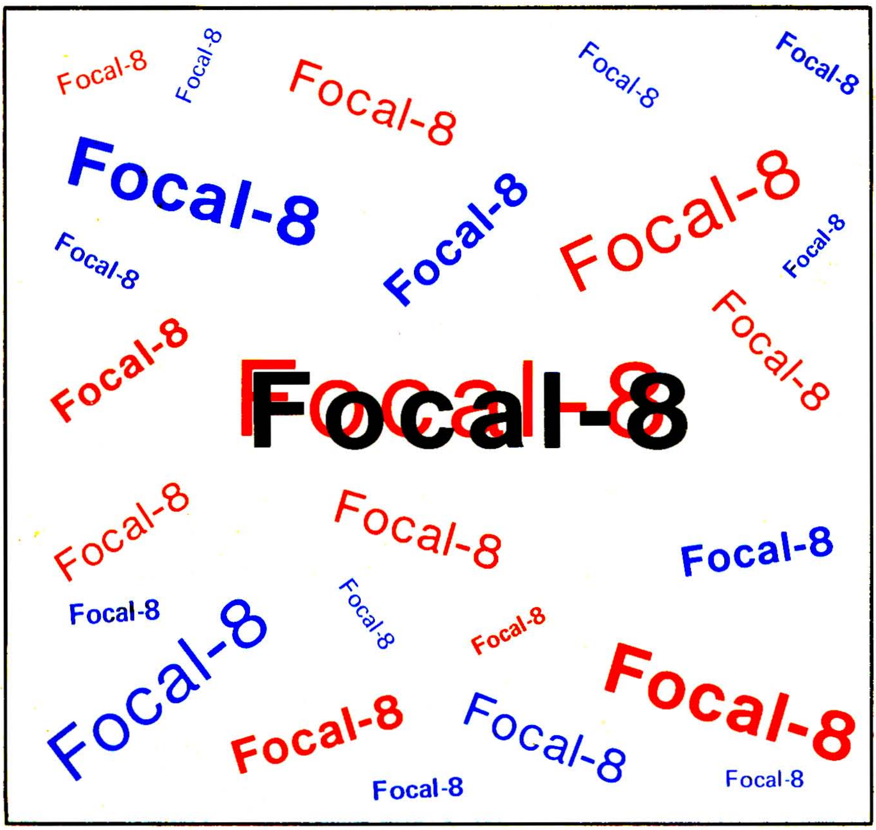
DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS

digital

Demonstration Programs 1970



FOCAL-8

DEMONSTRATION PROGRAMS

FOR PDP-8/I AND PDP-8/E

For additional copies, order No. DEC-08-XJFA-D from Program Library, Digital Equipment Corporation, Maynard, Massachusetts 01754 Price: $2.50

DIGITAL EQUIPMENT CORPORATION • MAYNARD, MASSACHUSETTS

1st Printing, July 1970

FOREWORD

FOCAL® is a conversational language developed by Digital Equipment Corporation for its PDP-8 family of small computers.

INTRODUCTION

FOCAL-8 Demonstration Programs: illustrates some features and applications of FOCAL as a conversational language; aids the FOCAL student to gain a significant amount of knowledge from studying the techniques that were used to solve each routine; and, satisfies those "computerniks" who just "enjoy"!

CONTENTS

®Teletype is a registered trademark of the Teletype Corp.

The following are referenced throughout the text.

*With or without extended functions.

**With extended functions.

***Without extended functions.

CONTENTS (Cont)

CONTENTS (Cont)

I. DISPLAY ROUTINES

Circle Plot

ABSTRACT

This FOCAL-8 demonstration routine plots a circle on the Teletype for a specific radius. The user must input the radius when requested. For best results the radius should be no larger than 10.

Operational Procedures

1. Load *Circle Plot* by FOCAL-8.*

2. Type *GO* and input the radius when requested. Execution begins.

3. A sample run follows.

```
*W
C-FOCAL,1969

10.10 ASK "WHAT IS THE RADIUS?"A; DO 12.2
10.20 QUIT

11.06 C A GOOD RADIUS IS 10 OR LESS
11.10 SET R=FABS[ ( <X↑2+Y↑2>↑2-A↑2*(X↑2-Y↑2) )]/A↑2
11.20 IF (R - 7.05) 11.5 ;TYPE " "; RETURN
11.50 TYPE "*"

12.10 FOR X=-10,.5,+10; DO 11
12.20 FOR Y=-5,.5,5; TYPE !; DO 12.1
*
```

```
G
WHAT IS THE RADIUS?:7
```

```
     ****           ****
  *************************
 ***************************
****  *****************  ****
**      *************      **
**      *************      **
**       ***********       **
**      *************      **
**      *************      **
****  *****************  ****
 ***************************
  *************************
     ****           ****
```

```
*GO
WHAT IS THE RADIUS?:10
```

```
        *                       *
    **********             **********
   **************       **************
  **       *******************       **
 **          ***************          **
**            *************            **
**            *************            **
*              ***********              *
*              ***********              *
*              ***********              *
**            *************            **
**            *************            **
 **          ***************          **
  **       *******************       **
   **************       **************
    **********             **********
        *                       *


                                         *
```

Figure-8

ABSTRACT

The *Figure-8* program is a Teletype display routine, which will type a figure "8" after the user has specified a radius between 5 and 25.

Operational Procedures

1. Load the *Figure-8* routine with FOCAL as described in the FOCAL manual.**

2. After the initial dialogue, the user responds to the request of the computer for specific radius (between 5 and 25).

3. The routine immediately begins to type the requested size of the figure "8".

4. A sample run follows.

```
*W
C-FOCAL,1969

09.10 T !!" COMPUTERS CAN BE USED TO PLOT PICTURES OF THINGS."!
09.20 T "THIS PROGRAM WILL PLOT A FIGURE EIGHT WITH A RADIUS
09.30 T !"BETWEEN 5 AND 25 ; YOU MAY SELECT THE SIZE."!
09.40 T "USE A LEFT ARROW TO CORRECT YOUR NUMBER AND TYPE
09.50 T !" A 'RE-TURN' TO PLOT."!!

10.10 ASK "WHAT IS THE RADIUS? "A;SET B=A↑2; DO 12.2
10.20 T !!!;GO

11.10 SET R= FABS[ ( <D+C>↑2 - B*(D-C) )]/B
11.20 IF (R - A*2/3)  11.5 ;TYPE " "; RETURN
11.50 TYPE "8"

12.10 FOR Y=-A/2,.5,+A/2;SET C=Y↑2;SET D=X↑2; DO 11
12.20 FOR X=-A,1,A; TYPE !; DO 12.1
*
```

```
*GO


 COMPUTERS CAN BE USED TO PLOT PICTURES OF THINGS.
THIS PROGRAM WILL PLOT A FIGURE EIGHT WITH A RADIUS
BETWEEN 5 AND 25 ; YOU MAY SELECT THE SIZE.
USE A LEFT ARROW TO CORRECT YOUR NUMBER AND TYPE
 A 'RE-TURN' TO PLOT.

WHAT IS THE RADIUS? :15

            8888888
        888         888
       8               8
     88                 88
     8                   8
    88                   88
    88                   88
    88                   88
    888                 888
     888               888
      888             888
       8888         8888
       88888888888888888
        888888888888888
         8888888888888
         8888888888888
         8888888888888
        888888888888888
       88888888888888888
       8888         8888
      888             888
     888               888
    888                 888
    88                   88
    88                   88
    88                   88
     8                   8
     88                 88
       8               8
        888         888
            8888888
```

Sine

ABSTRACT

Sine demonstrates the flexibility and adaptability of the ASR-33 Teletype to plot a given figure. The result of this routine is a reasonable looking plot. Therefore, one can conclude that the result of a specific function would be quite accurate.

Operational Procedures

1. Load *Sine* via FOCAL-8.**

2. Type *GO* and the routine immediately begins to plot. To interrupt plotting, type a control-C (↑C).

3. A sample run follows.

```
*W
C-FOCAL, 1969

01.10 SET A=A+6*1.5; SET X=A; SET SC=3; DO 3; TYPE "*"#
01.20 SET B=B+6*3; SET X=B; SET SC=2; DO 3; TYPE "."#
01.30 SET DD=DD+6*2; SET X=DD; SET SC=1; DO 3; TYPE "X"!

02.10 SET SINE=FSIN(X/57.2958)

03.10 DO 2; FOR J=0,17+5*SC*SINE; TYPE " "

10.10 FOR I=1,45; DO 1
*
```

```
*GO
                   X*.
                    X  .                 *
                    X   * .
                     X    *.
                      X    .*
                      X    .  *
                      X   .    *
                      X.        *
                     .X         *
                  .   X         *
              .      X          *
            .       X           *
         .          X          *
        .          X          *
        .         X         *
        .       X         *
         .     X        *
            .  X      *
              X     *
             X   .*
             X *     .
             X         .
           * X            .
         *   X             .
       *     X             .
     *        X            .
    *          X          .
   *           X       .
   *            X    .
   *             X.
   *          .    X
   *        .       X
    *    .          X
     *  .            X
       *.             X
        .*            X
         . *          X
            .*        X
              .*      X
                 *    X
                    *X
                    X *.
                    X   * .
                   X      *.
                  X        .*
```

II. GAMES AND QUIZZES

Dice Game

ABSTRACT

***Dice Game*, a FOCAL-8 demonstration program, simulates an actual game of throwing dice. A $1,000-house limit is set, $1-minimum bet for each throw is required.**

Operational Procedures

1. Load *Dice Game* via FOCAL-8 either with or without the extended functions.

2. Type *GO* and the game begins.

3. A sample run follows.

```
*W
C-FOCAL,1969

Ø1.1Ø S B=Ø;T !!"DICE GAME"!,"HOUSE LIMIT OF $1ØØØ
Ø1.13 T ".  MINIMUM BET IS $1"!!
Ø1.2Ø ASK "YOUR BET IS"A;IF (1ØØØ-A) 3.1
Ø1.22 I (A-1)3.4,1.26,1.26
Ø1.26 I (A-FITR(A))3.5,1.3,3.5
Ø1.3Ø ASK M;DO 2;S D=C;DO 2;T " ";S D=D+C
Ø1.32 I (D-7)1.42,3.2,1.42
Ø1.4Ø I (D-2)1.5,3.3,1.5
Ø1.42 I (D-11)1.4,3.2,1.4
Ø1.5Ø I (D-3) 1.6,3.3,1.6
Ø1.6Ø ASK M;DO 2;S E=C;DO 2;T " ";S E=E+C
Ø1.72 I (E-7) 1.74,3.3,1.74
Ø1.74 I (E-D)1.6,3.2,1.6

Ø2.1Ø S C=FITR(2Ø*FRAN()-1Ø);I (C-6)2.2,2.2,2.1
Ø2.2Ø I (C-1)2.1;T %1,"  "C;RETURN

Ø3.1Ø T "HOUSE LIMITS ARE $1ØØØ"!!; G 1.2
Ø3.2Ø S B=B+A;T %7,!"YOU WIN. ";G 3.4
Ø3.3Ø S B=B-A;T %7,!"SORRY, YOU LOSE.
Ø3.4Ø T " YOUR WINNINGS ARE "B,!!;GOTO 1.2
Ø3.5Ø T "NO PENNIES, PLEASE"!!;GOTO 1.2
*
```

```
*GO


DICE GAME
HOUSE LIMIT OF $1000.  MINIMUM BET IS $1

YOUR BET IS:45.50
NO PENNIES, PLEASE

YOUR BET IS:50
:
  = 4   = 2 :
  = 5   = 1
YOU WIN.  YOUR WINNINGS ARE =       50

YOUR BET IS:50
:
  = 3   = 6 :
  = 2   = 4 :
  = 1   = 3 :
  = 6   = 2 :
  = 4   = 3
SORRY, YOU LOSE.  YOUR WINNINGS ARE =        0

YOUR BET IS:50
:
  = 5   = 1 :
  = 4   = 2
YOU WIN.  YOUR WINNINGS ARE =       50

YOUR BET IS:50
:
  = 5   = 1 :
  = 3   = 6 :
  = 2   = 5
SORRY, YOU LOSE.  YOUR WINNINGS ARE =        0


*GO


DICE GAME
HOUSE LIMIT OF $1000.  MINIMUM BET IS $1

YOUR BET IS:50
:4
  = 2   = 4 :4
  = 3   = 6 :
  = 1   = 4 :
  = 2   = 5
SORRY, YOU LOSE.  YOUR WINNINGS ARE =-      50

YOUR BET IS:50
:
  = 1   = 4 :
  = 2   = 5
SORRY, YOU LOSE.  YOUR WINNINGS ARE =-     100
```

```
*GO


DICE GAME
HOUSE LIMIT OF $1000.  MINIMUM BET IS $1

YOUR BET IS:50
:
  = 1  = 3 :
  = 6  = 2 :
  = 4  = 3
SORRY, YOU LOSE.  YOUR WINNINGS ARE =-      50

YOUR BET IS:50
:
  = 6  = 1
YOU WIN.  YOUR WINNINGS ARE =        0

YOUR BET IS:35
:
  = 4  = 2 :
  = 5  = 1
YOU WIN.  YOUR WINNINGS ARE =       35


*
```

King of Sumeria

ABSTRACT

The *King of Sumeria* is a game that challenges your ability to foresee the consumer market. Hamurabi, your servant, will state the following facts about last year, and you must decide the number of acres you will need, and how many bushels of grain you expect to distribute as food. You will base your decisions on these facts:

a. **Number of people who died of starvation**
b. **Number of new people who came to the city**
c. **Number of acres owned by the city**
d. **Number of bushels harvested per acre**
e. **Total number of bushels that were harvested**
f. **Number of bushels that were destroyed**
g. **Number of bushels currently in storage.**

Based on your decisions, Hamurabi will state a new report of the above information.

Operational Procedures

1. Load FOCAL-8, without extended functions.***

2. Load the *King of Sumeria* according to the loading instructions for papertape in the FOCAL-8 manual.

3. Type *GO* and the game begins.

4. A sample run follows.

```
**
*************************************************
*WRITE ALL
C-FOCAL,1969

01.10 S P=95;S S=2800;S H=3000;S E=200;S Y=3;S A=1000;S I=5;S Q=1

02.20 D 6;T !!!"LAST YEAR"!D," STARVED,
02.25 T !I," ARRIVED,";S P=P+I;I (-Q)2.3
02.27 S P=FITR(P/2);T !"**PLAGUE**"!
02.30 T !"POPULATION IS"P,!!"THE CITY OWNS
02.35 T A," ACRES."!!;I (H-1)2.5;T "WE HARVESTED
02.40 D 3.2
02.50 T !" RATS ATE "E," BUSHELS,YOU NOW HAVE
02.60 T !S," BUSHELS IN STORE."!

03.10 D 6;D 8;S Y=C+17;T "LAND IS TRADING AT
03.20 T Y," BUSHELS PER ACRE;";S C=1
03.30 D 4.3;A " BUY?"!Q;I (Q)7.2,3.8
03.40 I (Y*Q-S)3.9,3.6;D 4.6;G 3.3
03.50 D 4.5;G 3.3
03.60 D 3.9;G 4.8
03.70 S A=A+Q;S S=S-Y*Q;S C=0
03.80 A !"TO SELL?"!Q;I (Q)7.2,3.9;S Q=-Q;I (A+Q)3.5
03.90 S A=A+Q;S S=S-Y*Q;S C=0

04.10 T !"BUSHELS TO USE
04.11 A " AS FOOD?"!Q;I (Q)7.2;I (Q-S)4.2,4.7;D 4.6;G 4.1
04.20 S S=S-Q;S C=1
04.30 A !"HOW MANY ACRES OF LAND DO YOU WISH TO
04.35 A !"PLANT WITH SEED? "D
04.40 I (D)7.2;I (A-D)4.45;I (FITR(D/2)-S-1)4.65;D 4.6;G 4.3
04.45 D 4.5;G 4.3
04.50 D 7;T A," ACRES."!
04.60 D 7;D 2.6
04.65 I (D-10*P-1)5.1;D 7;T P," PEOPLE."!;G 4.3
04.70 D 4.2
04.80 D 6;T "YOU HAVE NO GRAIN LEFT AS SEED !!"!S D=0

05.10 S S=S-FITR(D/2);D 8;S Y=C;S H=D*Y
05.20 D 8;S E=0;I (FITR(C/2)-C/2)5.3;S E=S/C
05.30 S S=S-E+H;D 8;S I=FITR(C*(20*A+S)/P/100+1);S C=FITR(Q/20)
05.40 S Q=FITR(10*FABS(FRAN()));I (P-C)2.1;S D=P-C;S P=C;G 2.2

06.10 T !!"HAMURABI:  "%5

07.10 I (C)7.2;S C=C-1;D 6;T "BUT YOU HAVE ONLY";R
07.20 D 6;T !"GOODBYE!"!!;E A

08.10 S C=FITR(5*FABS(FRAN()))+1
*
```

```
*GO


HAMURABI:


LAST YEAR
=      0 STARVED,
=      5 ARRIVED,
POPULATION IS=   100

THE CITY OWNS=  1000 ACRES.

WE HARVESTED=      3 BUSHELS PER ACRE;
 RATS ATE =   200 BUSHELS,YOU NOW HAVE
=  2800 BUSHELS IN STORE.


HAMURABI:  LAND IS TRADING AT =    21 BUSHELS PER ACRE;
HOW MANY ACRES OF LAND DO YOU WISH TO BUY?
:50

BUSHELS TO USE AS FOOD?
:1000

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :500


HAMURABI:


LAST YEAR
=     50 STARVED,
=      8 ARRIVED,
POPULATION IS=    58

THE CITY OWNS=  1050 ACRES.

WE HARVESTED=      4 BUSHELS PER ACRE;
 RATS ATE =     0 BUSHELS,YOU NOW HAVE
=  2500 BUSHELS IN STORE.


HAMURABI:  LAND IS TRADING AT =    22 BUSHELS PER ACRE;
HOW MANY ACRES OF LAND DO YOU WISH TO BUY?
:0

TO SELL?
:0

BUSHELS TO USE AS FOOD?
:1000

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :500
```

```
HAMURABI:


LAST YEAR
=      8 STARVED,
=     18 ARRIVED,
POPULATION IS=     68

THE CITY OWNS=  1050 ACRES.

WE HARVESTED=      5 BUSHELS PER ACRE;
 RATS ATE =      0 BUSHELS,YOU NOW HAVE
=  3750 BUSHELS IN STORE.


HAMURABI:  LAND IS TRADING AT =     20 BUSHELS PER ACRE;
HOW MANY ACRES OF LAND DO YOU WISH TO BUY?
:75

BUSHELS TO USE AS FOOD?
:1000

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :1000


HAMURABI:  BUT YOU HAVE ONLY=     68 PEOPLE.

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :1000


HAMURABI:  BUT YOU HAVE ONLY=     68 PEOPLE.

HOW MANY ACRES OF LAND DO YOU WISH TO
PLANT WITH SEED? :500


HAMURABI:


LAST YEAR
=     18 STARVED,
=     19 ARRIVED,
POPULATION IS=     69

THE CITY OWNS=  1125 ACRES.

WE HARVESTED=      4 BUSHELS PER ACRE;
 RATS ATE =    250 BUSHELS,YOU NOW HAVE
=  2750 BUSHELS IN STORE.


HAMURABI:  LAND IS TRADING AT =     21 BUSHELS PER ACRE;
HOW MANY ACRES OF LAND DO YOU WISH TO BUY?
:
```

Literature Quiz

ABSTRACT

READ: "to understand the meaning of (written or printed matter)"

The above is just one of the many definitions given in "Webster's Seventh New Collegiate Dictionary" that defines this art of communication. It is very important to understand exactly what you read. Therefore, from a very early age a child should be asked to explain or relate events that were read to him or what he read. Although this quiz is very basic, it may be used to introduce both the computer and comprehensive reading to the student. The quiz also reminds us that computers may be used not only for mathematics, but for a variety of things.

The quiz itself is multiple choice. The questions were derived from many well-known nursery tales. For fun, see how much you remember or how much your children remember about them.*

```
*W
C-FOCAL,1969

01.10 T !"TEST YOUR KNOWLEDGE IN CHILDREN'S LITERATURE."
01.20 T !"THIS IS A MULTIPLE CHOICE QUIZ."
01.30 T !"TYPE EITHER A 1, 2, 3, OR 4 AFTER THE COLON. IF
01.40 T !"YOU FAIL, THE CORRECT ANSWER WILL BE TYPED.
01.50 T !"GOOD LUCK!"!;GOTO 2.1

02.10 DO 4.0;DO 5.0;DO 6.0
02.20 DO 7.0;DO 8.0;DO 9.0
02.30 T "THE END!!";QUIT
02.40
02.50

04.10 S A=3;T !"IN 'PINOCCHIO', WHAT WAS THE NAME OF THE CAT?"
04.20 T !"1)TIGGER, 2)CICERO, 3)FIGARO, 4)GUIPETTO";ASK A,%1
04.30 IF (A-3)4.4,4.5,4.4
04.40 T !"SORRY- FIGARO WAS HIS NAME.";RETURN
04.50 T !"VERY GOOD, HERE IS ANOTHER QUESTION FOR YOU!"

05.10 S A=2; T !"FROM WHOSE GARDEN DID BUGS BUNNY STEAL THE CARROTS?"
05.20 T !"1)MR.MAGILLICUTY'S,2)ELMER FUDD'S,"
05.21 T " 3)CLEM JUDD'S, 4)STROMBOLI'S"
05.25 ASK A,%1
05.30 IF (A-2)5.4,5.5,5.4
05.40 T !"TOO BAD-IT WAS ELMER FUDD'S GARDEN!";RETURN
05.50 T !"PRETTY GOOD!!"

06.10 S A=4;T !"IN THE WIZZARD OF OZ, WHAT WAS THE NAME OF DOROTHY'S
06.15 T " DOG ?"
06.20 T !"1)CICERO, 2)TRIXIE, 3)KING, 4)TOTO";ASK A,%1
06.30 IF (A-4)6.4,6.5,6.4
06.40 T !"TOTO WAS HIS NAME.";RETURN
06.50 T !"YOUR ANSWER IS CORRECT."

07.10 S A=3; T !"WHO WAS THE FAIR MAIDEN WHO ATE THE POISON APPLE?"
07.20 T !"1)SLEEPING BEAUTY, 2)CINDERELLA, 3)SNOW WHITE, 4)WENDY"
07.25 ASK A,%1
07.30 IF (A-3) 7.4,7.5,7.4
07.40 T !"THAT WAS SNOW WHITE!";RETURN
07.50 T !"GOOD MEMORY!!"

08.10 S A=1; T !"IN 'PETER PAN', WHAT DID PETER ASK WENDY
08.15 T " TO SEW ON FOR HIM?"
08.20 T !"1)HIS SHADDOW, 2)HIS POCKET, "
08.25 T "3)A PATCH, 4)HIS SLEEVE";A A,%1
08.30 IF (A-1) 8.40,8.50,8.40
08.40 T !"WENDY SEWED PETER PAN'S SHADDOW BACK ON!";RETURN
08.50 T !"VERY GOOD."

09.10 S A=4; T !"IN WHAT STORY DID GUIPETTO GET"
09.15 T " SWALLOWED BY A WHALE?"
09.20 T !"1)MOBY DICK, 2)PETER PAN, 3)JONAH, 4)PINOCCHIO";ASK A,%1
09.30 IF (A-4)9.40,9.50,9.4
09.40 T !"THE ANSWER IS 'PINOCCHIO'";RETURN
09.50 T !"GOOD CHOICE."
*
```

```
*GO

TEST YOUR KNOWLEDGE IN CHILDREN'S LITERATURE.
THIS IS A MULTIPLE CHOICE QUIZ.
TYPE EITHER A 1, 2, 3, OR 4 AFTER THE COLON. IF
YOU FAIL, THE CORRECT ANSWER WILL BE TYPED.
GOOD LUCK!

IN 'PINOCCHIO', WHAT WAS THE NAME OF THE CAT?
1)TIGGER, 2)CICERO, 3)FIGARO, 4)GUIPETTO:4

SORRY- FIGARO WAS HIS NAME.
FROM WHOSE GARDEN DID BUGS BUNNY STEAL THE CARROTS?
1)MR.MAGILLICUTY'S,2)ELMER FUDD'S, 3)CLEM JUDD'S, 4)STROMBOLI'S:2

PRETTY GOOD!!
IN THE WIZZARD OF OZ, WHAT WAS THE NAME OF DOROTHY'S DOG ?
1)CICERO, 2)TRIXIE, 3)KING, 4)TOTO:4

YOUR ANSWER IS CORRECT.
WHO WAS THE FAIR MAIDEN WHO ATE THE POISON APPLE?
1)SLEEPING BEAUTY, 2)CINDERELLA, 3)SNOW WHITE, 4)WENDY:1

THAT WAS SNOW WHITE!
IN 'PETER PAN', WHAT DID PETER ASK WENDY TO SEW ON FOR HIM?
1)HIS SHADDOW, 2)HIS POCKET, 3)A PATCH, 4)HIS SLEEVE:1

VERY GOOD.
IN WHAT STORY DID GUIPETTO GET SWALLOWED BY A WHALE?
1)MOBY DICK, 2)PETER PAN, 3)JONAH, 4)PINOCCHIO:4

GOOD CHOICE.THE END!!*
```

Lunar Module

ABSTRACT

This is an exciting 21st century game that allows you to pilot your own spacecraft and land on the moon.

The example that follows was a successful lunar landing; therefore, some of the data has been removed so it cannot be duplicated without some effort.

Operational Procedures

1. Load *Lunar Module* by FOCAL-8.***

2. Type *GO* and the countdown begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.04 T "CONTROL CALLING LUNAR MODULE. MANUAL CONTROL IS NECESSARY"!
01.06 T "YOU MAY RESET FUEL RATE K EACH 10 SECS TO 0 OR ANY VALUE"!
01.08 T "BETWEEN 8 & 200 LBS/SEC. YOU'VE 16000 LBS FUEL. ESTIMATED"!
01.11 T "FREE FALL IMPACT TIME-120 SECS. CAPSULE WEIGHT-32500 LBS"!
01.20 T "FIRST RADAR CHECK COMING UP"!!!;E
01.30 T "COMMENCE LANDING PROCEDURE"!"TIME,SECS   ALTITUDE,"
01.40 T "MILES+FEET     VELOCITY,MPH    FUEL,LBS    FUEL RATE"!
01.50 S L=0;S A=120;S V=1;S M=33000;S N=16500;S G=.001;S Z=1.8

02.10 T "    ",%3,L,"         ",FITR(A),"  ",%4,5280*(A-FITR(A))
02.20 T %6.02,"          ",3600*V,"    ",%6.01,M-N,"       K=";A K;S T=10
02.70 T %7.02;I (K)2.72;I (200-K)2.72;I (K-8)2.71,3.1,3.1
02.71 I (K-0)2.72,3.1,2.72
02.72 T "NOT POSSIBLE";F X=1,51;T "."
02.73 T "K=";A K;G 2.7

03.10 I ((M-N)-.001)4.1;I (T-.001)2.1;S S=T
03.40 I ((N+S*K)-M)3.5,3.5;S S=(M-N)/K
03.50 D 9;I (I)7.1,7.1;I (V)3.8,3.8;I (J)8.1
03.80 D 6;G 3.1

04.10 T "FUEL OUT AT",L," SECS"!
04.40 S S=(-V+FSQT(V*V+2*A*G))/G;S V=V+G*S;S L=L+S

05.10 T "ON THE MOON AT",L," SECS"!;S W=3600*V
05.20 T "IMPACT VELOCITY OF",W," M.P.H."!,"FUEL LEFT:"
05.30 T M-N," LBS."!;I (-W+1)5.5,5.5
05.40 T "PERFECT LANDING !-(LUCKY)"!;G 5.9
05.50 I (-W+10)5.6,5.6;T "GOOD LANDING-(COULD BE BETTER)"!;G 5.9
05.60 I (-W+25)5.7,5.7;T "CONGRATULATIONS ON A POOR LANDING"!;G 5.9
05.70 I (-W+60)5.8,5.8;T "CRAFT DAMAGE. GOOD LUCK"!;G 5.9
05.80 T "SORRY,BUT THERE WERE NO SURVIVORS-YOU BLEW IT!"!"IN "
05.81 T "FACT YOU BLASTED A NEW LUNAR CRATER",W*.277777," FT. DEEP"!
05.90 T !!!"TRY AGAIN?"!
05.92 A "(ANS. YES OR NO)"P;I (P-0NO)5.94,5.98
05.94 I (P-0YES)5.92,1.2,5.92
05.98 T "CONTROL OUT";Q

06.10 S L=L+S;S T=T-S;S M=M-S*K;S A=I;S V=J

07.10 I (S-.005)5.1;S S=2*A/(V+FSQT(V*V+2*A*(G-Z*K/M)))
07.30 D 9;D 6;G 7.1

08.10 S W=(1-M*G/(Z*K))/2;S S=M*V/(Z*K*(W+FSQT(W*W+V/Z)))+.05;D 9
08.30 I (I)7.1,7.1;D 6;I (-J)3.1,3.1;I (V)3.1,3.1,8.1

09.10 S Q=S*K/M;S J=V+G*S+Z*(-Q-Q↑2/2-Q↑3/3-Q↑4/4-Q↑5/5)
09.40 S I=A-G*S*S/2-V*S+Z*S*(Q/2+Q↑2/6+Q↑3/12+Q↑4/20+Q↑5/30)
*
```

```
*GO
CONTROL CALLING LUNAR MODULE. MANUAL CONTROL IS NECESSARY
YOU MAY RESET FUEL RATE K EACH 10 SECS TO 0 OR ANY VALUE
BETWEEN 8 & 200 LBS/SEC. YOU'VE 16000 LBS FUEL. ESTIMATED
FREE FALL IMPACT TIME-120 SECS. CAPSULE WEIGHT-32500 LBS
FIRST RADAR CHECK COMING UP


COMMENCE LANDING PROCEDURE
TIME,SECS   ALTITUDE,MILES+FEET    VELOCITY,MPH    FUEL,LBS    FUEL RATE
    =   0        = 120  =     0     = 3600.00      = 16500.0        K=:0
    =  10        = 109  =  5016     = 3636.00      = 16500.0        K=:0
    =  20        =  99  =  4224     = 3672.00      = 16500.0        K=:0
    =  30        =  89  =  2904     = 3708.00      = 16500.0        K=:0
    =  40        =  79  =  1056     = 3744.00      = 16500.0        K=:0
    =  50        =  68  =  3960     = 3780.00      = 16500.0        K=:200
    =  60        =  58  =  3996     = 3410.87      = 14500.0        K=:200
    =  70        =  49  =  4360     = 3014.[illegible]  = 12500.0        K=:200
    =  80        =  42  =   195     = 25[illegible]      = 10500.0        K=:0
    =  90        =  34  =  4219     = [illegible]        10500.0        K=:0
    = 10[illegible]         27  =  2435     [illegible]            [illegible]500.0        K[illegible]
    [illegible]                     122     [illegible]            [illegible]0
      [illegible]        =   [illegible]        [illegible]95
     130         =   [illegible]        [illegible]6.63                          [illegible]200
    = 140        =   3  = [illegible]      [illegible]175.14      = [illegible]              K=:200
    = 150        =   1  =   96[illegible]     [illegible]562.60      =  250[illegible].0        K=:150
    = 160        =   0  =  1582     =   65.69      =  1000.0        K=:15
    = 170        =   0  =   763     =   45.91      =   850.0        K=:15
    = 180        =   0  =   238     =   25.64      =   700.0        K=:15
    = 190        =   0  =    13     =    4.89      =   550.0        K=:0
ON THE MOON AT=   191.27 SECS
IMPACT VELOCITY OF=     9.44 M.P.H.
FUEL LEFT:=   550.00 LBS.
GOOD LANDING-(COULD BE BETTER)


TRY AGAIN?
(ANS. YES OR NO):
```

Management Game

ABSTRACT

The *Management Game* tests your skills in handling a high level production budget. It is also a competitive game, in which two teams challenge one another on a quarterly basis for profit and loss in the hopes that one will go bankrupt.

Operational Procedures

1. Load procedure according to FOCAL instructions for loading from high/low speed reader.***

2. Issue the *GO* command; the game immediately begins.

3. Team 1 should input after the colon (:) their estimated production level, advertising budget, price per unit; Team 2 should also input their estimated budget.

4. FOCAL will interpret these estimates and state a quarterly return for each team.

5. An example (listing of the program followed by a sample run) to illustrate this game follows.

```
*W
C-FOCAL,1969

01.01 T !"THE MANAGEMENT GAME
01.10 SET C(1)=250000;SET C(2)=C(1);SET H(1)=1000;SET H(2)=H(1)
01.20 SET A(1)=125000;SET A(2)=A(1);SET TO=250000
01.30 T !!"IF THE JUNIOR EXECUTIVES ARE READY, I AM"!!;FOR I=1,2;D 4
01.40 SET IQ=1
01.50 FOR I=1,2;DO 5
01.55 SET TA=AC(1)+AC(2); SET D=A(1)*P(1) + A(2)*P(2)
01.60 SET QT=3.2*FSQT(IQ)*FRAN()
01.64 S TO=TO*(1.07+QT*TA/D*100);F I=1,2;D 1.99;S SU=SH(1)+SH(2)
01.65 S I=1;DO 3;S I=2;DO 3;IF (PD(1)-S(1))1.68,1.68;
01.66 IF (PD(2)-S(2))1.71,1.71;SET A(1)=S(1);SET A(2)=S(2);GOTO 1.85
01.68 IF (-PD(2)+S(2))1.75;:
01.69 SET A(1)=PD(1);SET A(2)=PD(2);SET TO=A(1)+A(2);GOTO 1.85
01.71 SET A(2)=PD(2);SET RS=TO-A(2);IF (PD(1)-RS)1.73,1.74
01.73 SET A(1)=RS;GOTO 1.85
01.74 SET A(1)=PD(1);GOTO 1.85
01.75 SET A(1)=PD(1);SET RS=TO-A(1);IF (-PD(2)+RS)1.77;
01.76 SET A(2)=PD(2);GOTO 1.85
01.77 SET A(2)=RS
01.85 FOR I=1,2; DO 1.97; DO 4
01.87 SET IQ=IQ+1;IF (IQ-4)1.5,1.5;
01.92 T !!"DO YOU WISH TO DO ANOTHER YEAR?";ASK YS
01.93 IF (YN-0NO)1.4,1.95,1.4
01.95 T !!"GOODBYE JUNIOR EXECUTIVES!"
01.97 SET H(I)=PD(I)-A(I);SET C(I)=C(I)+A(I)*P(I)-AC(I)-CI(I)
01.99 SET SH(I)=(AC(I)/TA)*(P(1)+P(2))/(2*P(I))

03.01 SET SH(I)=SH(I)/SU;SET S(I)=TO*SH(I);SET PD(I)=U(I)+H(I)
03.02 SET CI(I)=H(I)/50

04.01 T !!"QUARTERLY REPORT FOR TEAM",%1,I," FOR QUARTER ",IQ
04.02 T !!"MARKET SHARE   CASH ON HAND   NUMBER SOLD   INVENTORY"
04.04 T !!%8.02,100*SH(I),"% $",C(I),"  ",%9,A(I),"  ",%7,H(I)
04.05 IF (C(I))4.06,4.06,4.08
04.06 T !!"TEAM",%1," IS: *********BANKRUPT*********"
04.08 T !!

05.01 T !!"TEAM",%1,I," INPUT PRODUCTION LEVEL";ASK U(I)
05.02 T !!"TEAM",I," INPUT ADVERTISING BUDGET";ASK AC(I)
05.03 T !!"TEAM",I," INPUT PRICE PER UNIT";ASK P(I)
*
```

```
*GO

THE MANAGEMENT GAME

IF THE JUNIOR EXECUTIVES ARE READY, I AM

QUARTERLY REPORT FOR TEAM= 1 FOR QUARTER = 0

MARKET SHARE   CASH ON HAND   NUMBER SOLD   INVENTORY

=       0.00% $= 250000.00   =     125000   =     1000

QUARTERLY REPORT FOR TEAM= 2 FOR QUARTER = 0

MARKET SHARE   CASH ON HAND   NUMBER SOLD   INVENTORY

=       0.00% $= 250000.00   =     125000   =     1000


TEAM= 1 INPUT PRODUCTION LEVEL:1000

TEAM= 1 INPUT ADVERTISING BUDGET:5000

TEAM= 1 INPUT PRICE PER UNIT:100

TEAM= 2 INPUT PRODUCTION LEVEL:1500

TEAM= 2 INPUT ADVERTISING BUDGET:7500

TEAM= 2 INPUT PRICE PER UNIT:90

QUARTERLY REPORT FOR TEAM= 1 FOR QUARTER = 1

MARKET SHARE   CASH ON HAND   NUMBER SOLD   INVENTORY

=      37.50% $= 444980.00   =       2000   =        0

QUARTERLY REPORT FOR TEAM= 2 FOR QUARTER = 1

MARKET SHARE   CASH ON HAND   NUMBER SOLD   INVENTORY

=      62.50% $= 467480.00   =       2500   =        0


TEAM= 1 INPUT PRODUCTION LEVEL:
```

Perpetual Calendar

ABSTRACT

Given the month, day, and year, the *Perpetual Calendar* will type the day of the week.

Operational Procedures

1. The *Perpetual Calendar* is loaded by FOCAL-8.*

2. Type *GO*, respond to the dialogue, and your answer is immediately typed back.

3. A sample run follows.

```
*W
C-FOCAL,1969

01.10 ASK !"WHAT IS THE DATE ? (MM/DD/YYYY) "M,K,C,!
01.20 S C=C/100;S D=FITR(.1+100*(C-FITR(C)));S C=FITR(C)
01.30 S M=M-2; IF (M) 5.4, 5.4; GOTO 5.5

05.40 S M=M+12;S D=D-1;I (-D)5.5,5.5;S D=99;S C=C-1
05.50 S X=FITR<FITR[2.6*M-.2]+K+D+FITR[D/4]+FITR(C/4)-2*C>
05.60 IF (X-6) 5.7,5.7;S X=X-7;G 5.6
05.70 T !"THE DAY IS "; DO 6.1
05.80 IF (M*1E6+K*1E4+C - Q )5.9,5.85,5.9
05.85 T " , TODAY !"
05.90 T !!; GOTO 1.1

06.10 I (X)6.26,6.2;I (X-2)6.21,6.22,6.15
06.15 I (X-4)6.23,6.24;I (X-6)6.25,6.26;
06.20 T "SUNDAY
06.21 T "MONDAY
06.22 T "TUESDAY
06.23 T "WEDNESDAY
06.24 T "THURSDAY
06.25 T "FRIDAY
06.26 T "SATURDAY
06.50 ASK M,K,C;DO 1.2;D 1.3; SET Q=M*1E6+K*1E4+C;GOTO 1.1
*
*GO

WHAT IS THE DATE ? (MM/DD/YYYY) :10
:12
:1492

THE DAY IS WEDNESDAY

WHAT IS THE DATE ? (MM/DD/YYYY) :
```

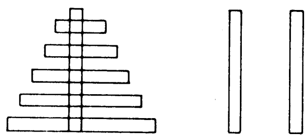
Towers of Hanoi

ABSTRACT

The Towers of Hanoi **is a challenging game in FOCAL and an example of recursive programming.*****

Origin of Game

According to legend, a secret society of monks lives beneath the city of Hanoi. They possess three large stacks of towers on which different size disks may be placed.



Moving one at a time and never placing a larger on a smaller disk, the monks endeavor to move the tower of disks from the left stack to the right stack. Legend says that when they have finished moving this 64-disk tower, the world will end!

What is the minimum number of moves they will have to make?

Using this program you can try your hand at a small stack or watch the computer solve it.

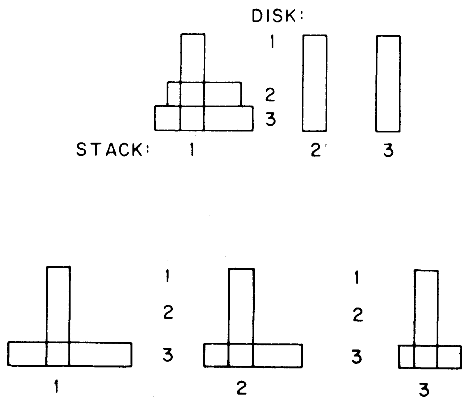
Method of Operation

The program is written in FOCAL and will run on a 4K PDP-8. To start the program, type *GO,* followed by a carriage return. Type a space following any response made to a question asked by the program. To terminate the program at any time, type a *CTRL/C*. The program must be run without extended functions.

The program first asks for the number of disks (3 to 8). It then asks for the output desired: either a plot of the disk positions or a list of the moves. The final question determines whether you will make the moves manually, or if the program will proceed automatically (0 to 1).

A move is selected by determining the stack out (SO) and the vertical disk number out (NO), and the stack in (SI) and the disk number in NI). Error checking is not performed.

Thus, the next move is this example : SO:1 NO:2 SI:2 NI:3



08-0509

Algorithm

The stacks are internally represented as an array (SS). The value of a member of this array represents the size of the disk in that position.

DISK			
1	0	0	0
2	0	0	0
3	3	2	1
STACK	1	2	3

Explanation of *Tower of Hanoi* Program

Group 1 Main Program
Ask for number of disks to be moved.
Initialize the stacks.
Move the stack (DO 2).

Group 2 Move a specified stack.
Save move request.
To move this stack first move all but one disk (i.e., NO-NO-1).
If the resultant stack has an odd number of disks, move the stack to temporary storage (i.e., SI=6-SO-SI).

If no disk remains to be moved in the output stack, return (line 2.3).
Find the first free position on the input stack, (line 2.5).
Move the remaining stack and the bottom piece (line 2.6). Then move that stack back onto the bottom piece (lines 2.7, 2.8).
Restore move request and return (line 2.9).

Group 5	User must specify which stack for output (SO) and which disk to move (NO). No check is made if the user cheats. (This opens the possibility of inventing new games, by the way.) A check is made to see if the user has finished (lines 5.4 and 5.5).
Group 6	Execute a single move. SO and NO and SI and NI are used to transfer a piece.
Group 23	Plot the status of the board. Each piece has a size value kept the stack array SS (I). The stacks are scanned (line 23.1) and positions on each stack are checked.

```
*W
C-FOCAL,1969

01.05 T !" TOWERS OF HANOI."!;E
01.10 A " NO. OF DISKS? "N,!
01.20 F I=1,N;S SS(I)=I
01.30 S SO=1;S SI=3
01.40 S NO=N;S NI=N;S I=0
01.45 A "MOVES#0, PLOTS#1 ? "MOVE,!
01.46 IF (MOVE)ERR,1.47; DO 23
01.47 ASK "AUTO#0, MANUAL#1 ? ",A,!
01.50 I (-A)5.1;D 2;T !!"DONE !"!!;Q

02.20 I [SS<(SO-1)*N+NO-1>]ER,2.95;
02.30 S  I=I+1;S NO(I)=NO;S SO(I)=SO;S SI(I)=SI
02.50 S SI=6-SO-SI;S NO=NO-1;D 3;S TE(I)=NI;D 2
02.60 S SI=SI(I);S NO=NO+1;D 3;D 6
02.70 S SO=6-SO-SI;S NO=TE(I); DO 3; DO 2
02.80 S SI=SI(I);S SO=SO(I);S NO=NO(I);S I=I-1
02.90 R
02.95 D 3;D 6;R

03.10 S NI=N
03.20 I [SS((SI-1)*N+NI)]ER,3.3;S NI=NI-1;G 3.2
03.30 R

05.10 A ? SO  NO ?!? SI  NI  ?!;D 6
05.30 S A=0
05.40 F I=1,N*2;S A=A+SS(I)
05.50 I (-A) 5.1;T !"WELL DONE!"!;Q

06.10 S DO=(SO-1)*N+NO
06.20 S DI=(SI-1)*N+NI
06.30 S SS(DI)=SS(DO)
06.40 S SS(DO)=0
06.50 I (MOVE)E,6.7;DO 23;R
06.70 T !%2,?SO,   NO,!SI,   NI,?!

23.10 F J=1,N;T !;F K=0,70;DO 23.3
23.20 T !!!!;R
23.30 IF [K-15+SS(J)*2]23.6;IF [-K+15+SS(J)*2]23.6;T "#
23.60 IF [K-35+SS(J+N)*2]23.7;IF [-K+35+SS(J+N)*2]23.7;T "#
23.77 S K=100;R
23.80 T " "
*
```

```
*GO

 TOWERS OF HANOI.
 NO. OF DISKS? :3

MOVES#0, PLOTS#1 ? :1


             #####                 #                   #
           #########               #                   #
         #############             #                   #


AUTO#0, MANUAL#1 ? :0


               #                   #                   #
           #########               #                   #
         #############             #                 #####


               #                   #                   #
               #                   #                   #
         #############         #########             #####


               #                   #                   #
               #                 #####                 #
         #############         #########               #


               #                   #                   #
               #                 #####                 #
               #               #########         #############


               #                   #                   #
               #                   #                   #
             #####             #########         #############


               #                   #                   #
               #                   #               #########
             #####                 #             #############


               #                   #                 #####
               #                   #               #########
               #                   #             #############

  DONE !
```

```
*GO

 TOWERS OF HANOI.
 NO. OF DISKS? :4

MOVES#Ø, PLOTS#1 ? :Ø

AUTO#Ø, MANUAL#1 ? :Ø


SO,=  1   NO,=  1!
SI,=  2   NI,=  4

SO,=  1   NO,=  2!
SI,=  3   NI,=  4

SO,=  2   NO,=  4!
SI,=  3   NI,=  3

SO,=  1   NO,=  3!
SI,=  2   NI,=  4

SO,=  3   NO,=  3!
SI,=  1   NI,=  3

SO,=  3   NO,=  4!
SI,=  2   NI,=  3

SO,=  1   NO,=  3!
SI,=  2   NI,=  2

SO,=  1   NO,=  4!
SI,=  3   NI,=  4

SO,=  2   NO,=  2!
SI,=  3   NI,=  3

SO,=  2   NO,=  3!
SI,=  1   NI,=  4

SO,=  3   NO,=  3!
SI,=  1   NI,=  3

SO,=  2   NO,=  4!
SI,=  3   NI,=  3

SO,=  1   NO,=  3!
SI,=  2   NI,=  4

SO,=  1   NO,=  4!
SI,=  3   NI,=  2

SO,=  2   NO,=  4!
SI,=  3   NI,=  1


DONE !

*
```

Word Game

ABSTRACT

Word Game is easy and fun to play. The player must choose a number; then he must guess the word stored in that number. For example: the computer asks "Which word no.?: 1." (User responded with a one.)

↑↑↑↑↑- - - - - - - - - - (this indicates the number of letters in the word)

GUESS A LETTER: E, ↑↑E↑↑
GUESS A LETTER: T, ↑↑E↑↑ ("T" is not a letter of this word)

The game continues until all the letters have been identified.

Operational Procedures

1. Load *Word Game* via FOCAL-8.

2. Type *GO*, respond to the dialogue, and the game begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.01 T !" 'WORD GAME' USES A SPECIAL 'FADC' FUNCTION."!
01.05 T %5,!"THERE ARE "1," TO "Q-1," WORDS."!
01.10 A !" WHICH WORD NO. ? " Z,!
01.20 IF (Q-1-Z)1.1;
01.30 SET JS=0; SET KF=NL(Z)-1
01.40 FOR J=0,Z-1; SET JS=JS+NL(J)
01.50 SET JF=JS+KF
01.60 FOR J=0,KF; SET N(J)=FADC(30)

02.10 ASK !" GUESS A LETTER"L,"      "
02.15 IF (L)2.5,2.16,2.2
02.16 SET L=5
02.20 S K=0;F J=JS,JF;D 5;S Z=FADC(N(K-1))
02.25 SET Z=0
02.30 FOR J=0,KF; DO 6
02.40 IF (Z),1.1,2.1
02.50 S K=0;F J=JS,JF;D 5;S Z=FADC(X)
02.60 GOTO 1.1

05.02 S X=FITR(J/4)
05.04 S Z=J-4*X
05.06 S X=M(X)/40↑(4-Z)
05.07 S X=FITR<.4+40*(X-FITR(X))>
05.10 IF (X-L) 5.3,5.2,5.3
05.20 SET N(K)=L
05.30 S K=K+1

06.10 IF (30-N(J)),6.2;R
06.20 SET Z=Z+1

07.01 E
07.02 S X=0; S K=0
07.05 S Q=Q+1; S NL(Q)=0; T !!
07.10 ASK L; IF (L)7.05,7.15,7.2
07.15 S L=5
07.20 S M(X)=M(X)*40+L; S K=K+1
07.25 S NL(Q)=NL(Q)+1
07.30 IF (K-4*FITR(K/4))7.1,7.4,7.1
07.40 S X=X+1; S K=0; GOTO 7.1

08.10 FOR Z=1,Q-1; SET JS=JS+NL(Z)
08.15 ;S K=0
08.20 FOR J=0,JS;DO 5; S Z=FADC(X)
*
```

```
*GO

'WORD GAME' USES A SPECIAL 'FADC' FUNCTION.

THERE ARE =     1 TO =      9 WORDS.

 WHICH WORD NO. ? :1
↑↑↑↑↑
 GUESS A LETTER:E ,    ↑↑E↑↑
 GUESS A LETTER:T/    ↑↑E↑↑
 GUESS A LETTER:N/    ↑↑E↑↑
 GUESS A LETTER:O/    ↑↑E↑↑
 GUESS A LETTER:I/    ↑↑E↑↑
 GUESS A LETTER:R/    ↑↑E↑↑
 GUESS A LETTER:S/    ↑↑ESS
 GUESS A LETTER:B/    ↑↑ESS
 GUESS A LETTER:A/    ↑↑ESS
 GUESS A LETTER:G/    G↑ESS
 GUESS A LETTER:U/    GUESS
 WHICH WORD NO. ? :2/
↑↑↑↑↑↑↑↑
```

III. MATHEMATICAL ROUTINES

A. Education

Addition Exerciser

ABSTRACT

This educational routine is designed for elementary school children. The purpose and result of this routine complement each other. First, the purpose of this routine is to quiz the child in basic addition; therefore he may learn to associate numbers and quantities at a more rapid pace. The result of this is that the student is introduced to the computer at an early age. He will eventually conclude that he can not only learn and have fun with the computer, but he may also conclude that it is a very applicable tool. And there is always that chance that later in life he may remember his past experience.

Operational Procedures

1. Load *Addition Exerciser* via FOCAL-8.*

2. Type *GO* and execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.05 TYPE "HELLO, PLEASE ADD THE FOLLOWING SETS OF NUMBERS."!
01.10 SET A=FABS( FITR(100*FRAN())); SET BFABS(FITR(99*FRAN()))
01.20 TYPE %7, A,!B,!"-------"!!
01.30 ASK REPLY,!
01.40 IF (REPLY-A-B) 2.1,1.5,2.1
01.50 SET WR=0YPE "THAT IS CORRECT."!
01.60 GOTO 1.
01.70 TYPE ?  (625-1)↑2,       (376-1)↑2,     ?
01.80 T !?  (25-1)↑2,     (76-1)↑2,     ?
01.90 T !!

02.10 SET WR=WR+1; IF (W2) 2.2,2.2,3.1
02.20 T "SORRY, TRY AGAIN,"!; GOTO 1.2
02.40 TYYPEERASE ALL

03.10 T "IF YOU ARE HAVING TROUBLE, ASK YOUR TEACHER FOR HELP."!
03.20 TYPE "THE CORRECT ANSWER IS "A+B,!
03.30 GOTO 1.1
*
```

```
**GO
*GO
HELLO, PLEASE ADD THE FOLLOWING SETS OF NUMBERS.
=       60
=       73
-------

:133

THAT IS CORRECT.
=       89
=       53
-------

:142

THAT IS CORRECT.
=       66
=       79
-------

:145

THAT IS CORRECT.
=       97
=       58
-------

:0

SORRY, TRY AGAIN,
=       97
=       58
-------

:154

SORRY, TRY AGAIN,
=       97
=       58
-------

:156

IF YOU ARE HAVING TROUBLE, ASK YOUR TEACHER FOR HELP.
THE CORRECT ANSWER IS =      155
=       71
=       86
-------
```

Numerical Relationships

ABSTRACT

This conversational routine causes the student to think about various numerical relationships. This routine should arouse the curiosity of the student enough to discover and to draw his own conclusions of numbers and how they relate to each other.

Operational Procedures

1. Load *Numerical Relationships* via FOCAL-8.

2. Type *GO* and the program begins.

3. A listing and sample run follow.*

```
*WRITE ALL
C-FOCAL,1969

01.05 T !!"HI,  THINK ABOUT THE FOLLOWING RELATIONSHIPS :"!!
01.10 TYPE !!%6,? FSQT(169),   FSQT(961),    FSQT(169*961),     ?!!
01.20 TYPE "  13 * 31 = " 13*31,!
01.30 T !!!!"DID YOU KNOW THAT A AUTOMORPHIC NUMBER
01.40 T !"IS ONE WHICH REAPPEARS AT THE END OF ITS SQUARE?
01.50 T !"HERE ARE A FEW: "!!? 5↑2,   6↑2,    25↑2,    76↑2, ?!
01.60 T !?  625↑2,          376↑2,    ?!!
01.70 TYPE ?  (625-1)↑2,        (376-1)↑2,     ?
01.80 T !?  (25-1)↑2,     (76-1)↑2,     ?
01.90 T !!

02.10 T !!!!"TWO FACTORS WHOSE PRODUCT IS  1 ARE  CALLED RECIPROCALS.
02.20 TYPE !,"3/4 IS THE RECIPROCAL OF 4/3
02.30 T !,"4/3 IS THE RECIPROCAL OF 3/4"!
02.40 TYPE !,"WE USE RECIPROCALS WHEN WE DIVIDE BY RATIONAL NUMBERS
02.41 TYPE !,?   3/(1/4)?,!?   3*4    ?,!!
02.42 T ,?   8/(2/5)?,!?   8*(5/2)?,!?   10/(5/6)?,!?   10*(6/5)?,!!
02.43 TYPE ?    (7/10)/(1/100)?,!?   (7/10)*(100/1)?,!!
02.44 TYPE ?    2.5/(2/5)?,!?   2.5*(5/2)?,!!
*
```

```
*GO


HI,  THINK ABOUT THE FOLLOWING RELATIONSHIPS :


  FSQT(169),=     13   FSQT(961),=     31    FSQT(169*961),=     403

  13 * 31 = =    403



DID YOU KNOW THAT A AUTOMORPHIC NUMBER
IS ONE WHICH REAPPEARS AT THE END OF ITS SQUARE?
HERE ARE A FEW:

 5↑2,=      25   6↑2,=      36    25↑2,=     625     76↑2,=    5776

  625↑2,= 390625         376↑2,= 141376

  (625-1)↑2,= 389376       (376-1)↑2,= 140625
  (25-1)↑2,=     576    (76-1)↑2,=    5625



TWO FACTORS WHOSE PRODUCT IS  1 ARE  CALLED RECIPROCALS.
3/4 IS THE RECIPROCAL OF 4/3
4/3 IS THE RECIPROCAL OF 3/4

WE USE RECIPROCALS WHEN WE DIVIDE BY RATIONAL NUMBERS
   3/(1/4)=      12
   3*4    =      12

   8/(2/5)=      20
   8*(5/2)=      20
   10/(5/6)=      12
   10*(6/5)=      12

   (7/10)/(1/100)=       70
   (7/10)*(100/1)=       70

   2.5/(2/5)=       6
   2.5*(5/2)=       6

*
```

Prime Factors of Positive Integers

ABSTRACT

By inputting a positive integer, this FOCAL-8 demonstration routine will dump on the Teletype all of the prime factors of the specified integer.

Operational Procedures

1. Load *Prime Factors of Positive Integers* by FOCAL-8.*

2. Type *GO* and respond to the request for a positive integer and the prime factors will be typed.

3. A sample run follows.

```
*WRITE
C-FOCAL,1969

01.10 ASK !!"A POSITIVE INTEGER>1 PLEASE" N ,!!;SET DI=2;SET PH=0
01.11 IF (FITR(N)-N) 1.1;IF (N-1) 1.1;SET  P=N
01.20 IF (P/DI-FITR(P/DI)) 1.4,1.3,1.4
01.30 TYPE "PRIME FACTOR" DI,!;SET P=P/DI;GOTO 1.2
01.40 IF (1-PH) 1.1,1.5;SET PH=1;SET DI=DI+1;GOTO 1.2
01.50 SET DI=DI+2;IF (DI-P) 1.6,1.6; TYPE !"DONE"!;GOTO 1.1
01.60 IF (DI-FSQT(FABS(N))) 1.2,1.2;SET DI=P;GOTO 1.2
*
```

```
*GO

A POSITIVE INTEGER>1 PLEASE:100

PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  5.00
PRIME FACTOR=  5.00

DONE

A POSITIVE INTEGER>1 PLEASE:10

PRIME FACTOR=  2.00
PRIME FACTOR=  5.00

DONE

A POSITIVE INTEGER>1 PLEASE:50

PRIME FACTOR=  2.00
PRIME FACTOR=  5.00
PRIME FACTOR=  5.00

DONE

A POSITIVE INTEGER>1 PLEASE:33

PRIME FACTOR=  3.00
PRIME FACTOR= 11.00

DONE

A POSITIVE INTEGER>1 PLEASE:4096

PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00
PRIME FACTOR=  2.00

DONE
```

Prime Number Generator

ABSTRACT

Input any given number and the *Prime Number Generator* will type all the prime numbers of the specified number.

Operational Procedures

1. Load *Prime Number Generator* by FOCAL-8*

2. Type *GO* and input a number and the prime numbers will be typed.

3. A sample run follows.

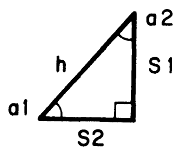
```
*WRITE ALL
C-FOCAL,1969

01.10 A ?N?,!;S PR=1;S TS=3;T "PRIME",2
01.20 S PR=PR+2; IF (PR-N)1.4,1.3,1.3
01.30 T !, "DONE  "; GOTO 1.1
01.40 IF (TS-FSQT(PR))1.6,1.6,1.5
01.50 T !,"PRIME",PR;S TS=3;GOTO 1.2
01.60 IF (PR/TS-FITR(PR/TS))1.8,1.7,1.8
01.70 S TS=3;GOTO 1.2
01.80 S TS=TS+2;GOTO 1.4
*
```

```
*GO
N:6

PRIME=  2.00
PRIME=  3.00
PRIME=  5.00
DONE  N:10

PRIME=  2.00
PRIME=  3.00
PRIME=  5.00
PRIME=  7.00
PRIME=  9.00
DONE  N:100

PRIME=  2.00
PRIME=  3.00
PRIME=  5.00
PRIME=  7.00
PRIME=  9.00
PRIME= 11.00
PRIME= 13.00
PRIME= 17.00
PRIME= 19.00
PRIME= 23.00
PRIME= 29.00
PRIME= 31.00
PRIME= 37.00
PRIME= 41.00
PRIME= 43.00
PRIME= 47.00
PRIME= 53.00
PRIME= 59.00
PRIME= 61.00
PRIME= 67.00
PRIME= 71.00
PRIME= 73.00
PRIME= 79.00
PRIME= 83.00
PRIME= 89.00
PRIME= 97.00
DONE  N:
```

Right Triangle

ABSTRACT

Given the length of the first side and the degrees of the adjacent angle, this routine computes the hypotenuse, the length of the second side, and the number of degrees for the other angle.



Operational Procedures

1. Load *Right Triangle* by FOCAL-8.

2. Type *GO*, supply the length of S1 and the degrees of the adjacent angle.

3. The results will be dumped on the Teletype.**

4. A sample run follows.

```
C-FOCAL., 1968

01.10 ASK  "SIDE S1 EQUALS" S1
01.20 A " ADJACENT ANGLE A2 EQUALS" A2; TYPE "DEGREES"!!
01.30 S RATIO=3.141592/180; SET A1=90-A2
01.40 SET  HYP=S1/FSIN(A1*RATIO); SET S2=FSQT(HYP↑2-S1↑2)
01.50 T "SIDE S2 ", S2,!, "HYPOTENUSE", HYP,!
01.60 T "ANGLE A1", A1, !
*


*GO
SIDE S1 EQUALS:5
 ADJACENT ANGLE A2 EQUALS:30
DEGREES

SIDE S2 =  2.89
HYPOTENUSE=  5.77
ANGLE A1= 60.00
*GO
SIDE S1 EQUALS:10
 ADJACENT ANGLE A2 EQUALS:44
DEGREES

SIDE S2 =  9.66
HYPOTENUSE= 13.90
ANGLE A1= 46.00
*
```

B. Mathematics

Automatic Curve Fitting

ABSTRACT

By stating five end points, the *Automatic Curve Fitting* routine will compute and print the coefficients of the fourth order equation that fits these end points. Output may be one or all of the following:

a. **Data**
b. **Plot of the graph on the TTY**
c. **None (no output).**

Operational Procedures

1. Load this demonstration program by FOCAL-8.*

2. Type *GO* and input the five end points. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.05 DO 5
01.10 ASK !!!"  INPUT  "; FOR L=0,4; ASK A(L)
01.20 S AB=(A+A(4))/2; S SB=(A(4)-A)/2
01.30 S AS=(A(1)+A(3))/2; S SS=A(3)-A(1)
01.50 S A2=(AB-AS)*4/3; SET A3=(SB-SS)*4/3
01.60 S A0=AB-A2; S A1=SB-A3
01.70 S TA=A0-A1/2 + A3/2; S TE=A0+A1/2 - A3/2; S AM=A0-A(2)
01.80 TYPE !?A0?,?  A1?,?  A2?,?  A3?,!!?  TA?,?  TE?,!!!
01.90 ASK "OUTPUT ? "X,!! ; IF (X-0NONE)1.91,1.1
01.91 IT (X-0PLOT)2.1,2.3,2.1

02.10 T "   X       COMPUTED  APPROX"!;F K=1,2,7;D 3
02.20 GOTO 1.9
02.30 FOR K=1,41;DO 4
02.40 GOTO 1.9

03.10 S X=K/4-1
03.20 IF (X) 3.3,3.5,3.4
03.30 S X=X-2E-6
03.40 S X=X+1E-6
03.50 S Y1=A0 + A1*X + A2*X↑2 + A3*X↑3
03.60 S Y2=Y1-AM*(1-X↑2)*(1-4*X↑2)
03.70 TYPE %6.03,X,Y1,Y2,!

04.10 SET X=(K-1)/20-1 ; DO 3.5
04.20 FOR L=0,9; T " "
04.30 TYPE "." #; FOR L=0,9+20*Y1; TYPE " "
04.35 TYPE "*" #
04.40 IF (K-1-10*FITR<(K-1)/10>),4.45;T !;R
04.45 FOR L=0,9+20*A(K/10); TYPE " "
04.50 T "X" !; R

05.10 T !"GIVEN AN 'INPUT' OF FIVE POINTS, THIS ROUTINE WILL
05.20 T !"COMPUTE AND PRINT OUT THE COEFFICIENTS OF A 4TH ORDER
05.30 T !"EQUATION THAT FITS THE END POINTS. THE USER
05.40 T !"MUST JUGE HOW GOOD IS THE RESULTANT FIT AT THE
05.50 T !"MIDDLE POINT.  OUTPUT MAY B'NONE', 'DATA', OR 'PLOT'.
*
```

```
*GO

GIVEN AN 'INPUT' OF FIVE POINTS, THIS ROUTINE WILL
COMPUTE AND PRINT OUT THE COEFFICIENTS OF A 4TH ORDER
EQUATION THAT FITS THE END POINTS. THE USER
MUST JUGE HOW GOOD IS THE RESULTANT FIT AT THE
MIDDLE POINT.  OUTPUT MAY B'NONE', 'DATA', OR 'PLOT'.

  INPUT  :.15,:.7,:.5 :.8 :.0

A0=      0.98  A1=      0.16  A2=-     0.90  A3=-     0.23

  TA=      0.78  TE=      1.17


OUTPUT ? :PLOT


          . *X
          .   *
          .    *
          .     *
          .      *
          .       *
          .         *
          .          *
          .           *
          .            *
          .            *X
          .             *
          .              *
          .               *
          .                *
          .                *
          .                 *
          .                 *
          .                  *
          .                  *
          .         X        *
          .                  *
          .                  *
          .                  *
          .                  *
          .                  *
          .                 *
          .                 *
          .                *
          .               *
          .              *X
          .              *
          .            *
          .           *
          .          *
          .        *
          .       *
          .     *
          .   *
          . *
         *X
OUTPUT ? :NONE
```

Base to Base Integer Conversion

ABSTRACT

The *New Math* introduced the idea of representing numbers by their place value other than multiples of 10. For example: 123, base 10, is $1 \times 10^2 + 2 \times 10^1 + 3$. A number in base 8 (octal) is abc or $a * 8^2 + b * 8^1 + c$. A binary number is a number whose base is 2. Only numbers less than the base itself may be used as digits; thus, in base 2 only 0s and 1s may be used, and in octal representation only 0, 1, 2,. . . 7 are used. For those numbers larger than 10, one must assign special symbols to digits greater than 9, such as JQK (Jack, Queen, King) for numbers in base 13.

By using place value representation in base 10, any numerical equivalences can be checked.

For example:

octal: $3*8 + 2*8^1 + 0$

equals

binary: $1*2^8 + 1*2^7 + 0 + 1*2^5 + 0 + 0 + 0 + 0$

or it may be represented by the following expression:

$320)_8 = 11010000)_2$

Operational Procedures

1. Load *Base to Base Conversion* with FOCAL-8.*

2. Type *GO* and execution begins.

3. The user must give:

a. The number to be converted

b. The base number of the representation

c. The base number of the desired representation.

4. A sample run follows.

```
@@@@@@@@
C-FOCAL,1969

01.04 T !"BASE TO BASE INTEGER CONVERSION ROUTINE."!!
01.05 ASK "CONVERT",D," FROM BASE",B1," TO BASE",B2,!
01.10 SET I=1;SET ANS=0
01.20 SET E=D/(10↑I); SET R2=FITR[.5+(E+.00005-FITR(E))*1(
01.30 SET ANS=ANR2*B1↑(I-1); SET R2=R2*10↑(I-1)
01.40 SET D=FITR(D)-FITR(R2); IF (D) 1.5, 1.5,1.41
01.41 SET I=I+1; GOTO 1.20
01.50 SET J=20
01.60 SET I=ANS
01.70 SET A=I/B2
01.80 SET I=FITR(A)
01.90 IF [-A] 1.91, 1.93
01.91 SET D(J)=(A-I)*B2
01.92 SET J=J-1; GOTO 1.7
01.93 FOR L=J-20,-1; TYPE %2, D(21+L)
01.94 TYPE !!!;ERASE
01.95 GOTO 1.05
*@@@@@@@@@@@@@@@@@@@@@@@@@@
```

Correlation Calculations

ABSTRACT

By inputting the number of column variables, *Correlation Calculations* will calculate the correlation points.

Operational Procedures

1. Load *Correlation Calculations* via FOCAL-8.*

2. Type *GO* and input the number of column variables. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.10 D 7
01.20 G 14.1

05.10 S N=N+1;F I=1,K-1;S S(I)=S(I)+D(I);D 5.2
05.20 S SS(I)=SS(I)+D(I)*D(I);S SC(I)=SC(I)+D(1)*D(I)

07.10 T "CORRELATION CALCULATOR
07.11 T !"CORRELATES ALL COLUMN VARIABLES TO FIRST, CAPACITY 13",!;E
07.12 A "TYPE NUMBER OF COLUMN VARIABLES",K,!
07.13 T %3.2,"'Y' TO CONTINUE INPUT, 'N' TO TERMINATE INPUT",!!

10.10 T !!,"CORRELATIONS",!!,"--VAR---COR-",!
10.20 S D(1)=FSQT(N*SS(1)-S(1)*S(1))
10.21 I (D(1)) 10.25,10.4,10.25
10.25 F I=1,1,K;D 11
10.26 Q
10.40 T !,"IMPROPER MAIN",!;QUIT

11.10 S D(2)=FSQT(N*SS(I)-S(I)*S(I));I (D(2)) 11.2,11.4,11.2
11.20 S D(3)=(N*SC(I)-S(1)*S(I))/(D(2)*D(1));T I,D(3),!;RETURN
11.40 T I,0," *",!;RETURN

14.10 A "MORE?",D(1);I (D(1)-0Y) 10.1,14.2,10.1
14.20 F I=1,K;A D(I)
14.30 T !;D 5;G 14.1
*
```

Exact Factorial

ABSTRACT

Answers are occasionally required with greater precision than is possible in a single variable. One such occasion is the exact computation of a large factorial, which requires multiple precision (i.e., enough to hold all of the answers).

This routine facilitates computing the factorial. The maximum factorial that can be computed is 200.

Operational Procedures

1. Load *Exact Factorial* by FOCAL-8.*

2. Type *GO* and input the number for which the factorial is to be computed. Execution begins.

3. A sample run follows.

```
*W@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C-FOCAL,1969

01.05 ERASE
01.08 ASK !"WHAT IS THE NUMBER? "N,!!
01.09 SET Z=1; FOR I=1,N; SET Z=Z*I
01.10 TYPE %8,N," FACTORIAL IS APPROXIMATELY "%,Z,!!
01.20 TYPE !"THE EXACT ANSWER IS "!
01.30 SET S(1)=1 ; SET MAX= 1 ; SET C=1000
01.40 FOR X=1,N; DO 2
01.45 T !"       "
01.50 FOR Z=1,MAX; DO 5
01.70 T !!!; QUIT

02.10 FOR I=1,MAX; DO 3
02.20 F I=1,MAX;DO 4

03.10 SET S(I)=S(I)*X
03.20 IF (I+1-MAX)3.3,3.3; SET I=MAX+1
03.30 R

04.10 IF (S(I)-C)4.3; SET Z=FITR(S(I)/C)
04.20 SET S(I)=S(I) - C*Z
04.25 S S(I+1)=S(I+1)+Z; IF ( I-MAX)4.3 ;SET MAX=I+1
04.30 R

05.05 SET I=MAX+1-Z
05.06 T %1,",""
05.10 IF (7*FITR(Z/7)-Z) 5.2; TYPE !
05.20 FOR J=1,3; DO 6

06.10 SET X=FITR(S(I)/10↑[3-J])
06.20 TYPE X
06.30 SET S(I)=S(I)-10↑[3-J]*X
*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@



*GO

WHAT IS THE NUMBER? :101

      101 FACTORIAL IS APPROXIMATELY  0.942589E+160


THE EXACT ANSWER IS

       , 0 0 9, 4 2 5, 9 4 7, 7 5 9, 8 3 8, 3 5 9,
 4 2 0, 8 5 1, 6 2 3, 1 2 4, 4 8 2, 9 3 6, 7 4 9,
 5 6 2, 3 1 2, 7 9 4, 7 0 2, 5 4 3, 7 6 8, 3 2 7,
 8 8 9, 3 5 3, 4 1 6, 9 7 7, 5 9 9, 3 1 6, 2 2 1,
 4 7 6, 5 0 3, 0 8 7, 8 6 1, 5 9 1, 8 0 8, 3 4 6,
 9 1 1, 6 2 3, 4 9 0, 0 0 3, 5 4 9, 5 9 9, 5 8 3,
 3 6 9, 7 0 6, 3 0 2, 6 0 3, 2 6 4, 0 0 0, 0 0 0,
 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0
```

```
*GO

WHAT IS THE NUMBER? :200

      200 FACTORIAL IS APPROXIMATELY  0.788647E+375


THE EXACT ANSWER IS

        , 7 8 8, 6 5 7, 8 6 7, 3 6 4, 7 9 0, 5 0 3,
 5 5 2, 3 6 3, 2 1 3, 9 3 2, 1 8 5, 0 6 2, 2 9 5,
 1 3 5, 9 7 7, 6 8 7, 1 7 3, 2 6 3, 2 9 4, 7 4 2,
 5 3 3, 2 4 4, 3 5 9, 4 4 9, 9 6 3, 4 0 3, 3 4 2,
 9 2 0, 3 0 4, 2 8 4, 0 1 1, 9 8 4, 6 2 3, 9 0 4,
 1 7 7, 2 1 2, 1 3 8, 9 1 9, 6 3 8, 8 3 0, 2 5 7,
 6 4 2, 7 9 0, 2 4 2, 6 3 7, 1 0 5, 0 6 1, 9 2 6,
 6 2 4, 9 5 2, 8 2 9, 9 3 1, 1 1 3, 4 6 2, 8 5 7,
 2 7 0, 7 6 3, 3 1 7, 2 3 7, 3 9 6, 9 8 8, 9 4 3,
 9 2 2, 4 4 5, 6 2 1, 4 5 1, 6 6 4, 2 4 0, 2 5 4,
 0 3 3, 2 9 1, 8 6 4, 1 3 1, 2 2 7, 4 2 8, 2 9 4,
 8 5 3, 2 7 7, 5 2 4, 2 4 2, 4 0 7, 5 7 3, 9 0 3,
 2 4 0, 3 2 1, 2 5 7, 4 0 5, 5 7 9, 5 6 8, 6 6 0,
 2 2 6, 0 3 1, 9 0 4, 1 7 0, 3 2 4, 0 6 2, 3 5 1,
 7 0 0, 8 5 8, 7 9 6, 1 7 8, 9 2 2, 2 2 2, 7 8 9,
 6 2 3, 7 0 3, 8 9 7, 3 7 4, 7 2 0, 0 0 0, 0 0 0,
 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0,
 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0, 0 0 0


*
```

Least Common Multiple

ABSTRACT

Least Common Multiple, **a mathematical FOCAL-8 demonstration, computes the least common multiple for a specified set of numbers.**

The user must input the number of items to be computed, and the numbers that will be considered.

Operational Procedures

1. Load *Least Common Multiple* by FOCAL-8.

2. Type *GO.*

3. Type the requested input data and the LCM will be typed on the Teletype.*

4. A sample run follows.

```
*W
C-FOCAL,1969

01.05 T "THIS ROUTINE WILL COMPUTE THE LEAST COMMON MULTIPLE."!!
01.10 A "NUMBER OF ITEMS? ",N;F I=1,N;A ?A(I)?
01.20 S I=1  ;S MA=A(I);IF (N-1) 1.3,2.1
01.30 S I=I+1;I (MA-A(I)) 1.4,1.4,1.5
01.40 S MA=A(I)
01.50 IF (I-N) 1.3,2.1,1.3

02.10 S LC=MA;T !!!!
02.20 S I=0
02.30 S I=I+1;IF (LC/A(I)-FITR(LC/A(I))) 2.5,2.4,2.5
02.40 IF (I-N) 2.3,2.6,2.3
02.50 S LC=LC+MA;GOTO 2.2
02.60 T !,!;T %4.2,"LCM",LC,!!!
*
```

```
*GO
THIS ROUTINE WILL COMPUTE THE LEAST COMMON MULTIPLE.

NUMBER OF ITEMS? :5
A(I):100
A(I):23.9
A(I):9
A(I):6
A(I):1

LCM= 0.3069E+06
```

```
*GO
THIS ROUTINE WILL COMPUTE THE LEAST COMMON MULTIPLE.

NUMBER OF ITEMS? :3
A(I):.50
A(I):.25
A(I):.5

LCM= 0.500
```

```
*GO
THIS ROUTINE WILL COMPUTE THE LEAST COMMON MULTIPLE.

NUMBER OF ITEMS? :2
A(I):100
A(I):.5

LCM= 100.0
```

```
*GO
THIS ROUTINE WILL COMPUTE THE LEAST COMMON MULTIPLE.

NUMBER OF ITEMS? :1
A(I):10000

LCM= 0.1000E+05

*
```

Linear Programming†

ABSTRACT

Linear Programming **is used to minimize (or maximize) the value of an expression subject to certain restraints.**

This FOCAL-8 implementation permits analysis of up to a 5 x 7 array in a 4K PDP-8/L. Larger expressions may be processed with an 8K or larger configuration.*

Example

What is the best way to spend a man-year on the PDP-8 Family production line (i.e., the optimum product mix from the manufacturing point of view)?

Assume that we want to maximize the return for making a PDP-8, 8/L, and 8/I with the following assumptions:

Returns Vector:	PDP-8	8/L	8/I	
	18,000	8,000	12,000	dollars

Each uses certain RE sources differently, and those resources are limited as follows:

Coefficient Array	PDP-8	PDP-8/L	PDP-8/I	Total Limits
Number of R-series required	20	4	8	⩽ 400 modules
Number of W-series required	4	2	2	⩽ 100 modules
Number of man-weeks needed	5	1	2	⩽ 52 weeks

Assume further that a small penalty must be paid for all unused parts and labor (e.g., carrying charges) at the following rates:

R-series	=	$2	each
W-series	=	$1	ea
man-week	=	$30	eacn

Before using the program to compute the answer, make your own estimate.

The input given a FOCAL program to solve this problem consists of two parts:

1. The number of rows and columns of coefficient matrix (3x3).

2. The data for the expression to be minimized (in this case the negative of returns).

† The program was originated by Dr. E. Woolsey at the Colorado School of Mines.

If non-usage costs are positive, and the returns are negative, notice that minimizing with negative coefficients is the same as maximizing with positive coefficients.

The program uses the following symbols to indicate different quantities:

E3. = thousands
X (= 1) = number of PDP-8s to make
X (= 2) = number of PDP-8/Ls to make
X (= 3) = number of PDP-8/Is to make
X (= 0) = number of unused man-weeks
X (=-1) = number of unused W-series modules
X (=-2) = number of unused S-series modules
Z* = negative of next returns

Interpreting the example below, we find a suggested production of 48 PDP-8/Ls, 2 PDP-8/Is and no PDP-8s.

Operational Procedures

1. Load *Linear Programming* by FOCAL-8.

2. Type *GO* and respond to the request for input data. Execution begins.

3. A sample run follows.

```
@@@@@@@@@@@@@
C-FOCAL,1969

01.05 E
01.10 A !"INPUT ROWS"M,!"INPUT COLS"N,!"MINIMIZE?"!
01.20 S N1=N+1;S M1=M+1;S L=M+N;FOR J=1,L;ASK C(J)
01.25 T !"COEFFICIENTS
01.30 F I=1,M;D 8
01.34 F I=1,L;S IB(I)=I
01.37 F J=1,N;S A(M1+10*(J-1))=A(M1+10*(J-1))-C(J+M)
01.39 S TE=0;S L=0;F J=1,N;D 2
01.40 I (L)1.97,1.97;S TE=1E6;S K=;F I=1,M;D 3
01.45 I (K)1.97,1.97;S I=IB(K);S J=M+L;S IB(K)=IB(J)
01.53 S IB(J)=I;F I=1,M1;S C(I)=A(I+10*(L-1));S A(I+10*(L-1))=0
01.55 S A(K+10*(L-1))=1
01.56 F J=1,N1;S A(K+10*(J-1))=A(K+10*(J-1))/C(K)
01.57 F I=1,1,M1;D 4
01.58 G 1.39
01.97 F I=1,M;T !"X("%2,IB(I)-M,")"%8.02,A(I+10*(N1-1))
01.98 T !!"Z*",A(M1+10*(N1-1)),!!;Q

02.01 S TL=A(M1+10*(J-1));I (TL-TE),;S L=J;S TE=TL

03.01 S TL=A(I+10*(L-1));I (TL),;S TL=A(I+10*(J-1))/TL
03.02 I (TE-TL),;S TE=TL;S K=I

04.01 I (K-I)4.02,;
04.02 F J=1,N1;S A(I+10*(J-1))=A(I+10*(J-1))-A(K+10*(J-1))*C(I)

07.01 S TE=A(I+10*(J-1))*C(I);S Z=M1+10*(J-1);S A(Z)=A(Z)+TE

08.01 T !;F J=1,N1;A A(I+10*(J-1));D 7
*@@@@@@@@@@@@@@@@@@




*
*GO

INPUT ROWS:3
INPUT COLS:3
MINIMIZE?
:2 :1 :30 :-18E3 :-8E3 :-12E3
COEFFICIENTS
:20 :4 :8 :400
:4 :2 :2 :100
:5 :1 :2 :52
RESULTS
X(=- 2)=     192.00
X(=  2)=      48.00
X(=  3)=       2.00

Z*=-407616.00

*
```

```
*
*
*GO

INPUT ROWS:7
INPUT COLS:3
MINIMIZE?
:0 :0 :0 :0 : 0 :0 :0 :-55 :-150 :-700
COEFFICIENTS
:25 :5 :35 :5000
:10 :15 :5 :1000
:5 :10 :9 :300
:401←40 :10 :60 :3000
:1 :0 :0 :50
:0 :1 :0 :50
:0 :0 :1 :50
X(=- 6)=    3833.33
X(=- 5)=     833.33
X(=  3)=      33.33
X(=- 3)=    1000.00
X(=- 2)=      50.00
X(=- 1)=      50.00
X(=  0)=      16.67

Z*=- 23333.30
```

Markov Process

ABSTRACT

5 x 5 Markov Process

	A	B	C	D	E
A	0.2	0.4	0.1	0.2	0.1
B	0.1	0.3	0.4	0.2	0.0
C	0.5	0.1	0.1	0.1	0.2
D	0.6	0.05	0.05	0.2	0.1
E	0.05	0.15	0.25	0.25	0.3

Distribution changes over a period of time. At one time a particular group or item may increase, at another time a decrease may be observed. One can calculate the steady-state vector of the distribution; therefore, one can judge an average flow for a specific group over a designated period.

Markov Process, **a mathematical process, calculates the rate of distribution over a specific period, and concludes with a steady-state vector for the requested time allotment.**

Operational Procedures

1. Load *Markov Process* by FOCAL-8.*

2. Type *GO* and supply the appropriate input data. Execution begins immediately.

3. A sample run follows.

```
*
*WRITE ALL
C-FOCAL,1969

01.01 G 2.01
01.02 F I=1,U;T !;F J=1,U+1;A A(I+U*(J-1))
01.10 S N=0
01.13 S N=N+1
01.14 S K=N-1
01.15 S K=K+1;I (K-U)1.95,1.95,1.90
01.17 I (K-N)1.18,1.30,1.18
01.18 FOR M=N,1,U+1;D 1.97
01.30 S R=A(N+U*(N-1));F J=N,U+1;S A(N+U*(J-1))=A(N+U*(J-1))/R
01.31 I (K+1-U)1.32,1.32,1.33
01.32 F I=K+1,1,U;F J=N+1,1,U+1;D 1.98
01.33 I (N-U)1.13,1.34,1.13
01.34 S I=U+1
01.35 S I=I-1;I (I-1)1.50;
01.36 S Y(I)=A(I+U*U)/A(I+U*(I-1)); S K=I
01.37 S K=K-1;I (K-1)1.35;
01.38 S A(K+U*U)=A(K+U*U)-A(K+U*(I-1))*Y(I)
01.39 G 1.37
01.50 F J=1,U;T !"P(0,",%2.00,J,")",%,Y(J)
01.51 Q
01.90 T !!"YOU GOOFED";Q
01.95 I (A(K+U*(N-1)))1.17,1.15,1.17
01.96 S A(N+U*(J-1))=A(N+U(J-1))/A(N+U*(N-1))
01.97 S TE=A(N+U*(M-1));S A(K+U*(M-1));S A(K+U*(M-1))=TE
01.98 S A(I+U*(J-1))=A(I+U*(J-1))-A(I+U*(N-1))*A(N+U*(J-1))
01.99 I (A(K+U*(N-1)),

02.01 A !"MARKOV STEADY-STATE",!!,"HOW MANY STATES?",U
02.04 F I=1,U;D 2.99
02.05 G 3.01
02.98 F J=1,U;A A(J+U*(I-1))
02.99 T !!"SUCESSIVE PROBABILITIES IN STATE",%2.00,I;D 2.98

03.01 F I=1,U;F J=1,U;D 5.0
03.02 F J=1,U+1; S A(1+U*(J-1))=1.0
03.03 S A(U+U*(U))=0.0
03.07 G 1.10

05.01 I (I-J),5.02,
05.02 S A(I+U*(J-1))=A(I+U*(J-1))-1
*
```

```
*GO

MARKOV STEADY-STATE

HOW MANY STATES?:4


SUCESSIVE PROBABILITIES IN STATE=  1:.080 :.184 :.368 :.368


SUCESSIVE PROBABILITIES IN STATE=  2:.632 :.368 :0 :0


SUCESSIVE PROBABILITIES IN STATE=  3:.264 :.368 :.368 :0


SUCESSIVE PROBABILITIES IN STATE=  4:.080 :.184 :.368 :.368

P(0,=  1)= 0.255192E+00
P(0,=  2)= 0.265282E+00
P(0,=  3)= 0.176466E+00
P(0,=  4)= 0.303061E+00*
```

Polynomial Expansion Program

ABSTRACT

By inputting the value of the exponent, ***Polynomial Expansion Program*** **computes the coefficient of (X+Y) ↑ N.**

Operational Procedures

1. Load this demonstration program by FOCAL-8.*

2. Type *GO* and when asked, state the exponent. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.01 TYPE !!, "WHAT IS THE EXPONENT?" !,; ASK EX;
01.02 FOR J=0,EX; DO 4;
01.03 FOR JB=0,EX; DO 5;
01.04 GOTO 1.01;

02.01 SET P=1;
02.02 FOR I=N-R+1,N; SET P=P*I;
02.03 RETURN;

03.01 SET PQ=1;
03.02 IF (R) RETURN,3.20,3.03;
03.03 FOR I=1,R; SET PQ=PQ*I;
03.04 RETURN;
03.20 SET PQ =1;
03.21 RETURN;

04.01 SET N=EX;
04.02 SET R=J;
04.03 DO 2;
04.04 DO 3;
04.06 SET K=J+1;
04.10 SET CO(J)=P/PQ;
04.11 SET JA=EX-J;
04.12 SET CO(JA)=CO(J);
04.13 RETURN;

05.01 SET K=JB+1;
05.02 TYPE ! , "COEF" %3 , K, " IS " % , CO(JB);
05.03 RETURN;

06.10 TYPE !!,"POLYNOMIAL EXPANSION PROGRAM - PROJECT ASC - G.FORD",!
06.20 TYPE "     COMPUTES COEFFICIENTS OF (X+Y)↑N"
*
```

```
*GO


WHAT IS THE EXPONENT?
:2

COEF=    1 IS = 0.100000E+01
COEF=    2 IS = 0.200000E+01
COEF=    3 IS = 0.100000E+01

WHAT IS THE EXPONENT?
:-3

COEF=    1 IS =-0.200000E+01

WHAT IS THE EXPONENT?
:1

COEF=    1 IS = 0.100000E+01
COEF=    2 IS = 0.100000E+01

WHAT IS THE EXPONENT?
:0

COEF=    1 IS = 0.100000E+01

WHAT IS THE EXPONENT?
:5.3

COEF=    1 IS = 0.187391E+01
COEF=    2 IS = 0.720734E+01
COEF=    3 IS = 0.125345E+02
COEF=    4 IS = 0.125345E+02
COEF=    5 IS = 0.720734E+01
COEF=    6 IS = 0.187391E+01

WHAT IS THE EXPONENT?
:-10

COEF=    1 IS =-0.900000E+01

WHAT IS THE EXPONENT?
:
```

Repeating Decimals

ABSTRACT

This FOCAL-8 routine computes and types the repeating decimals of a fraction. The user must input the numerator and the denominator, respectively.

If the output appears to be repeating for a line or two, interrupt the output by typing a CONTROL-C (C). FOCAL will give an error message and an asterisk (*). Type *GO* to continue.

Operational Procedures

1. Load *Repeating Decimals* with FOCAL-8.

2. Type *GO.**

3. Input the numerator and denominator followed by a carriage return; the results will be typed on the Teletype.

4. A sample run follows.

```
?00.00
*ERASE ALL
*W
C-FOCAL,1969
*
*C-FOCAL,1969
*
*01.05 ASK    " ENTER NUMERATOR AND DENOMINATOR "A,B,!
*01.10 SET Z=5
*01.20 IF (B-A)1.4,1.3; TYPE "       0 ."; GOTO 2.1
*01.30 TYPE !"1"!;QUIT
*01.40 TYPE !"THIS PROGRAM ONLY EVALUATES FRACTIONS<1"!;QUIT
*
*02.10 SET N=10
*02.20 IF (N*A-B) 2.3,4.1,4.1
*02.30 SET N=10*N
*02.40 TYPE 0.0;D 6
*02.50 GOTO 2.2
*02.87
*
*04.10 SET C=1
*04.20 IF (N*A-C*B) 5.1
*04.30 SET C=C+1
*04.40 GOTO 4.2
*
*05.10 TYPE %1,C-1; DO 6
*05.20 SET A=N*A-(C-1)*B
*05.30 IF (-A) 5.5;TYPE !; QUIT
*05.50 IF (A-B) 2.1,1.3,1.4
*
*06.10 IF (Z-20) 6.2; SET Z=0; TYPE !
*06.20 SET Z=Z+1;RETURN
```

```
**
*GO
 ENTER NUMERATOR AND DENOMINATOR :1,:4

      Ø .= 2= 5
*GO
 ENTER NUMERATOR AND DENOMINATOR :134,:25Ø

      Ø .= 5= 3= 6
*GO
 ENTER NUMERATOR AND DENOMINATOR :1,:5

      Ø .= 2
*GO
 ENTER NUMERATOR AND DENOMINATOR :25,:75

      Ø .= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3= 3
=  3= 3= 3= 3= 3= ?Ø1.ØØ @ Ø4.2Ø
*
```

Roots of a Quadratic

ABSTRACT

Given values of a, b, c of a first-degree quadratic equation, this FOCAL-8 demonstration program computes the roots of the equation.

Based on the quadratic equation theorem: given $ax^2+bx+c=0$, then $x=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$

If a, b, and c are real, the following principles are applied:

- *a.* **If b^2-4ac is positive, the roots are real and unequal.**
- *b.* **If b^2-4ac is 0, the roots are real and equal.**
- *c.* **If b^2-4ac is negative, the roots are imaginary and unequal.**

Operational Procedures

1. Load *Roots of a Quadratic* with FOCAL-8.**

2. Type *GO* and input the values of a, b, and c. Execution begins.

3. A sample run follows.

```
C-FOCAL,1969

01.10 ASK !!  ?A B C ?; SET ROOT=B↑2-4*A*C
01.20 IF (A) 1.4,1.3,1.4
01.30 TYPE ! "THIS IS A FIRST DEGREE EQUATION" !; GOTO 1.1
01.40 TYPE %6.03, ! " THE ROOTS ARE"; IF (ROOT) 1.7,1.6
01.50 TYPE  !,(-B+FSQT(ROOT))/2*A,!,(-B-FSQT(ROOT))/2*A;GOTO 1.1
01.60 TYPE ! -B/2*A,!; GOTO 1.1
01.70 TYPE " IMAGINARY "!,-B/2*A,"    +  (",FSQT(-ROOT)/2*A,")" "*
01.80 TYPE !,-B/2*A,"    -  ('',FSQT(-ROOT)/2*A,")*I",!;GOTO 1.1
*
```

```
*GO


A :2
B :5
C :2

 THE ROOTS ARE
=-   0.500
=-   2.000

A :1
B :2
C :1

 THE ROOTS ARE
=-   1.000


A :4
B :2
C :4

 THE ROOTS ARE IMAGINARY
=-   0.250    +   (=    0.968)*I
=-   0.250    -   (=    0.968)*I
```

Statistics

ABSTRACT

Statistics **is a FOCAL-8 program that takes statistical parameters of data and calculates the standard deviation, the mean deviation, and plots, and lists the data on a standard deviation curve.**

Operational Procedures

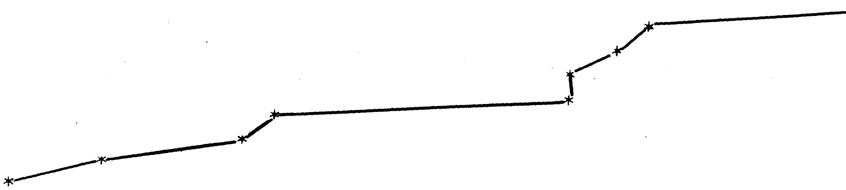
1. Load *Statistics* by FOCAL-8.*

2. Type *GO*. The program requests the number of items of data and the data, and execution begins.

3. A sample run follows.

```
C-FOCAL,1969

01.01 ERASE
01.10 ASK !!"HOW MANY DATA ARE THERE?" XN,!
01.20 TYPE !"PLEASE LIST YOUR DATA."!
01.30 FOR I=1,XN; ASK X(I)
01.40 TYPE !"OK"!;FOR I=1,XN; DO 3
01.55 SET XMU=SMX/XN
01.60 SET VAR=SQX/XN-XMU↑2
01.80 TYPE %6.03,!!"MEAN     ",XMU,!"VARIANCE"VAR
01.81 TYPE !!"STANDARD DEVIATION " FSQT(VAR),!!
01.85 FOR I=1,XN;SET MD=MD+FABS(X(I)-XM)
01.86 TYPE "    MEAN DEVIATION  "MD/XN,!
01.90 TYPE !"SORTING ...
01.91 SET X(0)=1E100;SET X(XN+1)=-X(0); FOR I=1,XN; DO 4
01.92 TYPE !"     THE MEDIAN IS  ";IF (FITR<XN/2>*2-XN) 1.93,1.94
01.93 SET C=X((XN-1)/2 + 1) ; GOTO 1.95
01.94 SET C=<X(XN/2)+X(XN/2+1)>/2
01.95 TYPE C,!!! "THE DATA ARE "!!

02.10 SET C=60/<X(1)-X(XN)>
02.20 FOR I=1,XN; TYPE "*"!X(I);FOR K=0,C*<X(I)-X(XN)>;TYPE " "
02.40 TYPE "*"!!!!!;RETURN

03.10 SET SMX=SMX+X(I)
03.20 SET SQX=SQX+X(I)↑2

04.05 SET K=I
04.10 IF (X(I+1)-X(I))4.3;
04.20 SET C=X(I);SET X(I)=X(I+1); SET X(I+1)=C; SET I=I-1; GOTO 4.1
04.30 SET I=K
*
```

```
*30

HOW MANY DATA ARE THERE?:1Ø


PLEASE LIST YOUR DATA.
:.2,:.4,:.7Ø,:2.1,:1.5,:.75,:1.45,:1.35,:1.352,:2.Ø

OK


MEAN       =    1.18Ø
VARIANCE=    Ø.374

STANDARD DEVIATION =    Ø.611

   MEAN DEVIATION   =    Ø.534

SORTING . . .
     THE MEDIAN IS   =    1.351
THE DATA ARE
*    2.1ØØ
=    2.ØØØ
=    1.5ØØ
=    1.45Ø
=    1.352
=    1.35Ø
=    Ø.75Ø
=    Ø.7ØØ
=    Ø.4ØØ
=    Ø.2ØØ
```



```
PLEASE LIST YOUR DATA.
:.4,:.6.:1.8,:.5.:2.1,:.25,:.3

OK


MEAN       =    Ø.85Ø
VARIANCE=    Ø.5Ø2

STANDARD DEVIATION =    Ø.7Ø9

   MEAN DEVIATION   =    Ø.629

SORTING ...
     THE MEDIAN IS   =    Ø.5ØØ


THE DATA ARE

*
=    2.1ØØ
=    1.8ØØ
=    Ø.6ØØ
=    Ø.5ØØ
=    Ø.4ØØ
=    Ø.3ØØ
=    Ø.25Ø
```

IV. PROBLEM SOLVING ROUTINES

Calculating Survival Rates†

ABSTRACT

This program computes the survival rates of subjects alive *N* periods after diagnosis or treatment.

Input:

N	=	Number of periods after diagnosis
L(1)	=	Initial number of subjects
D(I)	=	Died during period I
U(I)	=	Lost (survival status unknown) during period I
W(I)	=	Withdrawn alive during period I

The program computes the following information:

L(I) = Alive at beginning of period I

$$L(I) = L(I\text{-}1) - \left(D(I-1) + U(I-1) + W(I-1)\right)$$

E(I) = Effective number exposed to risk of dying during period I

$$E(I) = L(I) - \left(U(I) + W(I)\right)/2$$

Q(I) = Proportion dying during period I

$$Q(I) = D(I) \,/\, E(I)$$

P(I) = Proportion surviving during period I

$$P(I) = 1 - Q(I)$$

R(I) = Ith period survival rate

$$R(I) = P(1) * P(2) * \ldots *P(I)$$

S(I) = Standard error of survival rate

$$S(I) = R(I)\sqrt{\sum_{j=1}^{i} \frac{QJ}{Ej-Dj}}$$

† For reference see S. J. Cutler and F. Ederer, "Maximum Utilization of the Life Table Method in Analyzing Survival," Chronic Diseases, December 1958; pp. 699-712.

rational Procedures

Load *Calculating Survival Rates* with FOCAL-8.

Type *GO* and input the number of periods followed by a carriage return and the number of subjects followed by a carriage return.*

Type in the number of:

a. Periods followed by a space
b. Subjects who died followed by a space
c. Subjects lost followed by a space
d. Subjects withdrawn followed by a space

After all periods have been given, type a 0 when it requests a period. (I:), followed by a carriage return and input data will be complete.

After computing the input data, the output will be dumped on the Teletype.

A sample run follows.

```
RITE ALL
FOCAL,1969

.15 T !"SURVIVAL RATES!
.20 A !!"NUMBER OF PERIODS"N,!
.30 S D(0)=0;S U(0)=0;S W(0)=0;S R(0)=1;S SUM(0)=0
.40 A "INITIAL NUMBER OF SUBJECTS"L(1),!
.45 S L(0)=L(1);T "                DIED   LOST   WITHDRAWN
.50 FOR I=1,N; T %3,!,I; ASK ?  D(I) U(I) W(I)?,!
.65 T !"        ALIVE                       EXPOSED"!
.66 T "         AT                  WITH- TO RISK    %      %
1.67 T "    SURV.  STD.
1.68 T !"PERIOD START     DIED     LOST    DRAWN OF DYING DYING
1.69 T "  SURV.  RATE   ERROR"!
1.70 F I=1,N;D 2
1.80 T !!;Q

2.10 S L(I)=L(I-1)-(D(I-1)+U(I-1)+W(I-1))
2.20 S E(I)=L(I)-(U(I)+W(I))/2
2.30 S Q(I)=D(I)/E(I)
2.40 S P(I)=1-Q(I)
2.50 S R(I)=R(I-1)*P(I)
2.60 S SUM(I)=SUM(I-1)+Q(I)/(E(I)-D(I))
2.70 S S(I)=R(I)*FSQT(SUM(I))
2.80 T !,%2,I,%6,L(I),D(I),U(I),W(I),E(I),%4.02,Q(I),P(I),R(I),S(I)
```

```
GO

SURVIVAL RATES!

NUMBER OF PERIODS:4

INITIAL NUMBER OF SUBJECTS:123

              DIED    LOST   WITHDRAWN
:    1   D(I) :5,U(I) :10,W(I):1,

:    2   D(I) :6,U(I) :8,W(I):3,

:    3   D(I) :10,U(I) :7,W(I):5,

:    4   D(I) :15,U(I) :10,W(I):10,

       ALIVE                        EXPOSED
        AT                   WITH-  TO RISK    %        %       SURV.   STD.
PERIOD START   DIED   LOST  DRAWN  OF DYING  DYING    SURV.    RATE    ERROR

   1=     123=     5=    10=      1=      118=   0.04=   0.96=   0.96=   0.02
   2=     107=     6=     8=      3=      102=   0.06=   0.94=   0.90=   0.03
   3=      90=    10=     7=      5=       84=   0.12=   0.88=   0.79=   0.04
   4=      68=    15=    10=     10=       58=   0.26=   0.74=   0.59=   0.06
```

Interest Payments

ABSTRACT

Interest Payments **is a FOCAL-8 demonstration program that is not only an interesting program, but it also serves a useful purpose. "Interest Payments" will calculate monthly payments on a loan given the following itemized data:**

a. **Interest (in percent)**
b. **Amount of the loan**
c. **Number of payments per year.**

Operational Procedures

1. Load *Interest Payments* with FOCAL-8.*

2. Type *GO* and respond to the dialogue. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.10 ASK !!%7.02,"ENTER INTEREST IN PERCENT" J,!
01.14 SET J=J/100
01.16 ASK "ENTER AMOUNT OF LOAN"A,!
01.20 ASK "NO. OF YEARS"N,!
01.24 ASK "NO. OF PAYMENTS/YR" M,!
01.30 SET N=N*M; SET I=J/M
01.34 SET B=1+I
01.40 SET R=A*I/(1-1/B↑N)
01.42 TYPE "MONTHLY PAYMENT "R,!
01.48 TYPE "TOTAL INTEREST" R*N-A,!

02.05 SET B=A
02.10 TYPE "INT.            APP TO PRIN                BALANCE",!
02.12 SET L=B*I; SET P= R- L
02.16 SET B = B-P
02.18 TYPE L,"        "P,"                "B,!
02.20 IF (B-R) 2.24,2.24,2.12
02.24 TYPE B*I,"        "R-B*I,! "LAST PAYMENT!" B*I+B,!!!;GOTO 1.1
*
```

```
;O

NTER INTEREST IN PERCENT:8.5

NTER AMOUNT OF LOAN:1800

). OF YEARS:2

). OF PAYMENTS/YR:12

ONTHLY PAYMENT =     81.82
OTAL INTEREST=   163.75
NT.          APP TO PRIN              BALANCE
    12.75       =     69.07          =  1730.93
    12.26       =     69.56          =  1661.37
    11.77       =     70.06          =  1591.31
    11.27       =     70.55          =  1520.76
    10.77       =     71.05          =  1449.71
    10.27       =     71.56          =  1378.15
     9.76       =     72.06          =  1306.09
     9.25       =     72.57          =  1233.52
     8.74       =     73.09          =  1160.43
     8.22       =     73.60          =  1086.83
     7.70       =     74.13          =  1012.71
     7.17       =     74.65          =   938.06
     6.65       =     75.18          =   862.88
     6.11       =     75.71          =   787.17
     5.58       =     76.25          =   710.92
     5.04       =     76.79          =   634.13
     4.49       =     77.33          =   556.80
     3.94       =     77.88          =   478.92
     3.39       =     78.43          =   400.49
     2.84       =     78.99          =   321.50
     2.28       =     79.55          =   241.96
     1.71       =     80.11          =   161.85
     1.15       =     80.68          =    81.17
     0.58       =     81.25
AST PAYMENT!=    81.75
```

Inventory Schedule

ABSTRACT

Inventory Scheduler **assists the manager in scheduling the distribution of a particular product over a specified period. By inputting the requested data, the optimum cost of the production schedule is calculated and typed on the Teletype. This FOCAL program requests the following input data.**

a. **Holding cost per item per day**
b. **Set-up cost per production run**
c. **The number of items needed for distribution for a particular period**
d. **Shortage cost per item per day.**

Given the above data, the *Inventory Scheduler* **dumps the following information:**

a. **Optimal time interval between production runs**
b. **Optimum order quantity**
c. **Optimum cost of the production schedule.**

Operating Procedures

1. Load the FOCAL program using the described loading procedures for low/high speed reader from the FOCAL manual.*

2. Type *GO* and answer the dialogue.

3. Sample run of the *Inventory Scheduler* follows.

C-FOCAL

```
01.01 T !!"INVENTORY MODEL WITH SHORTAGES"!!
01.02 T "PLEASE INPUT HOLDING COST PER ITEM PER DAY";ASK C1
01.03 T !!"PLEASE INPUT SETUP COST PER PRODUCTION RUN";ASK CS
01.04 T !!"HOW MANY ITEMS ARE NEEDED";ASK R
01.05 T !!"OVER WHAT TIME PERIOD (IN DAYS)";ASK T
01.07 T !!"PLEASE INPUT SHORTAGE COST PER ITEM PER DAY";ASK C2
01.09 SET TS=FSQT(((2*T*CS)*(C1+C2))/(R*C1*C2)); SET D=FITR(TS)
01.10 SET U=(TS-D)*24; SET H=FITR(U); SET M=FITR((U-H)*60)
01.12 T !!"OPTIMAL INTERVAL BETWEEN PRODUCTION RUNS IS:"!
01.14 TYPE %3.00,D," DAYS ",H," HOURS ",M," MINUTES "
01.15 SET Q=FSQT((2*R*CS*(C1+C2))/(T*C1*C2));IF (Q-FITR(Q))1.17,1.17;
01.16 SET Q=Q+1
01.17 T !!"THE OPTIMUM ORDER QUANTITY IS:",%8.00,Q
01.18 SET Q=FSQT((2*R*T*C1*CS*C2)/(C1+C2))
01.19 T !!"OPTIMUM COST OF THIS PRODUCTION SCHEDULE IS:",%7.02,Q
01.20 T !!"DO ANOTHER?";ASK YN;IF (YN-0YES)1.21,1.01,1.21
01.21 QUIT
*
```

Max*Flow-Min*Cut

ABSTRACT

Preliminary: *1.* Designate the source node as "s"
2. Designate the sink node as "t"
3. Number the intermediate nodes from 1 to N.

Goal: To find a maximal flow from s to t.

1. Remove all labels and scan marks from nodes.

2. Label the source node s, (–,00).

3. Labeling Process

Select a labeled unscanned node, x where x may be node s,1,2,...N. It is labeled (z±, e(x)), or (–,00) if node s. To all nodes y that are unlabeled (and connected to node x) and that flow (x,y) <capacity (x,y), assign the label, (x+,e(y)), where: e(y) = minimum of (capacity (x,y)–flow (x,y), or e(x)0.

To all nodes w that are unlabeled (and connected to node s) and that flow (w,x) > 0, (i.e., a backward flow), assign the label (x–,e(w)), where: e(w) = minimum of (flow (w,x), or e(x)). If node t is labeled in this step, go to step 4. If node t is not labeled after all nodes connected to node x have been checked for labeling, mark node x *scanned*. If there are other labeled unscanned nodes, go to step 3. If not, go to step 9.

4. Flow Change

This occurrence is called *breakthrough*. A flow augmenting path is used to increase the flow from s to t. The sink t is labeled (z±,e(t)) . Replace t by x in the label. Go to step 5.

5. Node x is labeled (z±,e(x)). If the label is z+, go to step 6. If the label is z–, go to step 7. In either case replace z by y.

6. Node x is labeled (y+,e(x)). Increase the flow on arc (y,x) by an amount e(t). If node y is the source node s, go to step 8. If not, replace x by y and go to step 5.

7. Node s is labeled (y–,e(x)). Decrease the flow on arc (x,y) by an amount e(t). If node y is the source node s, go to step 8. If not, replace x by y and go to step 5.

8. The flow has been increased along a flow augmenting path from s to t. To seek another flow augmenting path, go to step 1.

9. **Maximal Flow-Minimal Cut**

This occurrence is called *nonbreakthrough*. A maximal flow has been found and is assigned to each arc. A minimal cut has also been found. The set X for the cut consists of the nodes that have been labeled. The set of arcs $(X,\bar{X})$ is the minimal cut.

Load Procedures

1. Load *Max * Flow-Min * Cut* with FOCAL-8 without the extended functions.***

2. Type *GO* and execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.10 TYPE !!"MAX-FLOW MIN-CUT"!!
01.20 TYPE "ENTER DATA AS:SOURCE,SINK,CAPACITY; END WITH:0,0,0"!
01.40 ASK I,J,A; SET K=I+10*(J-1); IF (I-1)1.60; SET NI(K)=1
01.50 SET C(K)=A; IF (J-NN)1.40,1.40;SET NN=J; GOTO 1.40
01.60 SET L(1)=9999; SET P(1)=9999;SET N=NN; DO 10.1;DO 11
01.61 SET J=0
01.62 SET J=J+1;IF (J-NN)1.63,1.63,1.90
01.63 IF (L(J))1.64,1.62,1.64
01.64 IF (S(J)-1)1.65,1.62,1.65
01.65 SET I=0
01.66 SET I=I+1;SET K=J+10*(I-1);IF (I-NN)1.67,1.67,1.85
01.67 IF (NI(K)-1)1.66;IF (L(I))1.66,1.68,1.66
01.68 SET QT=C(K)-R(K);IF (QT)1.69,1.69,1.76
01.69 IF (NI(K)-1)1.70,1.66;
01.70 IF (R(K))1.71;GOTO 1.66
01.71 SET L(I)=-J;IF (R(K)-P(J))1.83; SET P(I)=P(J);GOTO 1.84
01.76 SET L(I)=J;IF (QT-P(J))1.77;SET P(I)=P(J);GOTO 1.84
01.77 SET P(I)=QT;GOTO 1.84
01.83 SET P(I)=R(K)
01.84 IF (L(NN))2.50,1.66,2.50
01.85 SET S(J)=1;GOTO 1.62
01.90 SET I=0
01.91 SET I=I+1; IF (I-NN)1.92,1.92,6.01
01.92 IF (L(I))1.93,1.91,1.93
01.93 IF (S(I)-1)1.61,1.91,1.61

02.50 SET AD=P(NN)
02.51 SET JK=L(NN);SET K=JK+10*(NN-1);IF (L(NN))2.52;GOTO 2.53
02.52 SET R(K)=R(K)-AD;GOTO 2.54
02.53 SET R(K)=R(K)+AD
02.54 SET NN=L(NN);IF (L(NN)-9999)2.51,2.55,2.51
02.55 SET NN=N;DO 10.2;DO 11;FOR I=2,1,NN;DO 2.57
02.56 GOTO 2.58
02.57 SET S(I)=0;SET L(I)=0
02.58 SET S(1)=0; GOTO 1.61

06.01 DO 10.3; DO 11;QUIT

10.10 TYPE !!"NET AT START"
10.20 T !!"NET AT BREAKTHRU"
10.30 T !!"NET AT OPTIMUM"

11.01 TYPE !!"FROM NODE, TO NODE, CAPACITY,    FLOW"
11.02 FOR I=1,1,NN;DO 11.03
11.03 FOR J=1,1,NN;DO 12
11.04

12.01 SET K=I+10*(J-1);IF (NI(K))12.03,12.03;
12.02 TYPE !,%7,I,J,%8,C(K),R(K)
12.03
*
```

```
MAX-FLOW MIN-CUT

ENTER DATA AS:SOURCE,SINK,CAPACITY; END WITH:0,0,0
:1,:2,:3
:1,:3,:1
:2,:3,:1
:2,:4,:4
:3,:2,:1
:3,:4,:1
:0,:0,:0


NET AT START

FROM NODE, TO NODE, CAPACITY,     FLOW
=          1=          2=           3=          0
=          1=          3=           1=          0
=          2=          3=           1=          0
=          2=          4=           4=          0
=          3=          2=           1=          0
=          3=          4=           1=          0

NET AT BREAKTHRU

FROM NODE, TO NODE, CAPACITY,     FLOW
=          1=          2=           3=          3
=          1=          3=           1=          0
=          2=          3=           1=          0
=          2=          4=           4=          3
=          3=          2=           1=          0
=          3=          4=           1=          0

NET AT BREAKTHRU

FROM NODE, TO NODE, CAPACITY,     FLOW
=          1=          2=           3=          3
=          1=          3=           1=          1
=          2=          3=           1=          0
=          2=          4=           4=          3
=          3=          2=           1=          0
=          3=          4=           1=          1

NET AT OPTIMUM

FROM NODE, TO NODE, CAPACITY,     FLOW
=          1=          2=           3=          3
=          1=          3=           1=          1
=          2=          3=           1=          0
=          2=          4=           4=          3
=          3=          2=           1=          0
=          3=          4=           1=          1*
*@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

Minimize Late Jobs

ABSTRACT

By entering the number of jobs to be done, this routine will calculate the optimum job sequence. For each job, specifically state the time and due date. This information is dumped on the Teletype.

Operational Procedures

1. Load *Minimize Late Jobs* via FOCAL-8.*

2. Type *GO* and respond to the initialization dialogue. Execution begins.

3. A sample run follows.

```
*W
C-FOCAL,1969
01.01 T !!"SCHEDULE TO MINIMIZE LATE JOBS"!!"ENTER NUMBER OF "
01.02 T "JOBS TO BE DONE";A N;T !!"FOR EACH JOB ENTER PROCESSING "
01.03 T "TIME AND DUE DATE SEPARATED BY COMMAS.";F I=1,N;D 1.98
01.04 S NK=N
01.05 S II=1
01.06 S L=999;F I=1,N; D 1.99
01.07 S CS(II)=IO;S CD(II)=D(IO);S D(IO)=1000
01.08 S CP(II)=P(IO);S II=II+1;I (II-N)1.06,1.06;
01.10 F J=1,N;S P(J)=CP(J);S D(J)=CD(J)
01.12 S H=0;S I=0
01.15 S I=I+1;I (I-NK)1.20,1.20,1.90
01.20 S H=H+P(I);I (H-D(I))1.15,1.15;
01.21 T !"LATE JOB IS",CS(I)
01.25 S H=0;F J=1,I;D 1.97
01.30 S L=P(JO);S H=CS(JO);S IO=D(JO)
01.35 F J=JO,N-1;S CS(J)=CS(J+1);S D(J)=D(J+1);S P(J)=P(J+1)
01.40 S P(N)=L;S CS(N)=H;S D(N)=IO;S NK=NK-1;G 1.12
01.90 T !!"THE OPTIMUM SEQUENCE IS:";F J=1,N;T !,CS(J)
01.91 T !!,NK," JOBS ARE ON TIME,",N-NK," JOBS ARE LATE."
01.92 T !!"DO AGAIN?";A YN;I (YN-0YES)1.93,1.01;
01.93 Q
01.97 I (P(I)-H),;S JO=J
01.98 T !"JOB ",%2.00,I;A P(I),D(I)
01.99 I (L-D(I)),;S IO=I;S L=D(I)
*
```

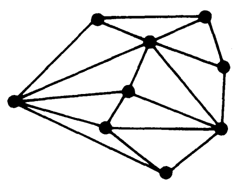
```
*GO
SCHEDULE TO MINIMIZE LATE JOBS

ENTER NUMBER OF JOBS TO BE DONE:4


FOR EACH JOB ENTER PROCESSING TIME AND DUE DATE SEPARATED BY COMMAS.
JOB =   1:6,:10,
JOB =   2:12,:15,
JOB =   3:5,:6,
JOB =   4:8,:9,
LATE JOB IS=  4
LATE JOB IS=  1
LATE JOB IS=  2

THE OPTIMUM SEQUENCE IS:
=   3
=   4
=   1
=   2

=   1 JOBS ARE ON TIME,=  3 JOBS ARE LATE.

DO AGAIN?:YES


*GO
SCHEDULE TO MINIMIZE LATE JOBS

ENTER NUMBER OF JOBS TO BE DONE:7


FOR EACH JOB ENTER PROCESSING TIME AND DUE DATE SEPARATED BY COMMAS.
JOB =   1:3,:5,
JOB =   2:4,:6,
JOB =   3:9,:10,
JOB =   4:1,:9,
JOB =   5:4,:12,
JOB =   6:6,:7,
JOB =   7:7,:18,
LATE JOB IS=  2
LATE JOB IS=  6
LATE JOB IS=  3

THE OPTIMUM SEQUENCE IS:
=   1
=   4
=   5
=   7
=   2
=   6
=   3

=   4 JOBS ARE ON TIME,=  3 JOBS ARE LATE.

DO AGAIN?:NO
*
```

Minimal Spanning Tree Algorithm[†]

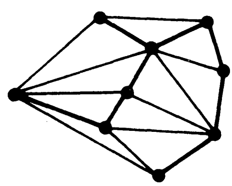
ABSTRACT

Arbitrarily select any node and connect it to the nearest node. Identify the unconnected node that is closest to a connected node, and connect these two nodes. Repeat this until all nodes have been connected.

Example: **The Quicksand Oil Company wishes to dig ditches to connect control cable to wells in a given field. Ditch digging in west Texas is quite expensive, therefore, the company wishes to dig the ditches (in terms of length) as cheaply as possible. A map of the allowed ditches in the field is shown below.**



The optimal solution is shown below.



[†]For reference see F. S. Hillier, G. J. Leiberman, Introduction to Operations Research, Holden-Day, Inc., San Francisco, 1967; p. 223.

Operational Procedures

1. Load demonstration program by FOCAL-8 without the extended functions.

2. Type *GO* and input the requested data. Execution begins immediately.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.10 ERASE
01.20 SET NN=1
01.40 TYPE !!"MINIMAL SPANNING TREE"!!!"ENTER ARCS AS FOLLOWS:"!!
01.50 TYPE "STARTING NODE, ENDING NODE, DISTANCE"!!
01.60 TYPE "STOP INPUT BY TYPING:0,0,0"!!
01.80 ASK I,J,A,!; IF (I-1)1.90;SET NI(I+10*J)=1;SET D(I+10*J)=A
01.81 IF (J-NN)1.80,1.80;SET NN=J;GOTO 1.80
01.90 SET M=1; SET ICON(1)=1
01.95 SET H=1E6; SET I=0
01.96 SET I=I+1; IF (I-M)2.01,2.01,5.01

02.01 SET K=IC(I); SET J=0
02.02 SET J=J+1;IF (J-NN)3.01,3.01,1.96

03.01 SET L=0
03.02 SET L=L+1; IF (L-M)4.01,4.01,3.03
03.03 IF (K-J)3.04,2.02,3.07
03.04 IF (NI(K+10*J))3.05,2.02,3.05
03.05 IF (D(K+10*J)-H)3.06,3.06,2.02
03.06 SET H=D(K+10*J); GOTO 3.10
03.07 IF (NI(J+10*K))3.08,2.02,3.08
03.08 IF (D(J+10*K)-H)3.09,3.09,2.02
03.09 SET H=D(J+10*K)
03.10 SET IO=K; SET JO=J
03.90 GOTO 2.02

04.01 IF (J-IC(L))3.02,3.90,3.02

05.01 SET M=M+1
05.02 IF (M-NN)5.03,5.03,5.90
05.03 IF (IO-JO)5.04,5.04,5.05
05.04 SET NI(IO+10*JO)=2; GOTO 5.06
05.05 SET NI(JO+10*IO)=2
05.06 SET IC(M)=JO
05.07 GOTO 1.95
05.90 TYPE "DONE"
05.91 TYPE !!"THE M.S.T. CONSISTS OF THE FOLLOWING ARCS:"!!
05.92 FOR I=1,NN;DO 5.93
05.93 FOR J=1,NN; DO 6
05.94 ASK !!"DO YOU WISH TO DO ANOTHER? "YS,!
05.95 IF (0YES-YS)5.96,1.1;
05.96 QUIT

06.01 IF (NI(I+10*J)-2)6.02; TYPE !"ARC(",%2,I,",",J,")"
06.02 S R=2.5
*
```

```
*GO
MINIMAL SPANNING TREE


ENTER ARCS AS FOLLOWS:

STARTING NODE, ENDING NODE, DISTANCE

STOP INPUT BY TYPING:0,0,0

:1,:2,:3,
:1,:3,:4,
:1,:4,:5,
:2,:3,:4,
:2,:4,:5,
:0,:0,:0,
DONE

THE M.S.T. CONSISTS OF THE FOLLOWING ARCS:

ARC(=   1,=   2)
ARC(=   2,=   3)
ARC(=   2,=   4)

DO YOU WISH TO DO ANOTHER? :YES


*GO
MINIMAL SPANNING TREE


ENTER ARCS AS FOLLOWS:

STARTING NODE, ENDING NODE, DISTANCE

STOP INPUT BY TYPING:0,0,0

:1,:2,:3,
:1,:3,:4,
:1,:4,:5,
:1,:5,:7,
:2,:3,:6,
:2,:4,:7,
:3,:5,:2,
:0,:0,:0,
DONE

THE M.S.T. CONSISTS OF THE FOLLOWING ARCS:

ARC(=   1,=   2)
ARC(=   1,=   3)
ARC(=   1,=   4)
ARC(=   3,=   5)

DO YOU WISH TO DO ANOTHER? :NO

*
```

Return on Investment

ABSTRACT

Return on Investment **(or "internal rate of return") is defined as the interest rate that causes "the present value of the expected future receipts" to be equal to the "present value of the investment outlay." The desired equality can also be called "discounted cash flow back" to be equal to "the present value of capital employed."†**

Operational Procedures

1. Type *GO* and answer the following:**

a. Size of periods (e.g., 0.25)
b. Number of years
c. Amount to be depreciated
d. Immediate expense (tax deductable)
e. Additional working capital (e.g., inventory).

2. A period by period estimate of savings (or income) of expenses follows. A negative number placed in the SAVE (T) column will cause the expense savings of the previous year to be repeated automatically for the remainder of the periods.

3. Assume the following (all assumptions may be changed in the example program):

a. Tax rate is taken as 57 percent per year (line 12.5 in the sample program).

b. Declining balance depreciation is used (something on the order of straight-line depreciation when it becomes faster; lines 5.1 and 5.2).

c. In computing present value, a discount factor is computed assuming daily compound interest and distributed receipts (line 5.4).

d. Annual compound interest may be substituted by (5.4 SET DI = 1/ (1+K) T).

---

†W. Brigham, Managerial Finance, Harcourt, Brace & World, 1969; p. 148.

```
WRITE ALL
C-FOCAL,1969
02.20 A "SIZE OF PERIODS,YRS.";E
02.30 A Z,"  NUMBER OF YEARS"N
02.40 A !"AMOUNT TO BE DEPRECIATED"A
02.50 S N=N/Z;S R=.57*Z
02.55 A !"IMMEDIATE EXPENSES"E,!""WORKING CAPITAL
02.60 A WC,!"   T   SAVE(T)    EXPENSE(T)
02.70 F T=1,N;D 3
02.80 D 4;R

03.20 I <ST>3.3;T !%4.02,T,"   ";A S(T);I <-S(T)>3.4,3.4;S ST=-1
03.30 S S(T)=S(T-1);S E(T)=E(T-1);R
03.40 A "        " E(T)

04.10 SET K=.25
04.20 SET BA=A;SET Y=A+WC+E*(1-R);SET X=0;FOR T=1,N;DO 5
04.30 SET Z=FABS(X/Y)
04.40 IF <FABS(Z-1)*K-.0001>4.8;SET K=K*Z;GOTO 4.2
04.80 T !!%6.02"    R.O.I."K*100," %
04.91 T !!"                PROFIT(BEFORE)  (AFTER)    CASH
04.93 T !"   PERIOD  DEPREC.   TAXES      TAXES      FLOW   FACTOR
04.94 T "   VALUE"!
04.95 S BA=A;F T=0,N;S X=0;D 5;D 6

05.10 S DE=FEXP(-2*T/N)
05.20 IF <DE-(1-T/N)>5.3;S DE=1-T/N
05.30 S DE=BA-A*DE;S BA=BA-DE
05.40 S DI=FEXP(K/2-K*T)
05.50 S X=X+[S(T)-E(T)]*(1-R)*DI
05.60 S Y=Y-DE*R*DI
05.70 S Z=S(T)-E(T);S Y=Y+Z*(FSGN(Z)-1)/(-2)*1.25↑T
06.10 S Z=X/(1-R)*DI
06.20 T T,DE,Z,(1-R)*Z,X/DI+DE,%6.04,DI,%6.02,X+DE*DI,!
*

*GO
SIZE OF PERIODS,YRS.:.25   NUMBER OF YEARS:2
AMOUNT TO BE DEPRECIATED:20000
IMMEDIATE EXPENSES:1500
WORKING CAPITAL:223
   T   SAVE(T)    EXPENSE(T)
=   1.00   :5000B←5000         :800
=   2.00   :6000          :900
=   3.00   :6500          :1000
=   4.00   :-1

     R.O.I.=    16.35 %

               PROFIT(BEFORE)  (AFTER)    CASH
   PERIOD  DEPREC.   TAXES      TAXES      FLOW     FACTOR    VALUE
=     0.00=     0.00=-1500.00=-1286.25=-1286.25=  1.0852=-1395.81
=     1.00= 4423.99= 4200.00= 3601.50= 8025.49=  0.9215= 7395.55
=     2.00= 3445.41= 5100.00= 4373.25= 7818.66=  0.7825= 6118.27
=     3.00= 2683.27= 5500.00= 4716.25= 7399.52=  0.6645= 4916.97
=     4.00= 2089.75= 5500.00= 4716.25= 6806.00=  0.5643= 3840.46
=     5.00= 1627.49= 5500.00= 4716.25= 6343.74=  0.4792= 3039.73
=     6.00= 1267.49= 5500.00= 4716.25= 5983.74=  0.4069= 2434.78
=     7.00= 1962.61= 5500.00= 4716.25= 6678.85=  0.3455= 2307.74
=     8.00= 2500.00= 5500.00= 4716.25= 7216.25=  0.2934= 2117.36
*
```

Schroedinger

ABSTRACT

By inputting the width of the tilted square well, the tilt slope of the well, the trial energy, and the number of steps, the equation

PSI + AX * PSI = E * PSI

is calculated and the plot is dumped on the Teletype for the specified number of steps.

Operational Procedures

1. Load *Schroedinger* via FOCAL-8 with the extended functions.***

2. Type *GO* and respond to the dialogue. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

Ø1.Ø2 T !,"SCHROEDINGER:DELSQUARED PSI + AX * PSI = E * PSI",!!
Ø1.Ø3 A "TILTED SQUARE WELL PROBLEM WITH WIDTH",XØ,!
Ø1.1Ø A "WELL TILT SLOPE A",A1,!,"TRIAL ENERGY E",B1
Ø1.11 A !"NUMBER OF STEPS",NT,!; S VF=Ø; S SL=1
Ø1.7Ø S P(Ø)=Ø;S DX=XØ/NT;S P(1)=SL*DX;S RØ=Ø
Ø1.75 S VF=Ø
Ø1.8Ø S PØ=Ø
Ø1.9Ø F N=Ø,NT-2;D 6
Ø1.93 T !"PSI ZEROS"%2,PØ
Ø1.95 G Ø7.Ø2

Ø5.1Ø T !,%3,PX," PSI",%,P(PX),"."
Ø5.2Ø S PZ=FITR(PM*SC);S PE=FITR((P(PX)+PM)*SC)
Ø5.3Ø F X=1,PZ;T " "
Ø5.4Ø T "."#;F X=1,PE+24;T " "
Ø5.5Ø T "*";R

Ø6.1Ø S P(N+2)=<(-B1+A1*DX*(N+1))*DX↑2+2>*P(N+1)-P(N)
Ø6.2Ø I (NT-N-2) 12.9Ø,6.9,6.3
Ø6.3Ø S RB=P(N+2)*P(N+1);I (-RB)6.9;
Ø6.4Ø S PØ=PØ+1;R
Ø6.9Ø C

Ø7.Ø2 S CF=(P(NT)/P(1))↑2;T "  CONV IND"%,CF
Ø7.Ø5 A "  NEW E?"NY
Ø7.Ø7 I (NY-ØYES) 7.9,7.Ø8,7.9
Ø7.Ø8 I (VF) 7.Ø9,7.8;
Ø7.Ø9 I (CF-1ØØ) 7.1,7.1,7.8
Ø7.1Ø S R2=P(NT)*VF;I (R2) 7.73,7.8Ø,7.85
Ø7.73 S DB=-Ø.5*DB;G Ø7.85
Ø7.8Ø S DB=Ø.1
Ø7.85 S B1=B1*(1+DB);T B1;S VF=P(NT);G Ø1.8Ø
Ø7.9Ø D 14

12.Ø1 T !,!,"EIGEN E"B1;S HP=B1/(A1*XØ)
12.2Ø T "  EN/MAX POT"HP,!
12.9Ø Q

14.1Ø S PM=Ø;S PP=Ø;F PX=1,NT;D 15
14.2Ø S PS=PM+PP;S SC=45/PS
14.3Ø T !!!;F PX=1,7Ø;T "."
14.4Ø F PX=Ø,1,NT;D 5
14.5Ø T !;F PX=1,7Ø;T "."
14.6Ø T !!;R

15.1Ø I (P(PX)) 15.2,15.9,15.5
15.2Ø I (PM+P(PX)) 15.3,15.4,15.4
15.3Ø S PM=FABS(P(PX))
15.4Ø R
15.5Ø I (P(PX)-PP) 15.9,15.9,15.6
15.6Ø S PP=P(PX)
15.9Ø R
*
```

```
*GO

SCHROEDINGER:DELSQUARED PSI + AX * PSI = E * PSI

TILTED SQUARE WELL PROBLEM WITH WIDTH:1
WELL TILT SLOPE A:.05
TRIAL ENERGY E:1
NUMBER OF STEPS:20

PSI ZEROS=  0  CONV IND= 0.285971E+03  NEW E?:YES = 0.110000E+01
PSI ZEROS=  0  CONV IND= 0.275920E+03  NEW E?:YES = 0.121000E+01
PSI ZEROS=  0  CONV IND= 0.265180E+03  NEW E?:NO


..........................................................................
=   0 PSI= 0.000000E+00.*.
=   1 PSI= 0.500000E-01. .*
=   2 PSI= 0.998491E-01. .     *
=   3 PSI= 0.149397E+00. .          *
=   4 PSI= 0.198496E+00. .             *
=   5 PSI= 0.247000E+00. .                  *
=   6 PSI= 0.294764E+00. .                       *
=   7 PSI= 0.341647E+00. .                          *
=   8 PSI= 0.387512E+00. .                               *
=   9 PSI= 0.432223E+00. .                                  *
=  10 PSI= 0.475652E+00. .                                       *
=  11 PSI= 0.517671E+00. .                                          *
=  12 PSI= 0.558159E+00. .                                             *
=  13 PSI= 0.597001E+00. .                                                *
=  14 PSI= 0.634085E+00. .                                                     *
=  15 PSI= 0.669307E+00. .                                                      *
=  16 PSI= 0.702566E+00. .                                                         *
=  17 PSI= 0.733770E+00. .                                                            *
=  18 PSI= 0.762832E+00. .                                                               *
=  19 PSI= 0.789672E+00. .                                                                 *
=  20 PSI= 0.814217E+00. .                                                                    *
..........................................................................


EIGEN E= 0.121000E+01  EN/MAX POT= 0.242000E+02
*
```

Stock Market Commissions

ABSTRACT

During a stock purchase through a broker, a commission is charged based on a series of rates for units of 100 shares (even lots) and a definite set of charges for smaller units (odd lots).

This program accepts a *buy* or a *sell* indication, the number, and price of the shares involved. Given these facts, the program computes the net you must *pay* or *receive.*

Operational Procedures

1. Load *Stock Market Commissions* by FOCAL-8.*

2. Type *GO* and respond to the dialogue. Execution begins.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.05 E
01.10 A !!"**** BUY OR SELL?"OR
01.20 A !"HOW MANY " ?SHARES  PRICE ?,!
01.40 T %8.02,?PRICE*SHARES?," $"
01.45 S ODD=SHARES-FITR(SHARES/100)*100
01.50 I (-OD) 2.05;
01.55 T !!"ROUND LOTS
01.60 S AM=PRICE*SHARES
01.70 I (AM-400) 1.73 ;I (AM-2400) 1.75;I (AM-5000) 1.77;C
01.71 S CO=AM*.001+39;G 1.8
01.73 S CO=AM*.020+ 3;G 1.8
01.75 S CO=AM*.010+ 7;G 1.8
01.77 S CO=AM*.005+19;G 1.8
01.80 T "COMMISSION IS "CO,!
01.85 I (FABS(OR-0BUY)),1.86;S NET=QU+AM-OC-CO; T "INCOME ";G 1.87
01.86 SET NET=QU+AM+OC+CO ;T "OUTGO
01.87 IF (CO+OC-6) 1.9; IF (<OC+CO>/<OD+SH>-1.50) 1.88,1.9,1.9
01.88 T "IS ",?NET ?,"  $",! ; GO
01.90 A "EXCEPTIONAL COMMISSION " CO;G 1.85

02.05 T !!"ODD LOTS
02.10 SET BROKER=.125; IF (PRICE-55) 2.2; SET BR=.250
02.20 S SH=SH-ODDS
02.30 S QU=OD*PR
02.40 IF (QU-400)2.47;IF (QU-2400)2.45;IF (QU-5000)2.43
02.41 S CO=QU*.001+37;G 2.8
02.43 S CO=QU*.005+17;G 2.8
02.45 S CO=QU*.010+5;G 2.8
02.47 S CO=QU*.020+1;G 2.8
02.80 T "COMMISSION ON "%3,OD," ODD SHARES IS "%7.02,CO+BR*OD
02.90 S OC=CO+BR*ODDS ; IF (OF-0BUY) 3.1, 2.9 , 3.1
02.91 T !" OUTGO", OC+QU,!
02.93 IF (SH)E,0,1.55

03.10 T !" INCOME "QU-OC,!
03.20 GOTO 2.93
*

*GO
**** BUY OR SELL?:BUY

HOW MANY SHARES :120  PRICE :22.50

PRICE*SHARES=   2700.00 $

ODD LOTS COMMISSION ON =  20 ODD SHARES IS =     12.00
 INCOME =   438.00

ROUND LOTS COMMISSION IS =     29.50
OUTGO IS NET =  2741.50  $

**** BUY OR SELL?:SELL
HOW MANY SHARES :20  PRICE :74
PRICE*SHARES=   1480.00 $

ODD LOTS COMMISSION ON =  20 ODD SHARES IS =     24.80
 INCOME =  1455.20
```

Two Process Job Simulation

ABSTRACT

This FOCAL-8 program facilitates scheduling jobs that involve two processes. The following data must be stated by the user:

a. **Number of jobs**
b. **Length of time for each of the two processes for each of the jobs.**

After comparing the data, FOCAL-8 will output on the Teletype the optimal production schedule. The jobs are listed according to performance.

Load Procedures

1. Load *Two Process Job Simulation* by FOCAL-8.*

2. Type *GO* and execution begins immediately.

3. A sample run follows.

```
*WRITE ALL
C-FOCAL,1969

01.10 ASK !"TWO PROCESS JOB SHOP SIMULATION"!!
01.11 ASK !"PLEASE ENTER NUMBER OF JOBS TO BE DONE"!, M
01.12 ASK !"PLEASE ENTER TIMES FOR EACH JOB ON PROCESS ONE AND"!
01.13 ASK "TWO, SEPARATED BY COMMAS"!

02.01 FOR I=1,M;TYPE !" JOB ",%2,I ; ASK A(I), B(I)

03.01 SET IL=M; SET I1=1; SET ID=0
03.02 SET SM=1E6; FOR I=1,M; DO 4
03.03 GOTO 5.1

04.01 IF (A(I)-B(I))4.06; IF (SM-B(I))4.09; SET JO=2;GOTO 4.08
04.06 IF (SM-A(I))4.09; SET JO=1
04.08 SET IO=I; SET SM=A(I); IF (JO-2)4.09; SET SM=B(I)
04.09 RETURN

05.10 IF (JO-2)5.11; SET IS(IL)=IO; SET IL=IL-1; GOTO 5.12
05.11 SET IS(I1)=IO;SET I1=I1+1
05.12 SET A(IO)=1E6;SET B(IO)=1E7; SET ID=ID+1
05.13 IF (ID-M)3.02,5.14
05.14 TYPE !"THE OPTIMAL PRODUCTION SCHEDULE IS"!
05.15 FOR I=1,M; TYPE ! , IS(I)
05.16 ASK !!?ANOTHER ?; IF (AN-0NO)2.01,5.2,2.01
05.20 TYPE !"GLAD TO BE OF HELP."!; QUIT
*


*GO

TWO PROCESS JOB SHOP SIMULATION


PLEASE ENTER NUMBER OF JOBS TO BE DONE
:3

PLEASE ENTER TIMES FOR EACH JOB ON PROCESS ONE AND
TWO, SEPARATED BY COMMAS

 JOB =   1:5,:3,
 JOB =   2:6,:5,
 JOB =   3:1,:1,
THE OPTIMAL PRODUCTION SCHEDULE IS

=  2
=  1
=  3

ANOTHER :NO

GLAD TO BE OF HELP.
*
```

HOW TO OBTAIN SOFTWARE INFORMATION

Announcements for new and revised software, as well as programming notes, software problems, and documentation corrections are published by Software Information Service in the following newsletters.

Digital Software News for the PDP-8 Family
Digital Software News for the PDP-9/15 Family
PDP-6/PDP-10 Software Bulletin

These newsletters contain information applicable to software available from Digital's Program Library.

Please complete the card below to place your name on the newsletter mailing list.

Questions or problems concerning DEC Software should be reported to the Software Specialist at your nearest DEC regional or district sales office. In cases where no Software Specialist is available, please send a Software Trouble Report form with details of the problem to:

Software Information Service
Digital Equipment Corporation
146 Main Street, Bldg. 1-2
Maynard, Massachusetts 01754

These forms, which are available without charge from the Program Library, should be fully filled out and accompanied by teletype output as well as listings or tapes of the user program to facilitate a complete investigation. An answer will be sent to the individual and appropriate topics of general interest will be printed in the newsletter.

New and revised software and manuals, Software Trouble Report forms, and cumulative Software Manual Updates are available from the Program Library. When ordering, include the document number and a brief description of the program or manual requested. Revisions of programs and documents will be announced in the newsletters and a price list will be included twice yearly. Direct all inquiries and requests to:

Program Library
Digital Equipment Corporation
146 Main Street, Bldg. 1-2
Maynard, Massachusetts 01754

Digital Equipment Computer Users Society (DECUS) maintains a user Library and publishes a catalog of programs as well as the DECUSCOPE magazine for its members and non-members who request it. For further information please write to:

DECUS
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

Send Digital's software newsletters to:

Name ______________________________

Company Name ______________________________

Address ______________________________ (zip code) __________

My computer is a

PDP-8/I	☐	PDP-8/L	☐
LINC-8	☐	PDP-12	☐
PDP-9	☐	PDP-15	☐
PDP-10	☐	OTHER	☐ ________ Please specify

My system serial number is ______________ (if known)

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

READER'S COMMENTS

Digital Equipment Corporation maintains a continuous effort to improve the quality and usefulness of its publications. To do this effectively we need user feedback – your critical evaluation of this manual.

Please comment on this manual's completeness, accuracy, organization, usability, and readability.

____

____

____

Did you find errors in this manual? ________________________

____

____

____

How can this manual be improved? ________________________

____

____

____

DEC also strives to keep its customers informed of current DEC software and publications. Thus, the following periodically distributed publications are available upon request. Please check the appropriate boxes for a current issue of the publication(s) desired.

- ☐ **Software Manual Update**, a quarterly collection of revisions to current software manuals.
- ☐ **User's Bookshelf**, a bibliography of current software manuals.
- ☐ **Program Library Price List**, a list of currently available software programs and manuals.

Please describe your position. ________________________

Name __________________ Organization __________________

Street __________________ Department __________________

City ____________ State ____________ Zip or Country ____________

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 33
MAYNARD, MASS.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Digital Equipment Corporation
Software Information Services
146 Main Street, Bldg. 3-5
Maynard, Massachusetts 01754

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS

APPLICATIONS

Computer aided instruction
Trigonometric problems
Numbers systems and base conversion
Geometric design
Factor analysis
Binomial expansions
Statistics
Single- or multi-function plotting
Linear algebra (simultaneous equations, matrices etc.)
Differential equations
Table generation
Simulation
Least squares fit
Eigenvalues
Digital filter design
Closed circuit traverse analysis
Measurement units conversion
Data monitoring
Compound interest
Stocks and bonds analysis
Decision theory
Budgeting
Scope and plotter output, and A/D input
Management games
Many others