUAGE WAS USED TO WRITE THIS PROGRAM ?

     ! . . !     ! . . !     ! . . !     ! . . !     ! . . !

           ! . . !     ! . . !     ! . . !     ! . . !     ! . . !

CATEGORIES UPDATES:
==================

      CODE  ! . . !  CATEGORY ! . . . . . . . . . . . . . . . . . . . . . !

      CODE  ! . . !  CATEGORY ! . . . . . . . . . . . . . . . . . . . . . !

             (FIRST COLUMN OF 'DETAIL'CATEGORIES MUST BE A HYPHEN - )

R E S U L T :   THIS PROPOSAL WAS/WAS NOT IMPLEMENTED.  DATE.....
                  (THIS LAST LINE TO BE COMPLETED BY COMPUTER CENTRE.)

DONE
```

```
RUN
INDEXU

NEVER CANCEL INDEXU PROGRAM BY USE OF BREAK NOR CONTROL C.
OTHERWISE DISC FILES MAY REMAIN NOT UPDATED, IN UNPREDICTABLE WAY.
PLEASE WAIT DURING 1 MINUTE PAUSES (CAUSED BY DISC SEARCHES).
USE SINGLE LETTER REPLIES TO QUESTIONS OF PURPOSE (FASTER).
USE * CARRIAGE RETURN AS REPLY IF NO LINE #2 DESCRIPTION.
FOR DESCRIPTIONS FILE, CHANGES ARE DONE BY DELETING & ADDING.


CATEGORIES, DESCRIPTIONS, OR END?CATEGORIES
BAD INPUT, RETYPE FROM ITEM 1
??C
ADD, MODIFY, REMOVE, OR END?A
WHAT NEW CODE?885
INVALID.  WHAT NEW CODE?85
NEW CODE ALREADY EXISTS.

CATEGORIES, DESCRIPTIONS, OR END?C
ADD, MODIFY, REMOVE, OR END?M
WHAT OLD CODE?85
WHAT NEW CODE?89
CATEGORY TITLE?EDUCATIONAL ADMINISTRATION
TOO LONG (20 COLUMNS EXCEEDED).
CATEGORY TITLE?EDUC. ADM.

CATEGORIES, DESCRIPTIONS, OR END?END
BAD INPUT, RETYPE FROM ITEM 1
??E

DONE
```

ITEM
36888-18007

TITLE: ITEM: SCORES MULTIPLE CHOICE TESTS

DESCRIPTION:

This program scores multiple choice tests with a maximum of 50 items in the test. Simultaneously an analysis of each item takes place and in the ITEM ANALYSIS section the response frequencies for responses A,B,C,D,E and O (omit) are printed for each item, together with an analysis of the candidates who answered correctly into four groups (upper, middle upper, middle lower and lower) depending on scores for the test as a whole. For example, from item No. 1, of the 18 candidates answering correctly, 6 were in the upper 27% of candidates, 4 lay between the 50th and 72nd percentile inclusive, 5 lay between the 28th and 50th percentiles and 3 were in the lower 27%. The groups are of unequal size so as to conform to the demands of the Kuder-Richardson formula 20 which is used to give a measure of the reliability of the list as a whole.

The facility of each item is printed together with the INDEX of discrimination. The analysis for each item is sufficient for an experienced user to establish the validity of each item, the effectiveness of distractions, etc.

The user is given the opportunity to have the scores standardized onto any mean and standard deviation of his choice. The mean, variance and standard deviation of the number of items correct is printed before the user is given the chance to standardize marks. Standardized scores are printed in ascending order. The identification of students is numerical on order of entry of raw responses.

Specification of statistical techniques used:

N = Number of candidates

K = Number of items in the test

N_H = Number of candidates from upper group who answered the item correctly

N_L = Number of candidates from lower group who answered the item correctly.

J = Number of candidates in the upper 27% group.

Reliability for whole test

$$R = \frac{K}{K-1}\left\{1 - \frac{2J\sum(N_H + N_L) - \sum(N_H + N_L)^2}{0.667\left[\sum(N_H - N_L)\right]^2}\right.$$

Facility for an item $F = \frac{\text{No of correct responses}}{N}$

Discrimination $D = \frac{N_H - N_L}{J}$

INSTRUCTIONS: See following page.

SPECIAL CONSIDERATIONS: See following page.

ACKNOWLEDGEMENTS:

John R. Tilbury
Arnold & Carlton College

INSTRUCTIONS:

Prepare data statements or data tape.

This should contain strings of candidate responses. Each string must contain the number of letter characters - (A,B,C,D,E or O). The first string must be the string of correct responses.

If using DATA statements, each candidate's responses string may utilize an individual DATA statement, or may be packed several response strings to a single data statement. No sentinel string or indicator is necessary.

Data statements should be numbered starting with a sequence number greater than 1580.

If using DATA statements, this must be terminated with an END statement.

CRE-STDT,48: STDT may be opened to more than 48 records if more than 48 students
CRE-STDT (candidates) are involved.

RUN

Program requests number of candidates and number of items.

The rest of the running procedure is straightforward.

If analysis of individual items is not required at some times but is required at others, the user may care to insert the following statements:

```
1001  PRINT "IS ITEM ANALYSIS REQUIRED";
1002  INPUT C$
1003  IF C$ = "NO" THEN 1580
```

Storing the program in a user's private library will thus give the user the choice of using the program to mark lists and standardize scores but not to have items analyzed for validity. This is, however, not the way the program is intended to be used.

SPECIAL CONSIDERATIONS:

This program is only of use to persons experienced in the construction and validation of objective tests. The analysis provided by this program must be interpreted in the context of the uses to which the test is being put and should not be taken as absolute evidence that the test is valid & reliable under all conditions, etc.

RUN

```
CRE-STDT, 48
GET-ITEM
```

```
1590  DATA "ABAEDCEACB"
1600  DATA "ABADDCEABA"
1610  DATA "ABAEDCEACB"
1620  DATA "ABBCDBEBAA"
1630  DATA "ABABDCEACO"
1640  DATA "ABEDDAEBCA"
1650  DATA "OBAADCEACC"
1660  DATA "ABDBDEEBAB"
1670  DATA "ABACDOACDA"
1680  DATA "AOCDDCEACC"
1690  DATA "CAAADAEBCD"
1700  DATA "ABOBDBEAAE"
1710  DATA "ABABDCEACB"
1720  DATA "ABBDDEEBBD"
1730  DATA "OEAAOOBBOC"
1740  DATA "ABCBDCEAEA"
1750  DATA "EBACDAEACB"
1760  DATA "ABDDDBEBCA"
1770  DATA "CBAADOBBBA"
1780  DATA "ABEBDCEACB"
1790  DATA "ADACDAEBAB"
1800  DATA "EBODDBEBCA"
1810  DATA "ABAEDCEACO"
1820  DATA "CBBBDAEBOB"
1830  DATA "ABACDBEACB"
1840  DATA "ABCDDOEBCB"
```

```
RUN
ITEM

ENTER NUMBER OF STUDENTS INVOLVED IN TEST?25
 HOW MANY ITEMS IN THE TEST?10
STDT.NO.        RIGHT/WRONG/OMIT              SCORE
 1               7      3      0               .7
 2               10     0      0               1
 3               4      6      0               .4
 4               8      1      1               .8
 5               5      5      0               .5
 6               7      2      1               .7
 7               5      5      0               .5
 8               4      5      1               .4
 9               6      3      1               .6
 10              4      6      0               .4
 11              5      4      1               .5
 12              9      1      0               .9
 13              4      6      0               .4
 14              1      5      4               .1
 15              6      4      0               .6
 16              7      3      0               .7
 17              5      5      0               .5
 18              3      6      1               .3
 19              8      2      0               .8
 20              5      5      0               .5
 21              4      5      1               .4
 22              9      0      1               .9
 23              4      5      1               .4
 24              8      2      0               .8
 25              6      3      1               .6


LOWER GROUP OF 7     WITH NUMBER OF ITEMS CORRECT
 14     1
 18     3
 3      4
 8      4
 10     4
 13     4
 21     4


TOP GROUP WITH NUMBER OF ITEMS CORRECT
 16     7
 4      8
 19     8
 24     8
 12     9
 22     9
 2      10
MEAN= 5.76        VARIANCE= 4.42239     STD. DEV.= 2.10295


DO YOU WANT STANDARDISED SCORES?YES
ENTER MEAN AND STANDARD DEVIATION ONTO WHICH YOU
WISH THE RESULTS TO BE STANDARDISED?50,15
```

NO.	SCORE
14	16
18	30
3	37
8	37
10	37
13	37
21	37
23	37
5	44
7	44
11	44
17	44
20	44
9	51
15	51
25	51
1	58
6	58
16	58
4	65
19	65
24	65
12	73
22	73
2	80

```
ITEM ANALYSIS
QUESTION NO. 1    ANSWER A
RESPONSE FREQUENCIES 18     0      3      0      2      2
UPPER= 6    MIDDLE UPPER= 4    MIDDLE LOWER= 5    LOWER= 3
FACILITY= .72          DISCRIMINATION= .428571


QUESTION NO. 2    ANSWER B
RESPONSE FREQUENCIES 1      21     0      1      1      1
UPPER= 7    MIDDLE UPPER= 4    MIDDLE LOWER= 5    LOWER= 5
FACILITY= .84          DISCRIMINATION= .285714


QUESTION NO. 3    ANSWER A
RESPONSE FREQUENCIES 13     3      3      2      2      2
UPPER= 6    MIDDLE UPPER= 2    MIDDLE LOWER= 1    LOWER= 4
FACILITY= .52          DISCRIMINATION= .285714


QUESTION NO. 4    ANSWER E
RESPONSE FREQUENCIES 4      7      5      7      2      0
UPPER= 2    MIDDLE UPPER= 0    MIDDLE LOWER= 0    LOWER= 0
FACILITY= .08          DISCRIMINATION= .285714


QUESTION NO. 5    ANSWER D
RESPONSE FREQUENCIES 0      0      0      24     0      1
UPPER= 7    MIDDLE UPPER= 5    MIDDLE LOWER= 6    LOWER= 6
FACILITY= .96          DISCRIMINATION= .142857
```

```
QUESTION NO. 6    ANSWER C
RESPONSE FREQUENCIES 5     5     9     Ø     2     4
UPPER= 5    MIDDLE UPPER= 4    MIDDLE LOWER= Ø    LOWER= Ø
FACILITY= .36         DISCRIMINATION= .714286
```

```
QUESTION NO. 7    ANSWER E
RESPONSE FREQUENCIES 1     2     Ø     Ø     22    Ø
UPPER= 7    MIDDLE UPPER= 5    MIDDLE LOWER= 6    LOWER= 4
FACILITY= .88         DISCRIMINATION= .428571
```

```
QUESTION NO. 8    ANSWER A
RESPONSE FREQUENCIES 12    12    1     Ø     Ø     Ø
UPPER= 7    MIDDLE UPPER= 4    MIDDLE LOWER= 1    LOWER= Ø
FACILITY= .48         DISCRIMINATION= 1
```

```
QUESTION NO. 9    ANSWER C
RESPONSE FREQUENCIES 4     3     14    1     1     2
UPPER= 7    MIDDLE UPPER= 3    MIDDLE LOWER= 2    LOWER= 2
FACILITY= .56         DISCRIMINATION= .714286
```

```
QUESTION NO. 1Ø   ANSWER B
RESPONSE FREQUENCIES 8     9     3     2     1     2
UPPER= 5    MIDDLE UPPER= 1    MIDDLE LOWER= 3    LOWER= Ø
FACILITY= .36         DISCRIMINATION= .714286
```

```
RELIABILITY  KR(2Ø)  = .5Ø5211

DONE
```

TITLE: SRP: STUDENT RESPONSE PRINT

DESCRIPTION:

The Student Response Print (SRP) is a program designed to read response files which are written by the Hewlett-Packard IDF program IDSF (Instructional Dialogue Student Facility), and print the responses in two formats:

A. Standard Response Print

The standard response print lists the responses for each student by section number.

B. Response Frequency Count

The Response Frequency Count print eliminates all duplicate responses and prints only unique responses and their associated frequency of occurrence by section number.

INSTRUCTIONS:

The SRP uses one disc file (SKR2) as a scratch file. The SKR2 file is routinely created as part of the IDF system. If the SKR2 file has not been created it should be opened in the following manner:

```
CRE-SKR2,32
```

SRP contains a variety of features which are designed to increase the program's flexibility and usefulness.

A. "Batch" Processing Feature

The user often must dump a number of response files. This can be a time consuming task if the number of sections are small, and if there are many files to dump, since the user must "baby sit" the computer terminal, dumping one file, then the next, etc. The SRP allows the user to enter from one to twelve filenames to be dumped, the SRP will dump the files in a continuous fashion.

B. Remove Blanks Option

IDSF answer checking is often performed after removal of embedded blanks from student answers, but the student answers are written into the response file without blanks removed. With the REMOVE BLANKS option, the user can print the responses without blanks, thus making the responses appear as they appear to the answer checking section of IDSF. If blanks are removed, the user can readily determine if this answer checking is working properly by comparing the print to the appropriate section.

C. Frequency Counts of //CALC, //HINT, and //STOP

Since it is often useful to determine the student use of CAI facilities, such as the calculator, the SRP does frequency counts of the 3 major IDF aids: //CALC, //HINT, and //STOP.

Continued on following page.

ACKNOWLEDGEMENTS: Verl Dennis and Daniel Krautheim
Office of Educational Development
The Ohio State University

INSTRUCTIONS continued.

D. Selective Section Dumps

If only one response filename is entered into the SRP, the program allows the user to selectively dump responses of any given section or sections.

E. Response File Protection

The user must enter the code $R_c U_c N_c$ before the program will permit a user to dump a response file/s.

RUN

```
RUN

SRP

IDF STUDENT RESPONSE PRINT    ***  OED VERSION B
CODE? RcUcNc
ENTER RESPONSE FILE LIST, ONE PER LINE,  TERMINATE WITH 'END'
FILE # 1     ?FRANCR
FILE # 2     ?END
CORRECTIONS?N


REMOVE BLANKS?Y

SELECT:  A. STD RESPONSE PRINT   B. RESPONSE FREQ COUNT?A
THERE ARE  6    SECTIONS IN FRANCR
START SRP AT SECTION #?1
STOP SRP AT SECTION #?6
------

FEB   5, 1975      RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 1        FRANCR       STUDENT RESPONSE PRINT


   S#          RESPONSE
  ====      ========================================

  1010       POMPIDOU
  1011       DESTANE
  1011       POMPDIOU
  1012       POMPIDOU
  1013       POMPIDOU
  1014       POMPIDOU
  1015       DESTANG
  1015       DEGAULLE
  1015       PICKARD
  1016       POMPIDOU
  1017       POMPDIOU
  1018       POMPIDOU
  1019       DESTANG
  1019       DEGALLE
  1019       LEGRANGE

NUMBER OF STUDENTS STOPPING IN THIS SECTION --  0
NUMBER OF //HINTS REQUESTED --  3
NUMBER OF //CALCS USED      --  0
```

```
FEB  5, 1975     RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 2        FRANCR      STUDENT RESPONSE PRINT


   S#         RESPONSE
  ====      ========================================

  1010     DEGAULLE
  1011     DEGAULLE
  1012     DEGAULL
  1013     DEGAULLE
  1014     DEGALL
  1015     DEGAULLE
  1016     DEGAULLE
  1017     DEGAULLE
  1018     DEGAULLE
  1019     DEGALL

NUMBER OF STUDENTS STOPPING IN THIS SECTION --  0
NUMBER OF //HINTS REQUESTED --  2
NUMBER OF //CALCS USED       --  0


------

FEB  5, 1975     RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 3        FRANCR      STUDENT RESPONSE PRINT


   S#         RESPONSE
  ====      ========================================

  1010     ROMANCATHOLIC
  1011     CATHOLIC
  1012     ROMANCATHOLIC
  1013     JEWISH
  1013     CATHOLIC
  1014     CATHOLIC
  1015     ROMANCATHOLIC
  1016     CATHOLIC
  1017     CATHOLIC
  1018     ROMANCATHOLIC
  1019     JEWISH
  1019     ITALIAN
  1019

NUMBER OF STUDENTS STOPPING IN THIS SECTION --  0
NUMBER OF //HINTS REQUESTED --  4
NUMBER OF //CALCS USED       --  0

------

FEB  5, 1975     RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 4        FRANCR      STUDENT RESPONSE PRINT


   S#         RESPONSE
  ====      ========================================

  1010     ALPS
  1011     ALPS
  1012     ALPS
  1013     ALPS
  1014     ALPS
  1015     ALPS
  1016     ALPS
  1017     ALPS
  1018     ALPS
  1019     ALPS

NUMBER OF STUDENTS STOPPING IN THIS SECTION --  0
NUMBER OF //HINTS REQUESTED --  1
NUMBER OF //CALCS USED       --  0
```

```
------

FEB   5, 1975      RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 5        FRANCR       STUDENT RESPONSE PRINT


   S#           RESPONSE
 ====       ========================================

 1010      SEINE
 1011      GARONNE
 1012      THESEINE
 1013      LOIRE
 1014      RHONE
 1015      SEINE
 1016      GARONNE
 1017      //RHONE
 1017      RHONE
 1018      RHONE
 1019      GARDEN

NUMBER OF STUDENTS STOPPING IN THIS SECTION --   0
NUMBER OF //HINTS REQUESTED --   2
NUMBER OF //CALCS USED       --   0

------

FEB   5, 1975      RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 6        FRANCR       STUDENT RESPONSE PRINT


   S#           RESPONSE
 ====       ========================================

 1010      NO
 1011      NO
 1012      NO
 1013      NO
 1014      NO
 1015      NO
 1016      NO
 1017      NO
 1018      NO
 1019      NEVER
 1019      NO

NUMBER OF STUDENTS STOPPING IN THIS SECTION --   0
NUMBER OF //HINTS REQUESTED --   0
NUMBER OF //CALCS USED       --   0



GET-$SRP
RUN
SRP

IDF STUDENT RESPONSE PRINT    ***  OED VERSION B
CODE? RCCNC
ENTER RESPONSE FILE LIST, ONE PER LINE,  TERMINATE WITH 'END'
FILE # 1     ?FRANCR
FILE # 2     ?END
CORRECTIONS?N


REMOVE BLANKS?Y

SELECT:  A. STD RESPONSE PRINT   B. RESPONSE FREQ COUNT?B
THERE ARE  6     SECTIONS IN FRANCR
START SRP AT SECTION #?5
STOP SRP AT SECTION #?6
------
```

```
FEB   5, 1975     RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 5        FRANCR    RESPONSE FREQUENCY COUNT


 FREQ             RESPONSE
 ====  =============================================

 2     SEINE
 2     GARONNE
 1     THESEINE
 1     LOIRE
 3     RHONE
 1     //RHONE
 1     GARDEN

NUMBER OF //STOPS     --  Ø
NUMBER OF //HINTS     --  2
NUMBER OF //CALCS     --  Ø



------

FEB   5, 1975     RESP FILE ASSOCIATED WITH LESSONFILE: FRANC

SECTION # 6        FRANCR    RESPONSE FREQUENCY COUNT


 FREQ             RESPONSE
 ====  =============================================

 1Ø    NO
 1     NEVER

NUMBER OF //STOPS     --  Ø
NUMBER OF //HINTS     --  Ø
NUMBER OF //CALCS     --  Ø



------

DONE
```

Documentation Date 3/75

EDUCATIONAL ADMINISTRATION (885)

CONTRIBUTED PROGRAM **BASIC**

TESTGD
36888-18041

TITLE: MULTIPLE-CHOICE TEST GRADER

DESCRIPTION:

This program is used to score multiple choice tests of up to 40 questions in length. Output includes an alphabetical list of the students with a listing of the answers they had incorrect. Correct responses are listed as "." except every fifth is a ",", double marked answers are listed as ":" and blanks are "   ". On the next line is the number of correct answers, the number of incorrect, and the percentage of the maximum. An item analysis of answers completes the report.

INSTRUCTIONS:

Both the students' responses and the answer key must be marked on 40 column mark sense cards. A file TEST must be opened to at least (N+4)/3 records in length where N is the number of students, 140 maximum. The card deck must be made up of alternating "NAME=name-of-student" cards and the corresponding response card. The first card must be "NAME=KEY", followed by the correct answer card, and then the pairs of student cards, not necessarily in alphabetical order. The deck must end with an "END" card.

Upon running TESTGD the computer will ask for the number of questions and then will start reading the cards. If the card reader is not ready at the start, push the READY button. If the card reader drops the ready status, you may have to start over.

If for some reason the program is interrupted after the END card has been read, there are two re-start options: After giving the number of questions, "SORT" will restart the program at the sorting phase and "PRINT" will re-start the program at the printing phase.

Equipment required is: TSB/2000E system, hard-copy terminal, and HP 7260A Optical Mark Reader with the image option.

SPECIAL CONSIDERATIONS:

Leaving the first column blank will mix up the answers to questions 36-40. Marking rows 12 and 2 or 4 and 8 in the same column will falsify any questions after that one.

ACKNOWLEDGEMENTS:

Lary R. Smith
Livonia Public Schools