FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '85
JUDGING CRITERIA

1.1 INPUT: Enter command: **ADD**
INPUT: Enter number: **18**
INPUT: Enter command: **TAKE**
OUTPUT: **18**
INPUT: Enter command: **ADD**
INPUT: Enter number: **5**
INPUT: Enter command: **ADD**
INPUT: Enter number: **99**
INPUT: Enter command: **TAKE**
OUTPUT: **99**
INPUT: Enter command: **ADD**
INPUT: Enter number: **34**
INPUT: Enter command: **TAKE**
OUTPUT: **34**
INPUT: Enter command: **TAKE**
OUTPUT: **5**
INPUT: Enter command: **QUIT**
OUTPUT: (program ends)

1.2 INPUT: Enter N, AV: **28, 15**
OUTPUT: **NUMBER ERASED WAS 1**

INPUT: Enter N, AV: **31, 16.2**
OUTPUT: **NUMBER ERASED WAS 10**

INPUT: Enter N, AV: **101, 50.68**
OUTPUT: **NUMBER ERASED WAS 83**

1.3 INPUT: Enter N, D: **18, -4**
OUTPUT: **S= 4.2426**
SUM=18

INPUT: Enter N, D: **1562500, 2**
OUTPUT: **S=1300.000**
SUM= 4

INPUT: Enter N, D: **1194, -1**
OUTPUT: **S= 34.6000**
SUM=13

1.4 RUN PROGRAM:
OUTPUT: A time dial simulation will be displayed in the center of the screen. The count starts with 1985, and then steadily increases the years by 1 until the year 2345 is reached: The time interval between each new year will decrease as the numbers increase, starting with one new year every second, to steadily and rapidly counting faster than the eye can comprehend, ending with the year 2345, which remains on the screen. The displays should not flicker. The program should take less than 60 seconds to run.

1.5 INPUT: Enter N: **18**
OUTPUT: **ROUND 1 9 GAMES**
ROUND 2 4 GAMES 1 BYE
ROUND 3 2 GAMES 1 BYE
ROUND 4 1 GAMES 1 BYE
ROUND 5 1 GAMES
TOTAL 17 GAMES 3 BYES

INPUT: Enter N: **67**
OUTPUT: **ROUND 1 33 GAMES 1 BYE**
ROUND 2 17 GAMES
ROUND 3 8 GAMES 1 BYE
ROUND 4 4 GAMES 1 BYE
ROUND 5 2 GAMES 1 BYE
ROUND 6 1 GAMES 1 BYE
ROUND 7 1 GAMES
TOTAL 66 GAMES 5 BYES

1.6 INPUT: Enter N, M: **900, 1300**
OUTPUT: **SMALLEST = 912**
LARGEST = 987
SUM = 53172

INPUT: Enter N, M: **33, 333**
OUTPUT: **SMALLEST = 123**
LARGEST = 329
SUM = 27990

1.7 INPUT: Enter name: **DOUG**
Enter part#: **T100**
Enter time: **3**

OUTPUT: **CUSTOMER NAME: DOUG**
PART #: T100
DESCRIPTION: 27X1 INCH TIRE TUBE
PART COST: 12.50
LABOR COST: 30.00
5% TAX: 0.63
TOTAL: 43.13

INPUT: Enter name: **BRAD**
Enter part#: **S445**
Enter time: **2.5**

OUTPUT: **CUSTOMER NAME: BRAD**
PART #: S445
DESCRIPTION: COMPUCYCLE COMPUTER
PART COST: 33.95
LABOR COST: 25.00
5% TAX: 1.70
TOTAL: 60.65

1.8 INPUT: Enter # of lines on label: **4**
OUTPUT:

PARKER, HARRY
222-3333

SIMON, BILL
123-4567

SIMON, BOB
123-4455

SPINXS, LISA
987-6543

TROUTMAN, HARRY
876-2174

1.9 RUN PROGRAM: A 5x5 matrix of 25 letters, A through Y, is randomly generated and centered on the top part of the screen, with every adjacent letter on a row separated by a space.

SECRETLY CHOOSE THE LETTER 'Y' AND NOTE ITS POSITION IN THE ARRAY.

The computer must ask the user yes(Y)-or-no(N) questions to logically determine the secret letter (using similar questions as shown in the example). The computer will start with a score of 11 points and will deduct 1 point for each question that is asked and answered. The score is displayed in the upper right corner after each question is asked. If the program does not determine the letter before the computer score reaches 0, then no credit is awarded at this time. If the letter Y is guessed and the score is greater than 0, then run this program one more time and ensure that the new matrix is different from the previous matrix:

SECRETLY CHOOSE THE LETTER 'P' AND NOTE ITS POSITION IN THE ARRAY.

```
OUTPUT:          Q W E R T          SCORE=11
                 Y U I O P
                 A S D F G
                 H J K L M
                 X C V B N

 OUTPUT/INPUT: IS THE LETTER IN ROW 1? N
 OUTPUT: (The score decreases to 10 at the top right)
 OUTPUT/INPUT: IS THE LETTER IN ROW 2? Y
 OUTPUT: (The score decreases to 9)
 OUTPUT/INPUT: IS THE LETTER IN COL 1? Y
 OUTPUT: (The score decreases to 8)
         YOUR LETTER IS Y
```

1.10 RUN PROGRAM: Press the appropriate keys I,J,K,M to place the cursor in the center of the screen.
INPUT: **1**
OUTPUT: The box below with respect to the cursor (#):

```
          #
          *********
          *       *
          *   1   *
          *       *
          *********
```

RUN PROGRAM: Place the cursor in the center of the screen.
INPUT: **2**
OUTPUT:

```
                    #
          *********
          *       *
          *   2   *
          *       *
          *********
```

RUN PROGRAM: Place the cursor in the center of the screen.
INPUT: **3**
OUTPUT:

```
          *********
          *       *
          *   3   *
          *       *
          *********
                    #
```

RUN PROGRAM: Place the cursor in the center of the screen.
INPUT: **4**
OUTPUT:

```
          *********
          *       *
          *   4   *
          *       *
          *********
          #
```

RUN PROGRAM: Place the cursor at the absolute left side of the screen
INPUT: **2**
OUTPUT: **OFF THE SCREEN**

RUN PROGRAM: Place the cursor at the top of the screen.
INPUT: **3**
OUTPUT: **OFF THE SCREEN**

2.1 RUN PROGRAM:

OUTPUT: A random letter outlines the border of the screen, then upon pressing the space bar, the inside border of the new screen will be outlined by a random letter; afterwards, when the space bar is pressed, the inside border of the new screen will be outlined by a random letter, and so on. These rectangles are drawn until the whole screen is filled, then press the space bar once again and the screen will clear and start over with a new outer border. A miniature sample run would look like this:

```
RRRRRRRRRR          RRRRRRRRRR          RRRRRRRRRR
R        R          RQQQQQQQQR          RQQQQQQQQR
R        R          RQ      QR          RQYYYYYYQR
R        R          RQ      QR          RQYYYYYYQR
R        R          RQQQQQQQQR          RQQQQQQQQR
RRRRRRRRRR          RRRRRRRRRR          RRRRRRRRRR
```

2.2 INPUT: Enter N: **10**
Enter letter: **G**
Enter letter: **H**
Enter letter: **L**
Enter letter: **L**
Enter letter: **R**
Enter letter: **S**
Enter letter: **S**
Enter letter: **Q**
Enter letter: **B**
Enter letter: **A**
OUTPUT: **G H L**
L R S

INPUT: Enter N: **15**
Enter letter: **Z**
Enter letter: **Z**
Enter letter: **A**
Enter letter: **C**
Enter letter: **B**
Enter letter: **G**
Enter letter: **P**
Enter letter: **Q**
Enter letter: **Y**
Enter letter: **T**
Enter letter: **W**
Enter letter: **E**
Enter letter: **F**
Enter letter: **M**
Enter letter: **X**
OUTPUT: **B G P Q Y**

2.3 INPUT: Enter text: **PROVIDE A 5 CHARACTER LEFT MARGIN. DO NOT PUT MORE THAN 30 CHARACTERS ON A LINE. THE LAST WORD IS FOLLOWED BY A PERIOD.**

OUTPUT:
```
     PROVIDE A 5 CHARACTER
LEFT MARGIN.  DO NOT PUT MORE
THAN 30 CHARACTERS ON A LINE.
THE LAST WORD IS FOLLOWED BY A
PERIOD.
```

INPUT: Enter text: **A WORD IS DEFINED AS A SET OF CHARACTERS IN BETWEEN TWO SPACES (EXCEPT FOR THE FIRST AND LAST WORDS OF THE STRING).**

OUTPUT:
```
     A WORD IS DEFINED AS A
SET OF CHARACTERS IN BETWEEN
TWO SPACES (EXCEPT FOR THE
FIRST AND LAST WORDS OF THE
STRING).
```

2.4 INPUT: Enter word: **INTERNATIONAL**
OUTPUT: **ALNANNERIITOT**

INPUT: Enter word: **CLASS**
OUTPUT: **CLASS**

INPUT: Enter word: **UNIVERSITY**
OUTPUT: **ENIRISTUVY**

2.5 INPUT: Enter number of words: **5**
Enter word: **COMPUTER**
Enter word: **APPLE**
Enter word: **PERSONAL**
Enter word: **CREATIVE**
Enter word: **POPULAR**
OUTPUT: **NO COMMON LETTERS**

INPUT: Enter N: **6**
Enter word: **CREATIVE**
Enter word: **ELECTRONIC**
Enter word: **PROCESS**
Enter word: **PEACH**
Enter word: **EDUCATION**
Enter word: **COMPLEX**
OUTPUT: **C E**
INPUT: Choose letter: **E**
OUTPUT:
```
   CREATIVE
     ELECTRONIC
 PROCESS
    PEACH
     EDUCATION
COMPLEX
```

2.6 INPUT: Place 1: **T**
Place 2: **D**
Place 3: **C**
Place 4: **T**
Place 5: **D**
Place 6: **C**
Place 7: **T**
Place 8: **D**
Place 9: **C**
Place 10: **C**
Place 11: **D**
Place 12: **T**
Place 13: **C**
Place 14: **D**
Place 15: **C**
Place 16: **D**
Place 17: **T**
Place 18: **T**
Place 19: **C**
Place 20: **D**
Place 21: **T**

OUTPUT: (in any order)
TEAM T: 28 POINTS
TEAM D: 28 POINTS
TEAM D WINS!

TEAM T: 28 POINTS
TEAM C: 28 POINTS
TEAM C WINS!

TEAM D: 27 POINTS
TEAM C: 28 POINTS
TEAM D WINS!

INPUT: Place 1: **A**
Place 2: **B**
Place 3: **A**
Place 4: **C**
Place 5: **C**
Place 6: **B**
Place 7: **A**
Place 8: **A**
Place 9: **B**
Place 10: **C**
Place 11: **C**
Place 12: **A**
Place 13: **C**
Place 14: **B**
Place 15: **A**
Place 16: **A**
Place 17: **B**
Place 18: **B**
Place 19: **B**
Place 20: **C**
Place 21: **C**

OUTPUT: (in any order)
TEAM A: 23 POINTS
TEAM B: 34 POINTS
TEAM A WINS!

TEAM A: 23 POINTS
TEAM C: 32 POINTS
TEAM A WINS!

TEAM B: 29 POINTS
TEAM C: 26 POINTS
TEAM C WINS!

2.7 RUN PROGRAM:

```
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: B
OUTPUT:  10.11   20.22   30.33   60.66
         11.10   22.20   33.30   66.60
         10.00   20.00   30.00   60.00
         31.21   62.42   93.63  187.26
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: A
        Enter row, col: 2, 1
        Enter number: 5.5
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: B
OUTPUT:  10.11   20.22   30.33   60.66
          5.50   22.20   33.30   61.00
         10.00   20.00   30.00   60.00
         25.61   62.42   93.63  181.66
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: A
        Enter row, col: 1, 3
        Enter number: 29.67
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: A
        Enter row, col: 3, 2
        Enter number: 39
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: B
OUTPUT:  10.11   20.22   29.67   60.00
          5.50   22.20   33.30   61.00
         10.00   39.00   30.00   79.00
         25.61   81.42   92.97  200.00
 INPUT: (press any key)
OUTPUT: A. EDIT OR CHANGE A VALUE
        B. DISPLAY THE RESULTS
        C. QUIT
 INPUT: Enter option: C
OUTPUT: (program terminates)
```

2.8 RUN PROGRAM:

```
OUTPUT: 2  5  1  0    2 X 5 = 10
        2  7  1  4    2 X 7 = 14
        2  8  1  6    2 X 8 = 16
        2  9  1  8    2 X 9 = 18
        3  4  1  2    3 X 4 = 12
        3  6  1  8    3 X 6 = 18
        3  7  2  1    3 X 7 = 21
        3  8  2  4    3 X 8 = 24
        3  9  2  7    3 X 9 = 27
        4  5  2  0    4 X 5 = 20
        4  7  2  8    4 X 7 = 28
        4  8  3  2    4 X 8 = 32
        4  9  3  6    4 X 9 = 36
        5  6  3  0    5 X 6 = 30
        5  8  4  0    5 X 8 = 40
        6  7  4  2    6 X 7 = 42
        6  9  5  4    6 X 9 = 54
        7  8  5  6    7 X 8 = 56
        7  9  6  3    7 X 9 = 63
        8  9  7  2    8 X 9 = 72
        TOTAL = 20
```

2.9

```
 INPUT: Enter N: 11
        Enter word: CREATE
        Enter word: CREATION
        Enter word: CREATIVE
        Enter word: CREATURE
        Enter word: EVERYBODY
        Enter word: EVERYONE
        Enter word: ELECTION
        Enter word: CREDIT
        Enter word: COMPUTER
        Enter word: PRINTER
        Enter word: EMPTY
 INPUT: Enter string: CREAT*
OUTPUT: CREATE
        CREATION
        CREATIVE
        CREATURE
 INPUT: Enter string: *TION
OUTPUT: CREATION
        ELECTION
 INPUT: Enter string: E*Y
OUTPUT: EVERYBODY
        EMPTY
 INPUT: Enter string: *ATER
OUTPUT: NO WORDS FOUND
 INPUT: Enter string: *PRINTER
OUTPUT: PRINTER
 INPUT: Enter string: END
OUTPUT: (program terminates)
```

2.10 INPUT: Enter last 5-minutes: **90**

OUTPUT: (Briefly scan only the messages from 40 MIN to 90 MIN)

OF	CO	DS	OFFICE	COMP.	DRY.	MIN:SE
:	:	:	:	:	:	:
:	:	:	:	:	:	:
1	0	0	72.5	68.0	83.0	40:00
0	0	0	71.7	68.4	83.1	40:30
0	1	0	72.6	70.2	83.3	42:45
0	1	0	73.5	67.5	83.5	45:00
0	0	0	74.4	64.8	83.7	47:15
0	0	0	75.5	67.0	84.0	50:00
0	1	0	77.1	70.2	84.4	54:00
0	1	0	77.5	69.0	84.5	55:00
1	1	0	78.1	67.2	84.7	56:30
1	1	1	76.0	66.5	85.0	60:00
1	1	1	74.7	67.2	82.2	65:00
1	1	1	73.3	67.8	79.3	70:00
0	1	1	72.0	68.5	76.5	75:00
0	1	0	72.7	68.1	74.9	76:45
0	0	0	73.8	64.8	75.2	79:30
0	0	0	74.0	65.2	75.3	80:00
0	0	0	76.0	69.2	75.8	85:00
0	1	0	76.4	70.0	75.9	86:00
0	1	0	78.0	65.2	76.3	90:00

3.1 INPUT: Enter top, front: **1, 3**
OUTPUT: **TOP=1 FRONT=3 RIGHT=5**
BACK=4 LEFT=2 BOTTOM=6

INPUT: Enter top, front: **2, 3**
OUTPUT: **TOP=2 FRONT=3 RIGHT=1**
BACK=4 LEFT=6 BOTTOM=5

INPUT: Enter top, front: **4, 6**
OUTPUT: **TOP=4 FRONT=6 RIGHT=2**
BACK=1 LEFT=5 BOTTOM=3

INPUT: Enter top, front: **6, 2**
OUTPUT: **TOP=6 FRONT=2 RIGHT=4**
BACK=5 LEFT=3 BOTTOM=1

3.2 INPUT: Enter A, B, C: **1, 0, -1**
OUTPUT: **(X-1)(X+1)** or **(X+1)(X-1)**

INPUT: Enter A, B, C: **-6, 7, -2**
OUTPUT: **(3X-2)(2X-1)** or **(2X-1)(3X-2)**

INPUT: Enter A, B, C: **18, 12, 2**
OUTPUT: **2(3X+1)(3X+1)**

INPUT: Enter A, B, C: **1, 2, 3**
OUTPUT: **CANNOT BE FACTORED**

3.3 INPUT: Enter expression: **5/8/100*100** OUTPUT: **0.625**

INPUT: Enter expression: **6-4+5/4*10** OUTPUT: **14.500**

INPUT: Enter expression: **4*1*0/6** OUTPUT: **0.000**

INPUT: Enter expression: **12/3+5-3*6*2+7** OUTPUT: **-20.000**

3.4 INPUT: Enter N: **12**
OUTPUT: **479001600**

INPUT: Enter N: **40**
OUTPUT: **8159152832478977343456112695961158942720000000000**

3.5 INPUT: Enter #1: **5678901234.5**
Enter #2: **45.610987**

OUTPUT: **SUM = 5678901280.110987**
DIFFERENCE = 5678901188.889013

INPUT: Enter #1: **8765432109.8765432109**
Enter #2: **2109.87654321**

OUTPUT: **SUM = 8765434219.7530864209**
DIFFERENCE = 8765430000.0000000009

INPUT: Enter #1: **69.1**
Enter #2: **2.3456**

OUTPUT: **SUM = 71.4456**
DIFFERENCE = 66.7544

3.6 RUN PROGRAM: A snake (a trail of 30 asterisks '*') is centered on the screen. Upon hitting appropriate keys, designated by students, the snake's head moves in the appropriate direction while the rest of the snake slithers along the same right angle paths. The snake must move CONTINUOUSLY in the designated direction UNTIL a new directional key is hit. The snake must be 30 asterisks long throughout the entire run; It must not leave a sketched path. The snake continues moving until it runs into itself or it runs off the screen or a non-directional key is pressed.

Run the program and have the snake move in all directions. Have the snake run into itself to check that the program will STOP. For the next execution, have the snake attempt to leave the screen, which should cause the program to STOP.

3.7 INPUT: Enter word: **LIFE**
Enter K: **5**

OUTPUT: **ELFI FILE IFEL**

INPUT: Enter word: **COMPUTE**
Enter K: **721**

OUTPUT: **ECMOPTU MCEOPUT OCEMTPU**

3.8 Check to see that no two pennies (asterisks) are in the same column, row, or main diagonal: Check that the Row equals the Column at most once; Check that the sum of the coordinates equals N+1 at most once.
NOTE: PLACEMENT OF THE ASTERISKS WILL VARY ALONG WITH THE SUMS.

```
 INPUT: Enter N: 6
OUTPUT: TOTAL = 6
          1 2 3 4 5 6
        1   *           (1,2) SUM = 3
        2       *       (2,4) SUM = 6
        3           *   (3,6) SUM = 9
        4 *             (4,1) SUM = 5
        5     *         (5,3) SUM = 8
        6         *     (6,5) SUM = 11

 INPUT: Enter N: 7
OUTPUT: TOTAL = 7
          1 2 3 4 5 6 7
        1 *               (1,1) SUM = 2
        2     *           (2,3) SUM = 5
        3         *       (3,5) SUM = 8
        4             *   (4,7) SUM = 11
        5   *             (5,2) SUM = 7
        6       *         (6,4) SUM = 10
        7           *     (7,6) SUM = 13

 INPUT: Enter N: 8
OUTPUT: (similar format as first two runs)
        TOTAL = 8
          1 2 3 4 5 6 7 8
          :

 INPUT: Enter N: 14
OUTPUT: (similar format as first two runs)
        TOTAL = 14
          1 2 3 4 5 6 7 8 9 0 1 2 3 4
          :
```

3.9 INPUT: Enter N: **5**
OUTPUT: **31**

INPUT: Enter N: **10**
OUTPUT: **1023**

INPUT: Enter N: **12**
OUTPUT: **4095**

3.10 INPUT: Enter S: **36**
OUTPUT: **P = 15678 Q = 39 R = 402**

INPUT: Enter S: **62**
OUPTUT: **P = 58401 Q = 63 R = 927**

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '86
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: (The screen is cleared and the following is centered)

THIS IS THE EASIEST PROGRAM!

1.2 INPUT: Enter two numbers: **12, -5**

Note: (One or two spaces may separate the equal sign and each of the numbers output)

OUTPUT: **SUM = 7**
DIFFERENCE = 17
PRODUCT = -60

INPUT: Enter two numbers: **-102, 50**

OUTPUT: **SUM = -62**
DIFFERENCE = -152
PRODUCT = -5100

1.3 INPUT: Enter test value E: **0.0001**
OUTPUT: **1.291263**

INPUT: Enter test value E: **0.001**
OUTPUT: **1.290943**

1.4 INPUT: Enter first name: **JANET**
Enter middle name: **CASPERSON**
Enter last name: **SMITH**
Enter amount: **4567.89**

OUTPUT:
```
**************************************
*                                    *
* BEN'S TOWING SERVICE               *
* 4563 WRECKER AVENUE                *
* WAVERLY, ARKANSAS 45632            *
*                                    *
*  PAY TO THE ORDER OF JANET C. SMITH *
*                                    *
*  THE SUM OF  $4567.89              *
*                                    *
**************************************
```

1.5 RUN PROGRAM:

OUTPUT: **CELL 2**
CELL 5
CELL 10
CELL 17
CELL 26
CELL 37
CELL 50
CELL 65
CELL 82

1.6 INPUT: Enter monthly investment: **120**
Enter end of year deposit: **450**
Enter annual rate of interest: **11**

OUTPUT: **AMOUNT AT END OF YEAR 20 IS $135685.95**

INPUT: Enter monthly investment: **50**
Enter end of year deposit: **125**
Enter annual rate of interest: **14**

OUTPUT: **AMOUNT AT END OF YEAR 20 IS $78523.27**

1.7 INPUT: Enter sentence:
PLAYING A BASEBALL GAME INVOLVES NINE INNINGS

OUTPUT: **PLAYIN A BASEBALL GAME INVOLVES NINE INNINS**

INPUT: Enter sentence: **GOING RIDING CAN BE THRILLING**

OUTPUT: **GOIN RIDIN CAN BE THRILLIN**

1.8 INPUT: Enter initial population: **1200**
Enter point of over population: **2600**

OUTPUT: **POPULATION FOR MONTH 1 IS 1440**
POPULATION FOR MONTH 2 IS 1728
POPULATION FOR MONTH 3 IS 2074
POPULATION FOR MONTH 4 IS 2488
POPULATION FOR MONTH 5 IS 2986
POPULATION FOR MONTH 6 IS 2538
POPULATION FOR MONTH 7 IS 2157
POPULATION FOR MONTH 8 IS 1834
POPULATION FOR MONTH 9 IS 2201
POPULATION FOR MONTH 10 IS 2641
POPULATION FOR MONTH 11 IS 2245
POPULATION FOR MONTH 12 IS 1908
POPULATION FOR MONTH 13 IS 1622
POPULATION FOR MONTH 14 IS 1946
POPULATION FOR MONTH 15 IS 2335
POPULATION FOR MONTH 16 IS 2802
POPULATION FOR MONTH 17 IS 2382
POPULATION FOR MONTH 18 IS 2025
POPULATION FOR MONTH 19 IS 1721
POPULATION FOR MONTH 20 IS 2065
POPULATION FOR MONTH 21 IS 2478
POPULATION FOR MONTH 22 IS 2974
POPULATION FOR MONTH 23 IS 2528

1.9 INPUT: Enter sentence: **GEORGE IS A NATIVE OF EELAND.**
OUTPUT: **GEEORGEE IS A NATIVEE OF EELAND.**

1.10 INPUT: Enter 1 of 12: **1**
Enter 2 of 12: **3**
Enter 3 of 12: **66**
Enter 4 of 12: **9**
Enter 5 of 12: **1**
Enter 6 of 12: **9**
Enter 7 of 12: **11**
Enter 8 of 12: **232**
Enter 9 of 12: **6**
Enter 10 of 12: **1**
Enter 11 of 12: **11**
Enter 12 of 12: **12**

Enter 1 of 11: **1**
Enter 2 of 11: **19**
Enter 3 of 11: **32**
Enter 4 of 11: **5**
Enter 5 of 11: **12**
Enter 6 of 11: **99**
Enter 7 of 11: **33**
Enter 8 of 11: **10**
Enter 9 of 11: **66**
Enter 10 of 11: **2**
Enter 11 of 11: **1** OUTPUT: **1 66 12**

2.1 INPUT: Enter sentence: **HERE IS A SHORT SENTENCE.**

OUTPUT: (The sentence is to be right-justified on a 65 column line. Spacing between words is approximately uniform.)

HERE IS A SHORT SENTENCE.

2.2 INPUT: Enter total number of X's and -'s: **7**
Enter number of X's: **3**
Enter number of rows: **14**

OUTPUT:
```
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
XXX----XXX----XXX----XXX----
---XXXX---XXXX---XXXX---XXXX
```

2.3 RUN PROGRAM:

OUTPUT:
1) ENCODE
2) DECODE
3) END

INPUT: Choose: **1**
Enter message: **THIS IS A BIG SECRET**

OUTPUT: **RASE SE Z XSM EBCWBR**

1) ENCODE
2) DECODE
3) END

INPUT: Choose: **2**
Enter message: **RASE SE Z XSM EBCWBR**

OUTPUT: **THIS IS A BIG SECRET**

1) ENCODE
2) DECODE
3) END

INPUT: Choose: **3**

OUTPUT: (program terminates)

2.4

```
INPUT: Enter number 1: 1
       Enter number 2: 4
       Enter number 3: 7
       Enter number 4: 9
       Enter number 5: 4
       Enter number 6: 5
       Enter number 7: 6
       Enter number 8: 7
       Enter number 9: 8
       Enter number 10: 7
       Enter number 11: 12
       Enter number 12: 11
       Enter number 13: 13
       Enter number 14: 33
       Enter number 15: 1

OUTPUT: MODE IS 7
```

```
INPUT: Enter number 1: 1
       Enter number 2: 2
       Enter number 3: 3
       Enter number 4: 4
       Enter number 5: 5
       Enter number 6: 6
       Enter number 7: 7
       Enter number 8: 8
       Enter number 9: 9
       Enter number 10: 1
       Enter number 11: 2
       Enter number 12: 3
       Enter number 13: 4
       Enter number 14: 5
       Enter number 15: 6

OUTPUT: NO UNIQUE MODE
```

```
2.5  INPUT: Enter original balance: 2345.15
    OUTPUT: 1. MAKE A DEPOSIT
            2. MAKE A WITHDRAWAL
            3. CREDIT INTEREST
            4. END
     INPUT: Enter option: 1

            Enter amount to deposit: 100
    OUTPUT: BALANCE BEFORE TRANSACTION $2,345.15
            MAKE A DEPOSIT
            NEW BALANCE $2,445.15

            1. MAKE A DEPOSIT
            2. MAKE A WITHDRAWAL
            3. CREDIT INTEREST
            4. END
     INPUT: Enter option: 2

            Enter amount to withdraw: 50
    OUTPUT: BALANCE BEFORE TRANSACTION $2,445.15
            MAKE A WITHDRAWAL
            NEW BALANCE $2,395.15

            1. MAKE A DEPOSIT
            2. MAKE A WITHDRAWAL
            3. CREDIT INTEREST
            4. END
     INPUT: Enter option: 3

    OUTPUT: BALANCE BEFORE TRANSACTION $2,395.15
            CREDIT INTEREST OF $ 13.97
            NEW BALANCE $2,409.12

            1. MAKE A DEPOSIT
            2. MAKE A WITHDRAWAL
            3. CREDIT INTEREST
            4. END
     INPUT: Enter option: 1

            Enter amount to deposit: 600
    OUTPUT: BALANCE BEFORE TRANSACTION $2,409.12
            MAKE A DEPOSIT
            NEW BALANCE $3,009.12

            1. MAKE A DEPOSIT
            2. MAKE A WITHDRAWAL
            3. CREDIT INTEREST
            4. END
     INPUT: Enter option: 4

    OUTPUT: FINAL BALANCE $3,009.12
```

2.6 INPUT: ENTER FIRST NUMBER: **23765879734265436854**
ENTER SECOND NUMBER: **65487904235412345876**

OUTPUT: **SUM IS 89253783969677782730**

INPUT: ENTER FIRST NUMBER: **912345678901234567890123456789012345**
ENTER SECOND NUMBER: **923443210123454321012345432101234567**

OUTPUT: **SUM IS 1004689999913579999999135799999991357900**

2.7 RUN PROGRAM:

```
OUTPUT:  1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
         2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
         3 METERS TO FEET           9 KILOGRAMS TO POUNDS
         4 FEET TO METERS          10 POUNDS TO KILOGRAMS
         5 KILOMETERS TO MILES     11 LITERS TO GALLONS
         6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                   13 END
 INPUT: Enter option: 6
        Enter number of MILES: 130

OUTPUT: THIS IS EQUIVALENT TO 209.209 KILOMETERS

OUTPUT:  1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
         2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
         3 METERS TO FEET           9 KILOGRAMS TO POUNDS
         4 FEET TO METERS          10 POUNDS TO KILOGRAMS
         5 KILOMETERS TO MILES     11 LITERS TO GALLONS
         6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                   13 END
 INPUT: Enter option: 5
        Enter number of KILOMETERS: 209.209

 OUTPUT: THIS IS EQUIVALENT TO 130.000 MILES

OUTPUT:  1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
         2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
         3 METERS TO FEET           9 KILOGRAMS TO POUNDS
         4 FEET TO METERS          10 POUNDS TO KILOGRAMS
         5 KILOMETERS TO MILES     11 LITERS TO GALLONS
         6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                   13 END
 INPUT: Enter option: 7
        Enter number of GRAMS: 64

OUTPUT: THIS IS EQUIVALENT TO   2.257 OUNCES
```

(Output for 2.7 Continued)

```
    OUTPUT:  1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
             2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
             3 METERS TO FEET           9 KILOGRAMS TO POUNDS
             4 FEET TO METERS          10 POUNDS TO KILOGRAMS
             5 KILOMETERS TO MILES     11 LITERS TO GALLONS
             6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                       13 END
     INPUT: Enter option: 8
            Enter number of OUNCES: 2.257496

     OUTPUT: THIS IS EQUIVALENT TO 64.0 GRAMS

     OUTPUT: 1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
             2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
             3 METERS TO FEET           9 KILOGRAMS TO POUNDS
             4 FEET TO METERS          10 POUNDS TO KILOGRAMS
             5 KILOMETERS TO MILES     11 LITERS TO GALLONS
             6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                       13 END
     INPUT: Enter option: 11
            Enter number of LITERS: 3.7

     OUTPUT: THIS IS EQUIVALENT TO   0.977 GALLONS

     OUTPUT: 1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
             2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
             3 METERS TO FEET           9 KILOGRAMS TO POUNDS
             4 FEET TO METERS          10 POUNDS TO KILOGRAMS
             5 KILOMETERS TO MILES     11 LITERS TO GALLONS
             6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                       13 END

     INPUT: Enter option: 12
            Enter number of GALLONS: 0.9774396

     OUTPUT: THIS IS EQUIVALENT TO   3.700 LITERS

     OUTPUT: 1 CENTIMETERS TO INCHES    7 GRAMS TO OUNCES
             2 INCHES TO CENTIMETERS    8 OUNCES TO GRAMS
             3 METERS TO FEET           9 KILOGRAMS TO POUNDS
             4 FEET TO METERS          10 POUNDS TO KILOGRAMS
             5 KILOMETERS TO MILES     11 LITERS TO GALLONS
             6 MILES TO KILOMETERS     12 GALLONS TO LITERS
                                       13 END
     INPUT: Enter option: 13

    OUTPUT: (program terminates)
```

2.8 INPUT: Enter principal: **12000**
Enter % rate of interest: **13**
Enter term in years: **3**
Enter # of month in year for first payment: **10**

OUTPUT:

INTEREST	**PRINCIPAL**
$130.00	**$11725.67**
$127.03	**$11448.37**
$124.02	**$11168.07**
YEAR'S INTEREST	**$ 381.05**
$120.99	**$10884.73**
$117.92	**$10598.32**
$114.82	**$10308.81**
$111.68	**$10016.16**
$108.51	**$ 9720.34**
$105.30	**$ 9421.32**
$102.06	**$ 9119.05**
$ 98.79	**$ 8813.52**
$ 95.48	**$ 8504.67**
$ 92.13	**$ 8192.47**
$ 88.75	**$ 7876.90**
$ 85.33	**$ 7557.90**
YEAR'S INTEREST	**$ 1241.76**
$ 81.88	**$ 7235.45**
$ 78.38	**$ 6909.51**
$ 74.85	**$ 6580.04**
$ 71.28	**$ 6246.99**
$ 67.68	**$ 5910.34**
$ 64.03	**$ 5570.04**
$ 60.34	**$ 5226.06**
$ 56.62	**$ 4878.35**
$ 52.85	**$ 4526.87**
$ 49.04	**$ 4171.58**
$ 45.19	**$ 3812.45**
$ 41.30	**$ 3449.42**
YEAR'S INTEREST	**$ 743.44**
$ 37.37	**$ 3082.46**
$ 33.39	**$ 2711.53**
$ 29.38	**$ 2336.57**
$ 25.31	**$ 1957.56**
$ 21.21	**$ 1574.44**
$ 17.06	**$ 1187.17**
$ 12.86	**$ 795.70**
$ 8.62	**$ 399.99**
$ 4.33	**$ 0.00**
YEAR'S INTEREST	**$ 189.53**
TOTAL INTEREST	**$ 2555.79**
MONTHLY PAYMENT	**$ 404.33**

2.9 INPUT: Enter N degrees: **150**

OUTPUT: **PARTIAL SUM = 0.4999578**
ACTUAL SINE = 0.5000000

INPUT: Enter N degrees: **225**

OUTPUT: **PARTIAL SUM = -0.7070960**
ACTUAL SINE = -0.7071068

2.10 INPUT: Enter Roman Numeral: **MCMLXXXVI**
OUTPUT: **ARABIC = 1986**

INPUT: Enter Roman Numeral: **CDXLIX**
OUTPUT: **ARABIC = 449**

3.1 RUN PROGRAM:

OUTPUT: (after a month is displayed with its name approximately centered, press any key to clear the screen and the next month will display):

1986

JANUARY

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

FEBRUARY

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

MARCH

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

APRIL

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

(Output continues on next page)

(Output continued)

MAY

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

JUNE

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

JULY

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

AUGUST

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

SEPTEMBER

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

(Output continues on next page)

(Output continued)

OCTOBER

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

NOVEMBER

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

DECEMBER

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

3.2 INPUT: Enter coefficients A,B,C,D,E,F: **1, -5, 8, 5, -9, 6**
OUTPUT: **ROOT IS -1.15078**

INPUT: Enter coefficients A,B,C,D,E,F: **2, -6, -7, -8, -9, -10**
OUTPUT: **ROOT IS 4.15395**

3.3 INPUT: Enter base A: **14**
Enter base B: **7**
Enter original number: **C3B7B8**

OUTPUT: **C3B78 BASE 14 EQUALS 110051321 BASE 7**

INPUT: Enter base A: **8**
Enter base B: **24**
Enter original number: **76543210**

OUTPUT: **76543210 BASE 8 EQUALS 21CKG8 BASE 24**

3.4 INPUT: Enter SSN: **564783219**
Enter C for charge or P for payment: **C**
Enter amount of transaction: **10**

OUTPUT: **NEW BALANCE IS $2,355.89**

INPUT: Enter SSN: **543876543**
Enter C for charge or P for payment: **P**
Enter amount of transaction: **1234.56**

OUTPUT: **NEW BALANCE IS $1279.74**

INPUT: Enter SSN: **345212342**
Enter C for charge or P for payment: **P**
Enter amount of transaction: **543.21**

OUTPUT: **NEW BALANCE IS $3999.30**

INPUT: Enter SSN: **000000000**

OUTPUT:

SSN	NAME	ADDRESS	BALANCE
873421765	TIM JONES	2387 PALM PLACE NOME ALASKA 77643	$6754.76
345212342	AL BROWN	PO BOX 234 TINSEL TOWN CALIFORNIA 77654	$3999.30
564783219	GAIL HUSTON	543 SOUTH THIRD BIG TOWN TEXAS 88642	$2355.89
543876543	JILL RUPERT	4536 123RD STREET TINY TOWN MAINE 76765	$1279.74
234567890	JOHN SMITH	1234 ANYWHERE LANE EXIST KANSAS 66754	$ 345.78
565656565	KERMIT TEU	1234 LOST LANE WIMPLE WISCONSIN 66543	$ 78.36

3.5 INPUT: Enter first number: **5678.90123456789**
Enter second number: **562.98765**

OUTPUT: **PRODUCT = 3197151.2606314751565585**

INPUT: Enter first number: **987654.321**
Enter second number: **123.4567890123**

OUTPUT: **PRODUCT = 121932631.1247834171483**

3.6 INPUT: Enter number: **23**
OUTPUT: **55 IS A PALINDROME**

INPUT: Enter number: **187**
OUTPUT: **8813200023188 IS A PALINDROME**

INPUT: Enter number: **295**
OUTPUT: **CANNOT GENERATE A PALINDROME**

3.7
```
INPUT: Enter N: 4
Enter coefficients for row1
Co1: 2
Co2: -1
Co3: 0
Co4: -1
Enter constant: 1
Enter coefficients for row2
Co1: 3
Co2: 0
Co3: 1
Co4: 1
Enter constant: 1
Enter coefficients for row3
Co1: 1
Co2: 1
Co3: 0
Co4: 2
Enter constant: 0
Enter coefficients for row4
Co1: 4
Co2: 0
Co3: -3
Co4: 2
Enter constant: 0

OUTPUT: (1, 3, 0, -2)
```

```
INPUT: Enter N: 3
Enter coefficients for row1
Co1: 3
Co2: 6
Co3: 3
Enter constant: 9
Enter coefficients for row2
Co1: 1
Co2: -1
Co3: 2
Enter constant: 9
Enter coefficients for row3
Co1: -2
Co2: 2
Co3: -1
Enter constant: -9

OUTPUT: (2, -1, 3)
```

3.8 INPUT: Enter word: **FILE**
Enter K: **5**
OUTPUT: **ELFI FILE IFEL**

INPUT: Enter word: **COMPUTE**
Enter K: **721**
OUTPUT: **ECMOPTU MCEOPUT OCEMTPU**

3.9 RUN PROGRAM: (Spot check the 17 solutions given below. Make sure 108 solutions are printed and numbered. The solutions may be in any numerical order.)

OUTPUT:
```
411 - 21 = 390  NUMBER 1
511 - 21 = 490  NUMBER 2
611 - 21 = 590  NUMBER 3
711 - 21 = 690  NUMBER 4
811 - 21 = 790  NUMBER 5
511 - 31 = 480  NUMBER 6
611 - 31 = 580  NUMBER 7
711 - 31 = 680  NUMBER 8
:
:
:
377 - 87 = 290  NUMBER 100
477 - 87 = 390  NUMBER 101
577 - 87 = 490  NUMBER 102
677 - 87 = 590  NUMBER 103
277 - 97 = 180  NUMBER 104
377 - 97 = 280  NUMBER 105
477 - 97 = 380  NUMBER 106
577 - 97 = 480  NUMBER 107
677 - 97 = 580  NUMBER 108

TOTAL NUMBER OF SOLUTIONS = 108
```

3.10 RUN PROGRAM: (All left-most 2-digit numbers must appear. The examples on the right may vary, as long as the addens tally to the left-most number.)

OUTPUT:
```
45 = 0 + 1 + 2 + 3 + 4 + 5 + 7 + 8 + 9
54 = 10 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9
63 = 20 + 1 + 3 + 4 + 5 + 6 + 7 + 8 + 9
72 = 10 + 23 + 4 + 5 + 6 + 7 + 8 + 9
81 = 10 + 32 + 4 + 5 + 6 + 7 + 8 + 9
90 = 20 + 31 + 4 + 5 + 6 + 7 + 8 + 9
99 = 10 + 24 + 35 + 6 + 7 + 8 + 9
```

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '87
JUDGING CRITERIA

1.1 INPUT: Enter a number: **0** OUTPUT: **ZERO**

INPUT: Enter a number: **34.5** OUTPUT: **POSITIVE**

INPUT: Enter a number: **-99** OUTPUT: **NEGATIVE**

1.2 INPUT: Enter n: **32** OUTPUT: **882**

INPUT: Enter n: **-10** OUTPUT: **0**

1.3 RUN PROGRAM:

OUTPUT: (The following is centered on the screen both top to bottom and from left to right):

```
P
 R
  O
   B
    L
     E
      M

        T
         H
          R
           E
            E
```

1.4 INPUT: Enter number on top: **1**
Enter number on front: **4**
Enter number on right: **5**

OUTPUT: **TOP= 1**
FRONT= 4
RIGHT= 5
BOTTOM= 6
BACK= 3
LEFT= 2

INPUT: Enter number on top: **4**
Enter number on front: **2**
Enter number on right: **6**

OUTPUT: **TOP= 4**
FRONT= 2
RIGHT= 6
BOTTOM= 3
BACK= 5
LEFT= 1

1.5 RUN PROGRAM:

OUTPUT: (The screen must be filled with random characters. The computer then pauses and waits for a key to be pressed)

INPUT: (Press a key)
OUTPUT: (The screen will then clear)

1.6 INPUT: Enter coordinates: **3,2, 7,8**

OUTPUT: (a rectangular array of dots 5 rows by 7 columns)
(The upper left hand corner of the rectangle will be in position 2 of row 3 on the screen):

```
.......
.......
.......
.......
.......
```

1.7 INPUT: Enter seed: **27**

OUTPUT: **68**
29
10
11
32
73
34
15
16
37

INPUT: Enter seed: **3**

OUTPUT: **64**
45
46
67
8
69
50
51
72
13

1.8 INPUT: Enter K,L,W,H: **100, 20, 10, 5**
OUTPUT: **28416.847 KILOGRAMS**

INPUT: Enter K,L,W,H: **26, 3, 2, 1**
OUTPUT: **195.90 KILOGRAMS**

1.9 RUN PROGRAM:

OUTPUT:
```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
B  B  B  B  B  B  B  B  B  B  B
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
D  D  D  D  D  D  D  D  D  D  D
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
F  F  F  F  F  F  F  F  F  F  F
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
H  H  H  H  H  H  H  H  H  H  H
IIIIIIIIIIIIIIIIIIIIIIIIIIIIII
J  J  J  J  J  J  J  J  J  J  J
KKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
L  L  L  L  L  L  L  L  L  L  L
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
N  N  N  N  N  N  N  N  N  N  N
OOOOOOOOOOOOOOOOOOOOOOOOOOOOOO
P  P  P  P  P  P  P  P  P  P  P
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
R  R  R  R  R  R  R  R  R  R  R
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
T  T  T  T  T  T  T  T  T  T  T
UUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
```

1.10 INPUT: Enter book title: **THE ART OF WINNING**
Enter rate (minutes/page): **2.5**
OUTPUT: **10 HOURS 25 MINUTES**

INPUT: Enter book title: **THE HISTORY OF THE COMPUTER**
Enter rate (minutes/page): **3**
OUTPUT: **20 HOURS 0 MINUTES**

2.1 INPUT: Enter string: **PROGRAMMER**
Enter N: **7**
OUTPUT: **MERPROGRAM**

INPUT: Enter string: **COMPUTER**
Enter N: **41**
OUTPUT: **OMPUTERC**

2.2 RUN PROGRAM:

OUTPUT: **33 VERS 46 MAXS 21 WABS**

2.3 INPUT: Enter list item: **2**
Enter list item: **4**
Enter list item: **6**
Enter list item: **8**
Enter list item: **10**
Enter list item: **12**
Enter list item: **14**
Enter list item: **16**
Enter list item: **18**
Enter list item: **20**
Enter list item: **22**
Enter list item: **24**
Enter list item: **-1**
OUTPUT: (5 unique random numbers in this list will appear)
INPUT: (press any key)
OUTPUT: (5 unique random numbers in the list will appear)
INPUT: (press any key)
OUTPUT: (5 unique random numbers in the list will appear)
(These 3 OUTPUTS must not all contain the same
5 numbers)

2.4 INPUT: **8**
OUTPUT:

```
        8
      4+4
    2+2+2+2
1+1+1+1+1+1+1+1
```

INPUT: **18**
OUTPUT:

```
                  18
                 9+9
               6+6+6
            3+3+3+3+3+3
        2+2+2+2+2+2+2+2+2
1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1
```

2.5 INPUT: **DAD**
OUTPUT: **3/2**

INPUT: **BOY**
OUTPUT: **91/150**

2.6 INPUT: Enter set item: **8**
Enter set item: **2**
Enter set item: **1**
Enter set item: **16**
Enter set item: **35**
Enter set item: **3**
Enter set item: **-1**
Enter N: **4**
Enter S: **16**
OUTPUT: **YES**
1 2 3 8

INPUT: Enter set item: **8**
Enter set item: **2**
Enter set item: **1**
Enter set item: **16**
Enter set item: **35**
Enter set item: **3**
Enter set item: **-1**
Enter N: **5**
Enter S: **29**
OUTPUT: **NO**

2.7 INPUT: Enter pattern: **ABABAAAAA**
OUTPUT: **LEGAL PATTERN**

INPUT: Enter pattern: **ABAABAA**
OUTPUT: **ILLEGAL PATTERN**

INPUT: Enter pattern: **CBCBCC**
OUTPUT: **ILLEGAL PATTERN**

INPUT: Enter pattern: **AAAA**
OUTPUT: **LEGAL PATTERN**

2.8 INPUT: Enter M,N,F: **2, 50, 3**

OUTPUT: **4**
9
25
49

INPUT: Enter M,N,F: **750, 999, 18**

OUTPUT: **768**
800
828
882
972
980

2.9 INPUT: Enter word 1: **PENCIL**
Enter word 2: **PAPER**
Enter word 3: **CONTEST**
Enter word 4: **FCIC**
Enter word 5: **COMPUTER**

OUTPUT: **PAPER**
FCIC
PENCIL
COMPUTER
CONTEST

2.10 INPUT: Enter ROW, COL: **5, 5**
Enter MAX: **8**
Enter TYPE: **1**
(The program should accept input at ROW 5, COLumn 5)
INPUT: **ABCD F2**
OUTPUT: **ABCD F** (The program must not display the "2")
INPUT: **GHI**
OUTPUT: **ABCD FGH**
INPUT: (Press the "backspace" key 6 times.)
OUTPUT: **AB**
INPUT: **-** (dash)
OUTPUT: **AB**
INPUT: (Press RETURN key)
OUTPUT: **AB** (will be printed two rows beneath the typed AB)

INPUT: Enter ROW, COL: **10, 5**
Enter MAX: **5**
Enter TYPE: **2**
(The program should accept input at ROW 10, COL 5)
INPUT: **123.45**
OUTPUT: **123.4** (The program should not display the "5")
INPUT: (Press the "backspace" key 7 times.)
OUTPUT: (The entry should clear starting with the last character, but the cursor must not go past the first character spot (where the 1 was)).
INPUT: **23A**
OUTPUT: **23** (The program should not display the "A").
INPUT: (Press the RETURN key.)
OUTPUT: **23** (will be printed two rows beneath the typed 23).

INPUT: Enter ROW, COL: **7, 15**
Enter MAX: **8**
Enter TYPE: **3**
(The program should accept input at ROW 7, COL 15)
INPUT: **105**
OUTPUT: **10** (The program must not display the 5)
INPUT: **-164**
OUTPUT: **10-16**
INPUT: **-66**
OUTPUT: **10-16-66**
INPUT: (Press the RETURN key.)
OUTPUT: **10-16-66** (will be printed two rows beneath the typed 10-16-66).

INPUT: Enter ROW, COL: **1,1**
Enter MAX: **10**
Enter TYPE: **4**
(The program should accept input at ROW 1, COL 1)
INPUT: **12AB. $34**
OUTPUT: **12AB. $34**
INPUT: (Press the RETURN key.)
OUTPUT: **12AB. $34** (will be printed two rows beneath the typed 12AB. $34).

3.1 INPUT: Enter word 1: **FINALLY**
Enter word 2: **FINLALY** OUTPUT: **CLOSE**

INPUT: Enter word 1: **REAL**
Enter word 2: **RALE** OUTPUT: **NOT CLOSE**

INPUT: Enter word 1: **PRINTER**
Enter word 2: **PRINTE** OUTPUT: **CLOSE**

INPUT: Enter word 1: **PROGRAM**
Enter word 2: **GROGRAM** OUTPUT: **CLOSE**

INPUT: Enter word 1: **APPLE**
Enter word 2: **APPPLE** OUTPUT: **CLOSE**

3.2 Note: numbers are entered one at a time, one per line.
INPUT: Enter Dimension N: **2**
Enter numbers: **1, 2** (1st row of determinant)
Enter numbers: **3, 4** (2nd row of determinant)
OUTPUT: **-2**

INPUT: Enter Dimension N: **3**
Enter numbers: **1, 2, 3** (1st row of determinant)
Enter numbers: **4, 5, 6** (2nd row of determinant)
Enter numbers: **7, 8, 9** (3rd row of determinant)
OUTPUT: **0**

INPUT: Enter Dimension N: **4**
Enter numbers: **1, 9, 2, 8** (1st row)
Enter numbers: **3, 7, 4, 6** (2nd row)
Enter numbers: **5, 5, 0, 9** (3rd row)
Enter numbers: **8, 7, 6, 5** (4th row)
OUTPUT: **-410**

3.3 INPUT: Enter text: **BE MY BE MY BABY. BE MY BABY GIRL.**
OUTPUT: **3 BE**
3 MY
2 BABY
1 GIRL

INPUT: Enter text: **CAN'T YOU SEE? THIS PROGRAM WORKS! YOU WILL SEE THIS TOO.**
OUTPUT: (continued on next page)

```
    OUTPUT: 1 CAN'T
            2 YOU
            2 SEE
            2 THIS
            1 PROGRAM
            1 WORKS
            1 WILL
            1 TOO


3.4  INPUT: /255ABCD//123
    OUTPUT: (will be a string of characters)

  ** INPUT: (the string received as OUTPUT above)
            (write this string on paper to use in last test case)
    OUTPUT: /255ABCD//123

     INPUT: $1.89/0132YZ
    OUTPUT: (will be a string of characters)

     INPUT: (the string received as OUTPUT above)
    OUTPUT: $1.89/0132YZ

  ** INPUT: (the string received from the 1st OUTPUT
             -- you have it written on paper)
    OUTPUT: /255ABCD//123


3.5  RUN PROGRAM: (the two sets below may be in reverse order)

     OUTPUT: 3512  4357  15301784
             3125  3547  11084375


3.6  INPUT: Enter T, F, or Q: T
    OUTPUT: O  O
            Y  Y
     INPUT: Enter T, F, or Q: F
    OUTPUT: Y  O
            Y  O
     INPUT: Enter T, F, or Q: T
    OUTPUT: W  G
            Y  O
     INPUT: Enter T, F, or Q: F
    OUTPUT: Y  W
            O  G
     INPUT: Enter T, F, or Q: F
    OUTPUT: O  Y
            G  W
     INPUT: Enter T, F, or Q: T
    OUTPUT: B  R
            G  W
     INPUT: Enter T, F, or Q: Q
    OUTPUT: (program terminates)
```

3.7 INPUT: Enter name: **FRED**
Enter date: **04-06-87**

OUTPUT: (Screen will clear and display menu:)
1. INSTRUCTION PAGE
2. PRACTICE 3 PROBLEMS
3. QUIT

INPUT: **1**

OUTPUT: **YOU WILL BE GIVEN 3 PROBLEMS TO WORK. A PROBLEM WILL CONSIST OF ADDING TWO RANDOMLY GENERATED ROMAN NUMERALS LESS THAN 20. YOU WILL TYPE YOUR ANSWER IN ROMAN NUMERALS AND PRESS `RETURN.' (PRESS ANY KEY TO RETURN TO MENU.)**

INPUT: (Press any key)

OUTPUT: (Screen will clear and display the menu:)
1. INSTRUCTION PAGE
2. PRACTICE 3 PROBLEMS
3. QUIT

INPUT: **2**

OUTPUT: (Two randomly generated Roman Numerals appear in the center of the screen with a + AND a space on the left of bottom Numeral and a dash underneath this numeral extending from the + to the right-most character of the bottom numeral.)
For example:

```
          XIX
       + XIII
       ------
```

INPUT: **I**

OUTPUT: (The Arabic numeral for the answer must appear on the screen (on the bottom). Another chance to solve the problem is given.)

INPUT: **I**

OUTPUT: (Another randomly generated set of Roman Numerals will appear on the screen in the CORRECT format.)

INPUT: (Enter a correct answer (if possible). If you answer incorrectly, make sure you answer it correctly on the second chance. (see appendix))

OUTPUT: (A third problem will appear on the screen.)

INPUT: (Enter a correct answer (if possible). If you answer incorrectly, make sure you answer it correctly on the second chance. (see appendix))

OUTPUT: **PROGRESS REPORT**
DATE: 04-06-87
NAME: FRED
NUMBER CORRECT: 2
NUMBER OF EXERCISES: 3
PERCENT CORRECT: 67

WRONG ANSWER	**CORRECT ANSWER**	**ARABIC**
I	(in Roman Numerals)	(the sum)

PRESS ANY KEY TO RETURN TO MENU.

INPUT: Press any key

OUTPUT: (The screen will clear and display the menu:)
1. INSTRUCTION PAGE
2. PRACTICE 3 PROBLEMS
3. QUIT

INPUT and OUTPUT: 2 (Choose to Practice 3 problems)
Quickly answer each problem. It is acceptable to miss every problem for the sake of time. (If there is an easy problem, you may answer it correctly- just remember how many problems you answered correctly).

OUTPUT: **PROGRESS REPORT**
DATE: 04-06-87
NAME: FRED
NUMBER CORRECT: (You decide)
NUMBER OF EXERCISES: 3
PERCENT CORRECT: (0 or 33 or 67 or 100)

WRONG ANSWER	**CORRECT ANSWER**	**ARABIC**
(user's last answer)	(Roman Numeral)	(the sum)

(If you miss 2 or 3 problems, then 2 or 3 wrong answer numerals will appear.)

PRESS ANY KEY TO RETURN TO MENU.

INPUT: (Press any key)

OUTPUT: (The screen will clear and display the menu:)
1. INSTRUCTION PAGE
2. PRACTICE 3 PROBLEMS
3. QUIT

INPUT: **3**
OUTPUT: (program terminates)

APPENDIX OF ROMAN NUMERALS FOR 3.7

1 = I	11 = XI	21 = XXI	31 = XXXI
2 = II	12 = XII	22 = XXII	32 = XXXII
3 = III	13 = XIII	23 = XXIII	33 = XXXIII
4 = IV	14 = XIV	24 = XXIV	34 = XXXIV
5 = V	15 = XV	25 = XXV	35 = XXXV
6 = VI	16 = XVI	26 = XXVI	36 = XXXVI
7 = VII	17 = XVII	27 = XXVII	37 = XXXVII
8 = VIII	18 = XVIII	28 = XXVIII	38 = XXXVIII
9 = IX	19 = XIX	29 = XXIX	
10 = X	20 = XX	30 = XXX	

3.8 INPUT: Enter X,Y: **-1,-4**
Enter X,Y: **-3,-4**
Enter X,Y: **-3,-1**
Enter X,Y: **-1,-1**

Enter A,B: **-3,-5**
Enter A,B: **-4,-5**
Enter A,B: **-4,-4**
Enter A,B: **-3,-4**

OUTPUT: **0**

INPUT: Enter X,Y: **-1,-5**
Enter X,Y: **-5,-5**
Enter X,Y: **-5,-2**
Enter X,Y: **-1,-2**

Enter A,B: **-2,-15**
Enter A,B: **-10,-15**
Enter A,B: **-10,-3**
Enter A,B: **-2,-3**

OUTPUT: **6**

INPUT: Enter X,Y: **-3,-9**
Enter X,Y: **-9,-9**
Enter X,Y: **-9,-3**
Enter X,Y: **-3,-3**

Enter A,B: **-4,-9**
Enter A,B: **-10,-9**
Enter A,B: **-10,-5**
Enter A,B: **-4,-5**

OUTPUT: **20**

3.9 INPUT: Enter first number: **152415787517146788750190521 0**
Enter second number: **12345678901234567890**

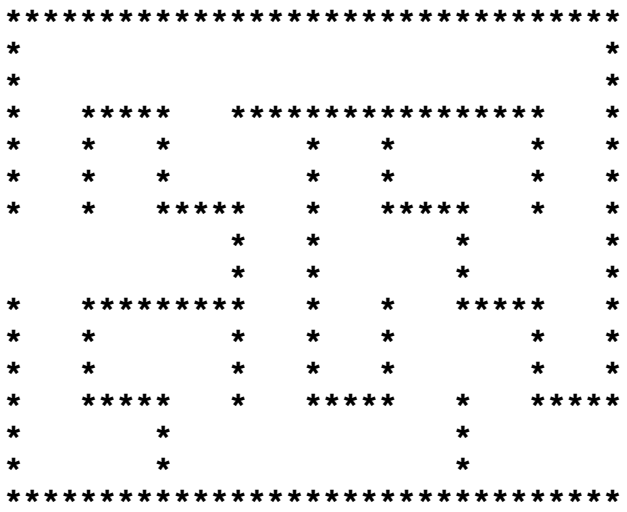
OUTPUT: **123456789 REMAINDER 0**

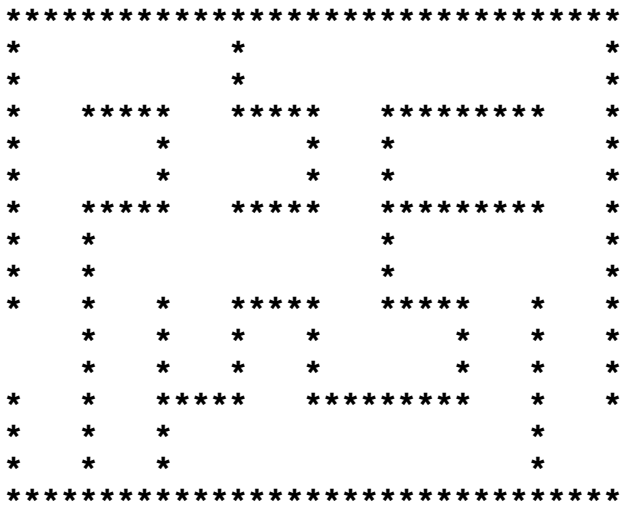
INPUT: Enter first number: **98765432109876543210987654321**
Enter second number: **123456789**

OUTPUT: **800000007370000067076 REMAINDER 75357**

3.10 RUN PROGRAM TWICE:

OUTPUT: A RANDOMLY generated maze (similar to below).





(The outer perimeter must be 33 asterisks long and 16 asterisks wide. The maze must contain 8 vertical paths and 5 horizontal paths. There must be one open spot on each side of the maze. There must be a UNIQUE solution. Every area in the maze must be attainable (no closed off areas). Every "spot" must have a wall of asterisks (no large blank areas).)

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '88
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: (The screen will clear, and the following will be displayed on the first 10 lines:)
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!
THE BEST COMPUTER CONTEST!

1.2 INPUT: Enter #: **-9.0**
OUTPUT: **INTEGER**

INPUT: Enter #: **3.21**
OUTPUT: **REAL**

1.3 INPUT: Enter N: **10**
OUTPUT: **1638400**

INPUT: Enter N: **5**
OUTPUT: **819200**

1.4 INPUT: Enter component: **PRIMARY**
Enter component: **CPU**
Enter component: **OUTPUT**
Enter component: **INPUT**
OUTPUT: **SECONDARY**

INPUT: Enter component: **CPU**
Enter component: **PRIMARY**
Enter component: **SECONDARY**
Enter component: **OUTPUT**
OUTPUT: **INPUT**

1.5 RUN PROGRAM: OUTPUT: (The screen's perimeter will be outlined with asterisks (*) and divided into four approximately congruent rectangles using *'s. The numbers 1, 2, 3, 4 will be centered in each rectangle as shown below in miniature.

```
**************
*     *     *
*  1  *  2  *
*     *     *
**************
*     *     *
*  3  *  4  *
*     *     *
**************
```

1.6 INPUT: Enter words: **CENTRAL PROCESSING UNIT**
OUTPUT: **CPU**

INPUT: Enter words: **PROGRAMMABLE READ ONLY MEMORY**
OUTPUT: **PROM**

1.7
```
INPUT: Enter name: MAX
       Enter type: MAINFRAME
       Enter name: MIKE
       Enter type: MICRO
       Enter name: MILTON
       Enter type: MINI

OUTPUT: MIKE
        MILTON
        MAX
```

```
INPUT: Enter name: ANDRIA
       Enter type: MICRO
       Enter name: LISA
       Enter type: MAINFRAME
       Enter name: KIM
       Enter type: MINI

OUTPUT: ANDRIA
        KIM
        LISA
```

1.8 INPUT: Enter N: **10**
OUTPUT: **30**

INPUT: Enter N: **15**
OUTPUT: **64**

1.9
```
INPUT: Enter command: ADD
       Enter integer: 11
       Enter command: TAKE
OUTPUT: 11
INPUT: Enter command: ADD
       Enter integer: 22
       Enter command: TAKE
OUTPUT: 22
INPUT: Enter command: QUIT
OUTPUT: (program terminates)
```

```
INPUT: Enter command: ADD
       Enter integer: 6
       Enter command: ADD
       Enter integer: 7
       Enter command: TAKE
OUTPUT: 6
INPUT: Enter command: ADD
       Enter integer: 4
       Enter command: TAKE
OUTPUT: 7
INPUT: Enter command: TAKE
OUTPUT: 4
INPUT: Enter command: QUIT
OUTPUT: (program terminates)
```

1.10 INPUT: Enter years: **1900, 1988**
OUTPUT: **HOWARD AIKEN INVENTED MARK I**
ECKERT AND MAUCHLY INVENTED ENIAC
VON NEUMAN INVENTED EDVAC

INPUT: Enter years: **1810, 1880**
OUTPUT: **CHARLES BABBAGE INVENTED DESIGN OF ANALYTIC ENGINE**

2.1 INPUT: Enter N: 15

OUTPUT:

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

INPUT: Enter N: **7**

OUTPUT:

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

2.2 INPUT: Enter N: **6**

OUTPUT:
BUBBLE SORT
QUICK SORT
SHELL SORT

INPUT: Enter N: **81**

OUTPUT:
QUICK SORT
BUBBLE SORT
SHELL SORT

INPUT: Enter N: **82**

OUTPUT:
QUICK SORT
SHELL SORT
BUBBLE SORT

2.3 RUN PROGRAM:

OUTPUT: **191**

2.4 INPUT: Enter seed: **1098**

OUTPUT: **560**
3600
9600
1600
6000

INPUT: Enter seed: **9987**

OUTPUT: **7401**
7748
315
2250
6250

2.5 INPUT: **11001100, EVEN**
OUTPUT: **CORRECT**

INPUT: **11111111, EVEN**
OUTPUT: **CORRECT**

INPUT: **00000000, ODD**
OUTPUT: **ERROR**

INPUT: **1A01A000, EVEN**
OUTPUT: **ERROR**

INPUT: **101010, ODD**
OUTPUT: **ERROR**

2.6 INPUT: Enter n: **6**
Enter vertex: **5,1**
Enter vertex: **2,4**
Enter vertex: **-3,3**
Enter vertex: **-3,-2**
Enter vertex: **-1,-4**
Enter vertex: **2,-2**

OUTPUT: **AREA = 41.5**

INPUT: Enter n: **3**
Enter vertex: **1,1**
Enter vertex: **5,5**
Enter vertex: **-1,-4**

OUTPUT: **AREA = 6.00**

2.7 INPUT: Enter month, day, year: **3, 1, 1988**

OUTPUT: **2-29-1988**
3-2-1988

INPUT: Enter month, day, year: **12, 31, 1777**

OUTPUT: **12-30-1777**
1-1-1778

INPUT: Enter month, day, year: **2, 29, 1980**

OUTPUT: **2-28-1980**
3-1-1980

2.8
```
 INPUT: Enter grade, credits: A, 2
        Enter grade, credits: D, 5
        Enter grade, credits: F, 5
        Enter grade, credits: F, 5
OUTPUT:  GPA= 0.765
        CGPA= 0.765
        STUDENT IS DISMISSED


 INPUT: Enter grade, credits: C, 2
        Enter grade, credits: D, 4
        Enter grade, credits: D, 5
        Enter grade, credits: C, 4
OUTPUT:  GPA= 1.400
        CGPA= 1.400

 INPUT: Enter grade, credits: A, 4
        Enter grade, credits: B, 4
        Enter grade, credits: C, 3
        Enter grade, credits: B, 3
OUTPUT:  GPA= 3.071
        CGPA= 2.207

 INPUT: Enter grade, credits: B, 4
        Enter grade, credits: C, 2
        Enter grade, credits: D, 5
        Enter grade, credits: F, 4
OUTPUT:  GPA= 1.400
        CGPA= 1.932

 INPUT: Enter grade, credits: C, 4
        Enter grade, credits: D, 5
        Enter grade, credits: A, 3
        Enter grade, credits: B, 2
OUTPUT:  GPA= 2.214
        CGPA= 2.000

 INPUT: Enter grade, credits: F, 5
        Enter grade, credits: F, 5
        Enter grade, credits: F, 4
        Enter grade, credits: F, 3
OUTPUT:  GPA= 0.000
        CGPA= 1.547

 INPUT: Enter grade, credits: C, 4
        Enter grade, credits: D, 2
        Enter grade, credits: C, 4
        Enter grade, credits: D, 3
OUTPUT:  GPA= 1.615
        CGPA= 1.557
        STUDENT IS DISMISSED
```

2.9 INPUT: Enter Desired Voltage, Tolerance: **6, 1.5**

OUTPUT: **NO BATTERY CAN BE FORMED**

INPUT: Enter Desired Voltage, Tolerance: **0.68, 0.0**

OUTPUT: **TIN IODINE 0.68**

INPUT: Enter Desired Voltage, Tolerance: 2.1, 0.5

OUTPUT: (in any order)

LITHIUM	**ZINC**	**2.29**
SODIUM	**ZINC**	**1.95**
SODIUM	**IRON**	**2.27**
SODIUM	**TIN**	**2.57**
ZINC	**MERCURY**	**1.61**
ZINC	**BROMINE**	**1.85**
ZINC	**CHLORINE**	**2.12**
IRON	**CHLORINE**	**1.80**

INPUT: Enter Desired Voltage, Tolerance: **1.5, 0.4**

OUTPUT: (continued in any order)

ZINC	**IODINE**	**1.30**
ZINC	**SILVER**	**1.56**
ZINC	**MERCURY**	**1.61**
ZINC	**BROMINE**	**1.85**
IRON	**SILVER**	**1.24**
IRON	**MERCURY**	**1.29**
IRON	**BROMINE**	**1.53**
IRON	**CHLORINE**	**1.80**

PRESS ANY KEY FOR MORE

INPUT: (press any key)

OUTPUT: (continued in any order)

TIN	**BROMINE**	**1.23**
TIN	**CHLORINE**	**1.50**

2.10 INPUT: Place 1: **A**
Place 2: **B**
Place 3: **C**
Place 4: **A**
Place 5: **B**
Place 6: **C**
Place 7: **A**
Place 8: **B**
Place 9: **C**
Place 10: **C**
Place 11: **B**
Place 12: **A**
Place 13: **C**
Place 14: **B**
Place 15: **C**
Place 16: **B**
Place 17: **A**
Place 18: **A**
Place 19: **C**
Place 20: **B**
Place 21: **A**

OUTPUT: (in any order)
TEAM A: 28 POINTS
TEAM B: 28 POINTS
TEAM B WINS!

TEAM A: 28 POINTS
TEAM C: 28 POINTS
TEAM C WINS!

TEAM B: 27 POINTS
TEAM C: 28 POINTS
TEAM B WINS!

INPUT: Place 1: **A**
Place 2: **B**
Place 3: **A**
Place 4: **C**
Place 5: **C**
Place 6: **B**
Place 7: **A**
Place 8: **A**
Place 9: **B**
Place 10: **C**
Place 11: **C**
Place 12: **A**
Place 13: **C**
Place 14: **B**
Place 15: **A**
Place 16: **A**
Place 17: **B**
Place 18: **B**
Place 19: **B**
Place 20: **C**
Place 21: **C**

OUTPUT: (in any order)
TEAM A: 23 POINTS
TEAM B: 34 POINTS
TEAM A WINS!

TEAM A: 23 POINTS
TEAM C: 32 POINTS
TEAM A WINS!

TEAM B: 29 POINTS
TEAM C: 26 POINTS
TEAM C WINS!

3.1 INPUT: Enter N: **4**
Enter #: **523**
Enter #: **321**
Enter #: **899.6**
Enter #: **66.79**

OUTPUT: **66.79**
899.6
523
321

INPUT: Enter N: **6**
Enter #: **-1.009**
Enter #: **54.32**
Enter #: **81.4**
Enter #: **-8.8**
Enter #: **7.3456**
Enter #: **-6.7**

OUTPUT: **-6.7**
-1.009
-8.8
7.3456
81.4
54.32

3.2 INPUT: Enter AMOUNT: **0.25**

OUTPUT: **13**

INPUT: Enter AMOUNT: **1.00**

OUTPUT: **242**

INPUT: Enter AMOUNT: **1.79**

OUTPUT: **1022**

3.3 INPUT: Enter point: **1.5, -1, -2.5**
Enter cube1 diagonal point1: **2, -3, 6**
Enter cube1 diagonal point2: **4.5, -4, 5.5**
Enter cube2 diagonal point1: **0, -1, -3**
Enter cube2 diagonal point2: **4.5, -3.5, 12**

OUTPUT: **POINT LIES INSIDE 2ND CUBE**
1ST CUBE DOES NOT LIE INSIDE 2ND CUBE

INPUT: Enter point: **1, 2, 3**
Enter cube1 diagonal point1: **4, 5, 6**
Enter cube1 diagonal point2: **-1, -2, 8**
Enter cube2 diagonal point1: **3, 6, 9**
Enter cube2 diagonal point2: **8, 2, 1**

OUTPUT: **POINT DOES NOT LIE INSIDE 2ND CUBE**
1ST CUBE DOES NOT LIE INSIDE 2ND CUBE

3.4 INPUT: **ABAA**

OUTPUT: **AAAB**
AABA
ABAA
BAAA
TOTAL= 4

INPUT: **ABC**

OUTPUT: **ABC**
ACB
BAC
BCA
CAB
CBA
TOTAL= 6

INPUT: **CBABB**

OUTPUT: **ABBBC**
ABBCB
ABCBB
ACBBB
BABBC
BABCB
BACBB
BBABC
BBACB
BBBAC
BBBCA
BBCAB
BBCBA
BCABB
BCBAB
BCBBA
CABBB
CBABB
CBBAB
CBBBA
TOTAL= 20

3.5 RUN PROGRAM:

OUTPUT: [A snake (a trail of 25 asterisks '*') is centered on the screen. Upon hitting appropriate keys (I, J, K, and M), the snake's head moves in the appropriate direction while the rest of the snake slithers along the same right angle paths. The snake is to move CONTINUOUSLY in the designated direction UNTIL a new directional key is hit. The snake will be 25 asterisks long throughout the entire run--no sketched path. The snake cannot go backwards, e.g. if it is going right, then its next direction cannot be to the left. The snake continues moving until it runs into itself or it runs off the screen or a non-directional key is pressed.

TEST FOR ALL THIS. YOU BE THE JUDGE.]

3.6 INPUT: Enter equation 1: **10X-5Y-5=0**
Enter equation 2: **-4X-3Y-7=0**

OUTPUT: **XSOLUTION= -0.4 YSOLUTION= -1.8**

INPUT: Enter equation 1: **X-0Y=2**
Enter equation 2: **3X+0Y=1**

OUTPUT: **NO UNIQUE SOLUTION EXISTS**

INPUT: Enter equation 1: **-X-Y=-2**
Enter equation 2: **2X-3Y-14=0**

OUTPUT: **XSOLUTION= 4.0 YSOLUTION= -2.0**

INPUT: Enter equation 1: **2X-5Y-20=0**
Enter equation 2: **4X-10Y-10=0**

OUTPUT: **NO UNIQUE SOLUTION EXISTS**

3.7 RUN PROGRAM:

OUTPUT:

SEMI #	EXAMPLE(S)
6	1 + 2 + 3
12	2 + 4 + 6
12	1 + 2 + 3 + 6
18	3 + 6 + 9
18	1 + 2 + 6 + 9
20	1 + 4 + 5 + 10
24	4 + 8 + 12
24	1 + 3 + 8 + 12
24	2 + 4 + 6 + 12
24	1 + 2 + 3 + 6 + 12
24	1 + 2 + 3 + 4 + 6 + 8
28	1 + 2 + 4 + 7 + 14
30	5 + 10 + 15
30	2 + 3 + 10 + 15
30	1 + 3 + 5 + 6 + 15

3.8 INPUT: **12,1/,2/,X,X,X,51,X,X,X9/**

OUTPUT:
```
-1- -2- -3- -4- -5- -6- -7- -8- -9- -10-
---!---!---!---!---!---!---!---!---!---!
 12! 1/! 2/!  X!  X!  X! 51!  X!  X!X9/!
3  !15 !35 !65 !90 !106!112!142!171!191!
---------------------------------------
```

INPUT: **72,90,X,72,7/,X,7/,9/,9/,-5**

OUTPUT:
```
-1- -2- -3- -4- -5- -6- -7- -8- -9- -10-
---!---!---!---!---!---!---!---!---!---!
 72! 90!  X! 72! 7/!  X! 7/! 9/! 9/! -5!
9  !18 !32 !46 !66 !86 !105!124!134!139!
---------------------------------------
```

3.9 INPUT: Enter M, N, #: **8, 16, 7.654321**

OUTPUT: **7.D6344**

INPUT: Enter M, N, #: **15, 11, A.CE**

OUTPUT: **A.954**

INPUT: Enter M, N, #: **10, 3, 2.987**

OUTPUT: **2.2221221**

3.10 INPUT: Enter the ORDER of p(x): **4**
Enter coefficient for x**4: **4**
Enter coefficient for x**3: **3**
Enter coefficient for x**2: **2**
Enter coefficient for x**1: **1**
Enter coefficient for x**0: **0**

Enter the ORDER of q(x): **1**
Enter coefficient for x**1: **2**
Enter coefficient for x**0: **-1**

OUTPUT: **P(Q(X))= 64X**4 + -104X**3 + 68X**2 + -20X**1 + 2X**0**
Q(P(X))= 8X4 + 6X**3 + 4X**2 + 2X**1 + -1X**0**

INPUT: Enter the ORDER of p(x): **2**
Enter coefficient for x**2: **-5**
Enter coefficient for x**1: **2**
Enter coefficient for x**0: **5**

Enter the ORDER of q(x): **0**
Enter coefficient for x**0: **-1**

OUTPUT: **P(Q(X))= -2X**0**
Q(P(X))= -1X0**

INPUT: Enter the ORDER of p(x): **2**
Enter coefficient for x**2: **-1**
Enter coefficient for x**1: **0**
Enter coefficient for x**0: **3**

Enter the ORDER of q(x): **2**
Enter coefficient for x**2: **4**
Enter coefficient for x**1: **0**
Enter coefficient for x**0: **-2**

OUTPUT: **P(Q(X))= -16X**4 + 0X**3 + 16X**2 + 0X**1 + -1X**0**
Q(P(X))= 4X4 + 0X**3 + -24X**2 + 0X**1 + 34X**0**

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '89
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: (The screen will display the following phrase, with each line indented 1 space more than the preceding)

```
1989 COMPUTER CONTEST
 1989 COMPUTER CONTEST
  1989 COMPUTER CONTEST
   :
    :
     :
      1989 COMPUTER CONTEST
```

1.2 INPUT: Enter number of gigabytes: **29**
OUTPUT: **29696 MEGABYTES**

INPUT: Enter number of gigabytes: **7**
OUTPUT: **7168 MEGABYTES**

1.3 INPUT: Enter word: **COMPUTER**

```
OUTPUT:        C
               O
               M
               P
               U
               T
               E
          COMPUTER
```

INPUT: Enter word: **EASY**

```
OUTPUT:    E
           A
           S
        EASY
```

1.4 INPUT: Enter N: **7**

```
OUTPUT:       1
             2 2
            3   3
           4     4
          5       5
         6         6
        7           7
```

INPUT: Enter N: **2**

```
OUTPUT:  1
        2 2
```

1.5
```
 INPUT: Enter date: 11
        Enter A.D. or B.C.: A.D.
OUTPUT: 15 A.D.

 INPUT: Enter date: 1
        Enter A.D. or B.C.: B.C.
OUTPUT: 4 A.D.

 INPUT: Enter date: 9
        Enter A.D. or B.C.: B.C.
OUTPUT: 5 B.C.
```

1.6
```
OUTPUT/INPUT:  ENTER PASSWORD: LETMEIN
OUTPUT/INPUT:  INVALID PASSWORD: ITSME
OUTPUT:        YOU HAVE ACCESS

OUTPUT/INPUT:  ENTER PASSWORD: DOUG
OUTPUT/INPUT:  INVALID PASSWORD: CRAIG
OUTPUT/INPUT:  INVALID PASSWORD: BRAD
OUTPUT:        YOU ARE TRESSPASSING
```

1.7
```
 INPUT: Enter N: 2
        Enter DBMS name: DEB
        Enter convenience, efficiency: 5, 3
        Enter DBMS name: KIM
        Enter convenience, efficiency: 8, 1
OUTPUT: KIM IS BEST

 INPUT: Enter N: 4
        Enter DBMS name: COM1
        Enter convenience, efficiency: 5, 9
        Enter DBMS name: COM2
        Enter convenience, efficiency: 9, 3
        Enter DBMS name: COM3
        Enter convenience, efficiency: 5, 2
        Enter DBMS name: COM4
        Enter convenience, efficiency: 5, 5
OUTPUT: COM1 IS BEST
```

1.8
```
 INPUT: Enter #: 10
        Enter #: -1
        Enter #: -6
        Enter #: -1
        Enter #: -1
        Enter #: -2
        Enter #: 10
        Enter #: -999
OUTPUT: 10 -1 -6 -2

 INPUT: Enter #: 15
        Enter #: 2
        Enter #: 3
        Enter #: 15
        Enter #: 2
        Enter #: 2
        Enter #: -999
OUTPUT: 15 2 3
```

1.9 INPUT: Enter probability: **5E17**
OUTPUT: **8 FEET DEEP**

INPUT: Enter probability: **9.8E18**
OUTPUT: **164 FEET DEEP**

1.10 INPUT: Enter Seg#, Address: **0, 250**
OUTPUT: **469**
INPUT: Enter Seg#, Address: **8, 50**
OUTPUT: (program terminates)

INPUT: Enter Seg#, Address: **4, 100**
OUTPUT: **ADDRESSING ERROR**
INPUT: Enter Seg#, Address: **3, 500**
OUTPUT: **1827**
INPUT: Enter Seg#, Address: **7, 10**
OUTPUT: (program terminates)

2.1
```
INPUT: Enter x: 7        INPUT: Enter x: 9        INPUT: Enter x: 2
OUTPUT: F(7)= 29         OUTPUT: F(9)= 169        OUTPUT: F(2)= 1
```

2.2
```
INPUT: 980                          INPUT: 79
OUTPUT: 2 X 2 X 5 X 7 X 7           OUTPUT: 79

INPUT: 608
OUTPUT: 2 X 2 X 2 X 2 X 2 X 19
```

2.3
```
INPUT: Enter word: COMPUTER         INPUT: Enter word: FLORIDA
OUTPUT: CMPTR                       OUTPUT: FLRD
```

2.4
```
INPUT: Enter name: NUMBER           INPUT: Enter name: MAXIMUM
       Enter name: INDEX                   Enter name: COUNT
       Enter name: INSIDE                  Enter name: COUNTER
       Enter name: NUM                     Enter name: HOURS
       Enter name: J                       Enter name: MAXNUM
       Enter name: COUNT                   Enter name: MARGIN
OUTPUT: NUMB                        OUTPUT: MAXI
        IND                                 COUNT
        INS                                 COUNTE
        NUM                                 H
        J                                   MAXN
        C                                   MAR
```

2.5
```
INPUT: Enter word: MISSISSIPPI
OUTPUT: 34650

INPUT: Enter word: REHEARSE
OUTPUT: 3360

INPUT: Enter word: RELEASE
OUTPUT: 840
```

```
2.6  INPUT: Enter sentence: *ONE*TWO*THREE*FOUR*FIVE*SIX
     OUTPUT: (Screen is cleared)
             *ONE*TWO*THREE*FOUR*FIVE*SIX

             ONETWOTHREEFOURFIVESIX
             ---   -----    ----

      INPUT: *THIS ENTIRE LINE IS UNDERLINED*
     OUTPUT: (Screen is cleared)
             *THIS ENTIRE LINE IS UNDERLINED*

             THIS ENTIRE LINE IS UNDERLINED
             ------------------------------



2.7  INPUT: Enter expression: 10+9900
     OUTPUT: 9910

      INPUT: Enter expression: 1005*19
     OUTPUT: 19095

      INPUT: Enter expression: 5-234
     OUTPUT: -229

      INPUT: Enter expression: 1224/24
     OUTPUT: 51



2.8  INPUT: Enter #Rows, #Cols: 3, 4
             Enter Row1 Col1: 2
             Enter Row1 Col2: -2
             Enter Row1 Col3: -4
             Enter Row1 Col4: -8
             Enter Row2 Col1: 0
             Enter Row2 Col2: 4
             Enter Row2 Col3: -2
             Enter Row2 Col4: 2
             Enter Row3 Col1: -8
             Enter Row3 Col2: -4
             Enter Row3 Col3: -6
             Enter Row3 Col4: 6
     OUTPUT: SADDLE POINT = -2 AT ROW 2 COL 3

      INPUT: Enter #Rows, #Cols: 2, 2
             Enter Row1 Col1: 4
             Enter Row1 Col2: 3
             Enter Row2 Col1: 1
             Enter Row2 Col2: 2
     OUTPUT: SADDLE POINT = 3 AT ROW 1 COL 2
```

2.9 INPUT: Enter # of dates: **4**
Enter month: **APRIL**
Enter day: **23**
Enter year: **1988**

Enter month: **OCTOBER**
Enter day: **16**
Enter year: **1966**

Enter month: **APRIL**
Enter day: **8**
Enter year: **1989**

Enter month: **JUNE**
Enter day: **3**
Enter year: **1980**

OUTPUT: **OCTOBER 16 1966**
JUNE 3 1980
APRIL 23 1988
APRIL 8 1989

INPUT: Enter # of dates: **2**
Enter month: **MARCH**
Enter day: **3**
Enter year: **1980**

Enter month: **MARCH**
Enter day: **1**
Enter year: **1980**

OUTPUT: **MARCH 1 1980**
MARCH 3 1980

2.10 RUN PROGRAM:

OUTPUT:

NAME	Q1	Q2	Q3	Q4
D. WOOLY	100	92	90	90
M. SMITH	55	75	70	65
C. BROWN	94	70	62	70
R. GREEN	90	74	80	85
T. STONE	85	98	100	70

INPUT: Enter 5 grades for quiz 4: **95, 68, 70, 85, 75**

OUTPUT: (Screen is cleared)

MS. HEINDEL'S MUSIC CLASS
FINAL GRADES
SPRING 1989

NAME	Q1	Q2	Q3	Q4	AVERAGE
D. WOOLY	100	92	90	95	94.25
M. SMITH	55	75	70	68	67.00
C. BROWN	94	70	62	70	74.00
R. GREEN	90	74	80	85	82.25
T. STONE	85	98	100	75	89.50
AVERAGE:	84.80	81.80	80.40	78.60	

CLASS AVERAGE: 81.40

3.1

INPUT: Enter word: **ABLE**	OUTPUT: **CORRECT**
INPUT: Enter word: **SPELLL**	OUTPUT: **MISSPELLED**
INPUT: Enter word: **SIBLING**	OUTPUT: **CORRECT**
INPUT: Enter word: **PIERCE**	OUTPUT: **CORRECT**
INPUT: Enter word: **PEACEABLE**	OUTPUT: **MISSPELLED**
INPUT: Enter word: **CONCIEVE**	OUTPUT: **MISSPELLED**
INPUT: Enter word: **SEIVE**	OUTPUT: **MISSPELLED**

3.2 RUN PROGRAM:

```
OUTPUT: P =  0.05  V = 0.4097
        P =  0.70  V = 0.4122
        P = 10.00  V = 0.4518
        P = 70.00  V = 1.2263

 INPUT: Enter value for P: 50.00

OUTPUT: P = 50.00  V = 0.7744
```

3.3 INPUT: Enter number: **8720**
Enter magnification: **1**

OUTPUT:
```
****  ****  ****  ****
*  *     *     *  *  *
****     *  ****  *  *
*  *     *  *     *  *
****     *  ****  ****
```

INPUT: Enter number: **631**
Enter magnification: **2**

OUTPUT:
```
********    ********           *
*                   *          *
*                   *          *
*                   *          *
********    ********           *
*      *            *          *
*      *            *          *
*      *            *          *
********    ********           *
```

INPUT: Enter number: **9**
Enter magnification: **3**

OUTPUT:
```
************
*          *
*          *
*          *
*          *
*          *
************
           *
           *
           *
           *
           *
           *
```

3.4 INPUT: Enter month, year: **4, 1989**

OUTPUT:

```
          APRIL 1989
 S   M   T   W   T   F   S
-------------------------
                        1
 2   3   4   5   6   7   8
 9  10  11  12  13  14  15
16  17  18  19  20  21  22
23  24  25  26  27  28  29
30
```

INPUT: Enter month, year: **2, 1980**

OUTPUT:

```
          FEBRUARY 1980
 S   M   T   W   T   F   S
-------------------------
                    1   2
 3   4   5   6   7   8   9
10  11  12  13  14  15  16
17  18  19  20  21  22  23
24  25  26  27  28  29
```

3.5 RUN PROGRAM:

OUTPUT:

```
ROWS =  1 2 3 4 5
----------------
COLUMNS
        1 3 5 2 4
        1 4 2 5 3
        2 4 1 3 5
        2 5 3 1 4
        3 1 4 2 5
        3 5 2 4 1
        4 1 3 5 2
        4 2 5 3 1
        5 2 4 1 3
        5 3 1 4 2
```

3.6 INPUT: Enter base: **4**
Enter first integer: **-123012301230123012301230**
Enter second integer: **-111100002222333300001111**
OUTPUT: **2100000102012300000020331233333231321033333313O**

INPUT: Enter base: **10**
Enter first integer: **12345678901234567890123456789O**
Enter second integer: **-9876543210**
OUTPUT: **-121932631124828532112482853211126352690O**

INPUT: Enter base: **8**
Enter first integer: **12345670123456701234567**
Enter second integer: **7654321076543210**
OUTPUT: **12170533614661671657306704402333351047O**

3.7 INPUT: Enter cost, amount: **9.19, 10.00**
Enter missing coin: **NICKEL**
OUTPUT: **1 PENNY**
3 DIMES
2 QUARTERS
TOTAL CHANGE RETURNED = 81 CENTS

INPUT: Enter cost, amount: **9.06, 10.00**
Enter missing coin: **QUARTER**
OUTPUT: **4 PENNIES**
0 NICKELS
9 DIMES
TOTAL CHANGE RETURNED = 94 CENTS

INPUT: Enter cost, amount: **19.60, 20.00**
Enter missing coin: **PENNY**
OUTPUT: **1 NICKEL**
1 DIME
1 QUARTER
TOTAL CHANGE RETURNED = 40 CENTS

3.8 INPUT: Enter number: **63** OUTPUT: **0111111**
Enter number: **62** **0111110**
Enter number: **63** **0111111**
Enter number: **100** **1100100**
Enter number: **127** **1111111**
Enter number: **31** **0011111**

(1,2)(3,6)
(5,3)(6,7)
(4,1)(5,2)

(Note: 3 lines of coordinates may be in any order)

INPUT: Enter number: **84** OUTPUT: **1010100**
Enter number: **127** **1111111**
Enter number: **127** **1111111**
Enter number: **127** **1111111**
Enter number: **84** **1010100**
Enter number: **84** **1010100**

(2,1)(4,7)

3.9 RUN PROGRAM:

```
OUTPUT: BIBLE  94   OBESE  89
        IDYLL 110   TITHE  95
        NOISE  79   INLET  95
        GULLY  98   IGLOO 100
        OBESE  89   TOWER  94
              470         473
                          ***

 INPUT: Enter word: BOOST
        Enter word: BILLY
        Enter word: GLORY
        Enter word: TOOTH
        Enter word: OLIVE
        Enter word: (Enter key pressed)

OUTPUT: BILLY 108   OBESE  89
        IDYLL 110   BILLY 108
        NOISE  79   INLET  95
        GULLY  98   IGLOO 100
        OLIVE  99   TOOTH  98
              494         490
              ***

 INPUT: Enter word: NOVEL
        Enter word: TIGER
        Enter word: BELOW
        Enter word: OWLET
        Enter word: TOTAL
        Enter word: TITLE
        Enter word: (Enter key pressed)

OUTPUT: BILLY 108    OBESE  89
        IDYLL 110    TITLE 118
        NOVEL  81    INLET  95
        GULLY  98    IGLOO 100
        OWLET 112    TOTAL 108
              509           510
                            ***

 INPUT: Enter word: QUIT

OUTPUT: (program terminates)
```

3.10 INPUT: Enter TOP side: **B**
Enter FRONT side: **B**
Enter BOTTOM side: **B**
Enter BACK side: **B**
Enter RIGHT side: **G**
Enter LEFT side: **G**
OUTPUT: **NUMBER OF DISTINGUISHABLE CUBES = 3**

INPUT: Enter TOP side: **B**
Enter FRONT side: **B**
Enter BOTTOM side: **B**
Enter BACK side: **B**
Enter RIGHT side: **G**
Enter LEFT side: **Y**
OUTPUT: **NUMBER OF DISTINGUISHABLE CUBES = 6**

INPUT: Enter TOP side: **G**
Enter FRONT side: **G**
Enter BOTTOM side: **G**
Enter BACK side: **B**
Enter RIGHT side: **B**
Enter LEFT side: **B**
OUTPUT: **NUMBER OF DISTINGUISHABLE CUBES = 12**

INPUT: Enter TOP side: **R**
Enter FRONT side: **R**
Enter BOTTOM side: **B**
Enter BACK side: **B**
Enter RIGHT side: **R**
Enter LEFT side: **B**
OUTPUT: **NUMBER OF DISTINGUISHABLE CUBES = 8**

INPUT: Enter TOP side: **G**
Enter FRONT side: **G**
Enter BOTTOM side: **B**
Enter BACK side: **B**
Enter RIGHT side: **B**
Enter LEFT side: **Y**
OUTPUT: **NUMBER OF DISTINGUISHABLE CUBES = 24**

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '90
JUDGING CRITERIA

1.1 RUN PROGRAM:

```
OUTPUT: NN    N  CCCCC  NN    N  BBBB
        N N   N  C      N N   N  B   B
        N  N  N  C      N  N  N  BBBBB
        N   N N  C      N   N N  B   B
        N    NN  CCCCC  N    NN  BBBB
```

1.2 INPUT: Enter #: **2**
OUTPUT: **SYSTEM 2**

INPUT: Enter #: **1**
OUTPUT: **SYSTEM 1**

1.3 INPUT: Enter N: **8**
OUTPUT: **74 BILLION DOLLARS**

INPUT: Enter N: **20**
OUTPUT: **86 BILLION DOLLARS**

1.4 INPUT: Enter zip code: **33613**
OUTPUT: **HILLSBOROUGH**

INPUT: Enter zip code: **34249**
OUTPUT: **PASCO**

INPUT: Enter zip code: **34646**
OUTPUT: **PINELLAS**

1.5 INPUT: Enter MMM: **120**
Enter YYYY: **1998**

OUTPUT: **HUGH MCCOLL WOULD LIKE NCNB TO GROW TO 120 BILLION DOLLARS IN ASSETS BY THE YEAR 1998**

INPUT: Enter MMM: **150**
Enter YYYY: **2000**

OUTPUT: **HUGH MCCOLL WOULD LIKE NCNB TO GROW TO 150 BILLION DOLLARS IN ASSETS BY THE YEAR 2000**

1.6 INPUT: Enter N associates: **7**
Enter C coupons: **50000**
OUTPUT: **7143**

INPUT: Enter N associates: **8**
Enter C coupons: **48800**
OUTPUT: **6100**

1.7 INPUT: Enter division: **PROCEDURE**
OUTPUT: **BEFORE = IDENTIFICATION ENVIRONMENT DATA**
AFTER = NONE

INPUT: Enter division: **ENVIRONMENT**
OUTPUT: **BEFORE = IDENTIFICATION**
AFTER = DATA PROCEDURE

1.8 INPUT: Enter N: **11**
OUTPUT: **MD**

INPUT: Enter N: **7**
OUTPUT: **FL NC SC TX MD GA VA**

INPUT: Enter N: **9**
OUTPUT: **FL TX MD GA VA**

1.9 INPUT: Enter date: **2**
Enter A.D. or B.C.: **B.C.**
OUTPUT: **3 A.D.**

INPUT: Enter date: **15**
Enter A.D. or B.C.: **B.C.**
OUTPUT: **11 B.C.**

INPUT: Enter date: **10**
Enter A.D. or B.C.: **A.D.**
OUTPUT: **14 A.D.**

1.10 INPUT: Enter word: **FLORIDA**

OUTPUT:
R
ORI
LORID
FLORIDA
LORID
ORI
R

INPUT: Enter word: **PROGRAM**

OUTPUT:
G
OGR
ROGRA
PROGRAM
ROGRA
OGR
G

2.1 INPUT: Enter phrase: **CALL THE POLICE**

OUTPUT: **BZKK SGD ONKHBD**

INPUT: Enter phrase: **DON'T PANIC**

OUTPUT: **CNM'S OZMHB**

2.2 INPUT: Enter year: **1000** OUTPUT: **END OF DECADE**
END OF CENTURY
END OF MILLENNIUM

INPUT: Enter year: **1001** OUTPUT: **BEGINNING OF DECADE**
BEGINNING OF CENTURY
BEGINNING OF MILLENNIUM

INPUT: Enter year: **1990** OUTPUT: **END OF DECADE**

INPUT: Enter year: **1801** OUTPUT: **BEGINNING OF DECADE**
BEGINNING OF CENTURY

2.3 INPUT: Enter scores for Bob: **170, 160, 215**
Enter scores for Doug: **199, 209, 198**
Enter scores for Jackie: **135, 144, 150**
Enter scores for Jose: **110, 101, 180**

OUTPUT: **BOB: AVERAGE = 181 HANDICAP = 16**
DOUG: AVERAGE = 202 HANDICAP = 0
JACKIE: AVERAGE = 143 HANDICAP = 51
JOSE: AVERAGE = 130 HANDICAP = 62

2.4 INPUT: Enter date: **02/11/1732** OUTPUT: **ADD 11 DAYS**

INPUT: Enter date: **02/28/1900** OUTPUT: **ADD 12 DAYS**

INPUT: Enter date: **03/01/1600** OUTPUT: **ADD 10 DAYS**

INPUT: Enter date: **12/01/1900** OUTPUT: **ADD 13 DAYS**

2.5 INPUT: Enter N: **6** OUTPUT: **BUBBLE SORT**
QUICK SORT
SHELL SORT

INPUT: Enter N: **81** OUTPUT: **QUICK SORT**
BUBBLE SORT
SHELL SORT

INPUT: Enter N: **82** OUTPUT: **QUICK SORT**
SHELL SORT
BUBBLE SORT

2.6 INPUT: Enter score for hole 1: **6**
Enter score for hole 2: **4**
Enter score for hole 3: **4**
Enter score for hole 4: **4**
Enter score for hole 5: **2**
Enter score for hole 6: **4**
Enter score for hole 7: **2**
Enter score for hole 8: **5**
Enter score for hole 9: **6**

OUTPUT:

HOLE	PAR	SCORE	STATUS
1	4	6	DOUBLE BOGEY
2	3	4	BOGEY
3	4	4	PAR
4	5	4	BIRDIE
5	4	2	EAGLE
6	3	4	BOGEY
7	5	2	DOUBLE EAGLE
8	4	5	BOGEY
9	4	6	DOUBLE BOGEY
	36	37	

2.7 Note: Output must be within 0.1 second of correct answer.

INPUT: Enter N: **95**
OUTPUT: **0 DAYS 0 HOURS 15 MIN 41.0 SEC AHEAD**

INPUT: Enter N: **7**
OUTPUT: **0 DAYS 16 HOURS 41 MIN 34.6 SEC AHEAD**

INPUT: Enter N: **132**
OUTPUT: **1 DAYS 0 HOURS 38 MIN 50.4 SEC BEHIND**

INPUT: Enter N: **1507**
OUTPUT: **10 DAYS 23 HOURS 23 MIN 25.4 SEC BEHIND**

2.8 INPUT: Enter month, year: **8, 1990**

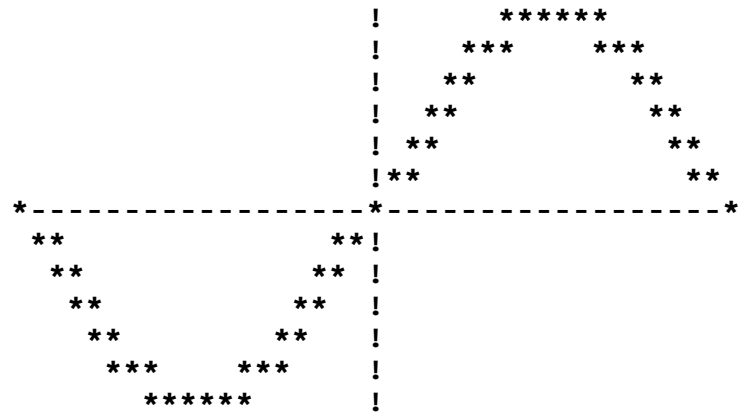
OUTPUT:
```
 9/1989 - BARB  JOE  DOUG
12/1989 - JACKIE  JOE  DOUG
 2/1990 - JACKIE  TOM  DOUG
 3/1990 - JACKIE  TOM  LOVETTA
 6/1990 - GREG  TOM  LOVETTA
 8/1990 - GREG  TONY  LOVETTA
```

INPUT: Enter month, year: **1, 1992**

OUTPUT:
```
 9/1989 - BARB  JOE  DOUG
12/1989 - JACKIE  JOE  DOUG
 2/1990 - JACKIE  TOM  DOUG
 3/1990 - JACKIE  TOM  LOVETTA
 6/1990 - GREG  TOM  LOVETTA
 8/1990 - GREG  TONY  LOVETTA
 9/1990 - GREG  TONY  AL
12/1990 - KAREN  TONY  AL
 2/1991 - KAREN  JAN  AL
 3/1991 - KAREN  JAN  NORM
 6/1991 - TRUDY  JAN  NORM
 8/1991 - TRUDY  THERESA  NORM
 9/1991 - TRUDY  THERESA  ALICE
12/1991 - DAVE  THERESA  ALICE
```

2.9 RUN PROGRAM:

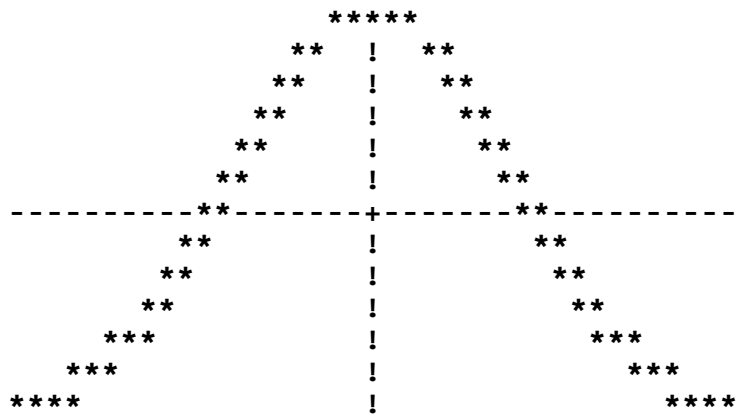
OUTPUT: (Screen clears and the axes is drawn before the graph is drawn from left to right. Graph will look similar to below, but it extends to the dimensions of the terminal.)



INPUT: (Press any key)

OUTPUT: (continued on next page)

OUTPUT: (Screen clears and the axes is drawn before the graph is drawn from left to right- similar to below.)



INPUT: (Press any key) OUTPUT: (Screen clears)

2.10 RUN PROGRAM:

OUTPUT: **NCNB IN-HOUSE TRAINING LIST**

COURSE #	**COURSE NAME**	**EST. HOURS**
187-11X	**ISPF/PDS FUNDAMENTALS**	**6.5 - 8**
187-15X	**ISPF/PDS FOR PROGRAMMERS**	**4.5 - 6**
220-AXX	**JCL FUNDAMENTALS**	**15 - 20**
200-AXX	**VSAM CONCEPTS**	**4 - 7**
123-2XX	**MVS/SP/XA VSAM**	**7 - 11**
130-11X	**CICS/VS SKILLS I**	**6 - 8**
130-15X	**CICS/VS SKILLS II**	**4 - 6**

INPUT: Enter course # (or 000-000 to end): **187-15X**
Enter course # (or 000-000 to end): **130-15X**
Enter course # (or 000-000 to end): **123-2XX**
Enter course # (or 000-000 to end): **200-AXX**
Enter course # (or 000-000 to end): **000-000**

OUTPUT: (Screen is cleared)

COURSE NAME	**EST. HOURS**
ISPF/PDS FOR PROGRAMMERS	**4.5 - 6**
CICS/VS SKILLS II	**4 - 6**
MVS/SP/XA VSAM	**7 - 11**
VSAM CONCEPTS	**4 - 7**
	TOTAL = 19.5 - 30 HOURS

3.1 INPUT: Enter phone #: **555-6625** OUTPUT: **55K-NOCK**

INPUT: Enter phone #: **555-7283** OUTPUT: **555-SAVE**
555-PAVE
555-RATE

INPUT: Enter phone #: **555-6229** OUTPUT: **55L-OBBY**

3.2 INPUT: Enter string: **COMPUTE***
OUTPUT: **COMPUTE COMPUTER COMPUTERS COMPUTES COMPUTED**
INPUT: Enter string: ***TIVE**
OUTPUT: **ATTRACTIVE ADAPTIVE ACCEPTIVE CREATIVE**
INPUT: Enter string: **CONTEST*S**
OUTPUT: **CONTESTS CONTESTERS**
INPUT: Enter string: **EVERY*TY**
OUTPUT: **NO WORDS FOUND**
INPUT: Enter string: **QUIT**
OUTPUT: (Program terminates)

3.3 INPUT: Place 1: **A**
Place 2: **B**
Place 3: **C**
Place 4: **A**
Place 5: **B**
Place 6: **C**
Place 7: **A**
Place 8: **B**
Place 9: **C**
Place 10: **C**
Place 11: **B**
Place 12: **A**
Place 13: **C**
Place 14: **B**
Place 15: **C**
Place 16: **B**
Place 17: **A**
Place 18: **A**
Place 19: **C**
Place 20: **B**
Place 21: **A**

OUTPUT: (On next page)

INPUT: Place 1: **A**
Place 2: **B**
Place 3: **A**
Place 4: **C**
Place 5: **C**
Place 6: **B**
Place 7: **A**
Place 8: **A**
Place 9: **B**
Place 10: **C**
Place 11: **C**
Place 12: **A**
Place 13: **C**
Place 14: **B**
Place 15: **A**
Place 16: **A**
Place 17: **B**
Place 18: **B**
Place 19: **B**
Place 20: **C**
Place 21: **C**

OUTPUT: (On next page)

(Output continued)

OUTPUT: (in any order)

```
TEAM A: 28 POINTS
TEAM B: 28 POINTS
TEAM B WINS!

TEAM A: 28 POINTS
TEAM C: 28 POINTS
TEAM C WINS!

TEAM B: 27 POINTS
TEAM C: 28 POINTS
TEAM B WINS!
```

OUTPUT: (in any order)

```
TEAM A: 23 POINTS
TEAM B: 34 POINTS
TEAM A WINS!

TEAM A: 23 POINTS
TEAM C: 32 POINTS
TEAM A WINS!

TEAM B: 29 POINTS
TEAM C: 26 POINTS
TEAM C WINS!
```

3.4 INPUT: Enter X, Y, Z: **4, 3, 5**

OUTPUT:

```
AL, DOUG, AND JAN = NONE
AL AND DOUG = 12 24
AL AND JAN = 20
DOUG AND JAN = 15 30
AL = 4 8 16 28
DOUG = 3 6 9 18 21 27
JAN = 5 10 25
NORM = 1 2 7 11 13 14 17 19 22 23 26 29
```

INPUT: Enter X, Y, Z: **4, 3, 2**

OUTPUT:

```
AL, DOUG, AND JAN = 12 24
AL AND DOUG = NONE
AL AND JAN = 4 8 16 20 28
DOUG AND JAN = 6 18 30
AL = NONE
DOUG = 3 9 15 21 27
JAN = 2 10 14 22 26
NORM = 1 5 7 11 13 17 19 23 25 29
```

3.5 RUN PROGRAM: OUTPUT: (A 3 x 3 array of random digits 1 - 8 are displayed along with a blank location). It will be similar (but not identical) to the following 3 x 3 array of numbers:

```
3  5  7
8  1
4  2  6
```

Press the numbers 1 - 8, and check to see if a number that is vertically or horizontally adjacent to the blank moves into the blank location (such as 7, 1, or 6 in this case). Press a number that is not next to a blank to make sure that it does not move (such as 3, 5, 8, 4, or 2). Press the digit 9 to terminate program. Run program one more time and perform the same tests.

3.6 RUN PROGRAM:

```
    OUTPUT: BR1 BK1 BB1 BQ  BK  BB2 BK2 BR2  !  8
            BP1 BP2 BP3 BP4 BP5 BP6 BP7 BP8  !  7
                                             !  6
                                             !  5
                                             !  4
                                             !  3
            WP1 WP2 WP3 WP4 WP5 WP6 WP7 WP8  !  2
            WR1 WK1 WB1 WQ  WK  WB2 WK2 WR2  !  1
           ----------------------------------
             A   B   C   D   E   F   G   H

     INPUT: Enter white move: E2-E4

    OUTPUT: BR1 BK1 BB1 BQ  BK  BB2 BK2 BR2  !  8
            BP1 BP2 BP3 BP4 BP5 BP6 BP7 BP8  !  7
                                             !  6
                                             !  5
                        WP5                  !  4
                                             !  3
            WP1 WP2 WP3 WP4     WP6 WP7 WP8  !  2
            WR1 WK1 WB1 WQ  WK  WB2 WK2 WR2  !  1
            ----------------------------------
             A   B   C   D   E   F   G   H

     INPUT: Enter black move: B7-B6

    OUTPUT: BR1 BK1 BB1 BQ  BK  BB2 BK2 BR2  !  8
            BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
                BP2                          !  6
                                             !  5
                        WP5                  !  4
                                             !  3
            WP1 WP2 WP3 WP4     WP6 WP7 WP8  !  2
            WR1 WK1 WB1 WQ  WK  WB2 WK2 WR2  !  1
            ----------------------------------
             A   B   C   D   E   F   G   H

     INPUT: Enter white move: B1-C3

    OUTPUT: BR1 BK1 BB1 BQ  BK  BB2 BK2 BR2  !  8
            BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
                BP2                          !  6
                                             !  5
                        WP5                  !  4
                WK1                          !  3
            WP1 WP2 WP3 WP4     WP6 WP7 WP8  !  2
            WR1     WB1 WQ  WK  WB2 WK2 WR2  !  1
            ----------------------------------
             A   B   C   D   E   F   G   H

     INPUT: Enter black move: B8-C6
    OUTPUT: (on next page)
```

```
OUTPUT: BR1     BB1 BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
            BP2 BK1                      !  6
                                         !  5
                    WP5                  !  4
                WK1                      !  3
        WP1 WP2 WP3 WP4     WP6 WP7 WP8  !  2
        WR1     WB1 WQ  WK  WB2 WK2 WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter white move: E1-E2

OUTPUT: BR1     BB1 BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
            BP2 BK1                      !  6
                                         !  5
                    WP5                  !  4
                WK1                      !  3
        WP1 WP2 WP3 WP4 WK  WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2 WK2 WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter black move: C8-A6

OUTPUT: BR1         BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
        BB1 BP2 BK1                      !  6
                                         !  5
                    WP5                  !  4
                WK1                      !  3
        WP1 WP2 WP3 WP4 WK  WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2 WK2 WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter white move: C3-B5

OUTPUT: BR1         BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
        BB1 BP2 BK1                      !  6
            WK1                          !  5
                    WP5                  !  4
                                         !  3
        WP1 WP2 WP3 WP4 WK  WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2 WK2 WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter black move: A6-B5

OUTPUT: (on next page)
```

```
OUTPUT: BR1          BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
            BP2 BK1                      !  6
            BB1                          !  5
                        WP5              !  4
                                         !  3
        WP1 WP2 WP3 WP4 WK  WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2 WK2 WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter white move: G1-F3

OUTPUT: BR1          BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
            BP2 BK1                      !  6
            BB1                          !  5
                        WP5              !  4
                            WK2          !  3
        WP1 WP2 WP3 WP4 WK  WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2     WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

 INPUT: Enter white move: B5-E2

OUTPUT: BR1          BQ  BK  BB2 BK2 BR2  !  8
        BP1     BP3 BP4 BP5 BP6 BP7 BP8  !  7
            BP2 BK1                      !  6
                                         !  5
                        WP5              !  4
                            WK2          !  3
        WP1 WP2 WP3 WP4 BB1 WP6 WP7 WP8  !  2
        WR1     WB1 WQ      WB2     WR2  !  1
        ---------------------------------
         A   B   C   D   E   F   G   H

        CHECK MATE, BLACK WON
```

3.7 INPUT: Enter year: **1972**

OUTPUT: **EASTER IS ON APRIL 2**
LENT IS ON FEBRUARY 16

INPUT: Enter year: **1999**

OUTPUT: **EASTER IS ON APRIL 4**
LENT IS ON FEBRUARY 17

INPUT: Enter year: **1992**

OUTPUT: **EASTER IS ON APRIL 19**
LENT IS ON MARCH 4

3.8 INPUT: Enter frame 1: **12**
Enter frame 2: **1/**
Enter frame 3: **2/**
Enter frame 4: **X**
Enter frame 5: **X**
Enter frame 6: **X**
Enter frame 7: **51**
Enter frame 8: **X**
Enter frame 9: **X**
Enter frame 10: **X9/**

OUTPUT:
```
-1- -2- -3- -4- -5- -6- -7- -8- -9- -10-
---!---!---!---!---!---!---!---!---!---!
 12! 1/! 2/!  X!  X!  X! 51!  X!  X!X9/!
3  !15 !35 !65 !90 !106!112!142!171!191!
----------------------------------------
```

INPUT: Enter frame 1: **72**
Enter frame 2: **9-**
Enter frame 3: **X**
Enter frame 4: **72**
Enter frame 5: **7/**
Enter frame 6: **X**
Enter frame 7: **7/**
Enter frame 8: **9/**
Enter frame 9: **9/**
Enter frame 10: **-5**

OUTPUT:
```
-1- -2- -3- -4- -5- -6- -7- -8- -9- -10-
---!---!---!---!---!---!---!---!---!---!
 72! 9-!  X! 72! 7/!  X! 7/! 9/! 9/! -5!
9  !18 !32 !46 !66 !86 !105!124!134!139!
----------------------------------------
```

3.9
```
INPUT: Enter N: 4
Enter coefficients for row1
Co1: 2
Co2: -1
Co3: 0
Co4: -1
Enter constant: 1
Enter coefficients for row2
Co1: 3
Co2: 0
Co3: 1
Co4: 1
Enter constant: 1
Enter coefficients for row3
Co1: 1
Co2: 1
Co3: 0
Co4: 2
Enter constant: 0
Enter coefficients for row4
Co1: 4
Co2: 0
Co3: -3
Co4: 2
Enter constant: 0

OUTPUT: (1, 3, 0, -2)
```

```
INPUT: Enter N: 3
Enter coefficients for row1
Co1: 3
Co2: 6
Co3: 3
Enter constant: 9
Enter coefficients for row2
Co1: 1
Co2: -1
Co3: 2
Enter constant: 9
Enter coefficients for row3
Co1: -2
Co2: 2
Co3: -1
Enter constant: -9

OUTPUT: (2, -1, 3)
```

3.10
```
INPUT: Enter first addend:  AB
       Enter second addend: CD
       Enter sum: EBC

OUTPUT: (Only one of the following solutions must be shown)
A = 3    4    4    6    6    7    7    7    8    8    8    8
B = 2    2    3    3    5    2    5    6    3    4    5    7
C = 9 or 8 or 9 or 7 or 9 or 5 or 8 or 9 or 5 or 6 or 7 or 9
D = 7    6    6    4    4    3    3    3    2    2    2    2
E = 1    1    1    1    1    1    1    1    1    1    1    1
```

```
INPUT: Enter first addend:  AB
       Enter second addend: BC
       Enter sum: DCB

OUTPUT: (Only one of the following solutions must be shown)
       A = 2    3    4    6    7    8
       B = 8    7    6    4    3    2
       C = 0 or 0 or 0 or 0 or 0 or 0
       D = 1    1    1    1    1    1
```

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '91
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: (After the screen is cleared, the following appears):

```
COMPUTER CONTEST 1991
O                   9
M                   9
P                   1
U
T                   T
E                   S
R                   E
                    T
C                   N
O                   O
N                   C
T
E                   R
S                   E
T                   T
                    U
1                   P
9                   M
9                   O
1991 TSETNOC RETUPMOC
```

1.2 RUN PROGRAM:

OUTPUT: (2 random integers between -9 and 9, inclusive, are displayed, along with their sum. Output must be displayed in the form: X + Y = ZZ, where X and Y are random). Run the program four times. Make sure X and Y are random and the sum is correct. At least one space must appear between numbers and symbols. Examples:

```
 3  + -9 = -6    or

-5  + -2 = -7    or

-1  +  4 =  3    or

 2  +  0 =  2
```

1.3 INPUT: Enter team name: **KING HS**
Enter # of 1 point programs: **9**
Enter # of 2 point programs: **5**
Enter # of 3 point programs: **3**

OUTPUT: **KING HS SCORED 28 POINTS**

INPUT: Enter team name: **PLANTATION HS**
Enter # of 1 point programs: **10**
Enter # of 2 point programs: **7**
Enter # of 3 point programs: **4**

OUTPUT: **PLANTATION HS SCORED 36 POINTS**

1.4 RUN PROGRAM:

OUTPUT: **A B C D E F G H I J K L M N O P Q R S T**
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

1.5 INPUT: Enter number of students: **200**

OUTPUT: **50 TEAMS**

INPUT: Enter number of students: **172**

OUTPUT: **43 TEAMS**

1.6 INPUT: Enter word: **FLORIDA**
Enter letter: **D**

OUTPUT:
```
     F
     L
     O
     R
     I
FLORIDA
     A
```

INPUT: Enter word: **BRANCH**
Enter letter: **B**

OUTPUT:
```
BRANCH
R
A
N
C
H
```

1.7 INPUT: Enter account key: **00200712340023456 78**

OUTPUT:
```
ORGANIZATION 002
BRANCH 007
DEALER 1234
CLASS 002
UNIT 345678
```

INPUT: Enter account key: **01102203304405506 60**

OUTPUT:
```
ORGANIZATION 011
BRANCH 022
DEALER 0330
CLASS 440
UNIT 550660
```

1.8 INPUT: Enter line: **JOB**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **DD**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **//**

OUTPUT: **4 JOB STEPS**

INPUT: Enter line: **JOB**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **EXEC**
Enter line: **DD**
Enter line: **DD**
Enter line: **DD**
Enter line: **//**

OUTPUT: **2 JOB STEPS**

1.9 INPUT: Enter sentence:
THE MAN-EATING TIGER MANGLED THE WOMAN'S PURSE.

OUTPUT: **THE PERSON-EATING TIGER PERSONGLED THE WOPERSON'S PURSE.**

INPUT: Enter sentence:
MANY HUMAN ACTIVITIES EMANATE FROM MENTAL CONCENTRATION.

OUTPUT: (the following sentence must extend on one line):
PERSONY HUPERSON ACTIVITIES EPERSONATE FROM PERSONSTAL CONCENTRATION.

1.10 INPUT: Enter team name: **MIAMI HS**
Enter points, time, penalties: **41, 142, 7**
Enter team name: **CORAL GABLES**
Enter points, time, penalties: **41, 234, 1**

OUTPUT: **MIAMI HS WINS**

INPUT: Enter team name: **TARAVELLA**
Enter points, time, penalties: **32, 210, 5**
Enter team name: **CORAL SPRINGS**
Enter points, time, penalties: **33, 234, 1**

OUTPUT: **CORAL SPRINGS WINS**

INPUT: Enter team name: **KING HS**
Enter points, time, penalties: **33, 210, 5**
Enter team name: **CORAL SPRINGS**
Enter points, time, penalties: **33, 234, 1**

OUTPUT: **KING HS WINS**

2.1 INPUT: Enter N: **45**

OUTPUT:

```
                    01
                  02  03
                04  05  06
              07  08  09  10
            11  12  13  14  15
          16  17  18  19  20  21
        22  23  24  25  26  27  28
      29  30  31  32  33  34  35  36
    37  38  39  40  41  42  43  44  45
```

INPUT: Enter N: **10**

OUTPUT:

```
        01
      02  03
    04  05  06
  07  08  09  10
```

2.2 INPUT: Enter #: **2.2345**
Enter #: **4.567**
Enter #: **234.56**
Enter #: **7891.027**
Enter #: **123.456**

OUTPUT:

```
    2.2345
     4.567
   234.56
  7891.027
   123.456
  ---------
  8255.8445
```

INPUT: Enter #: **1234.5**
Enter #: **123.45**
Enter #: **12.345**
Enter #: **1.23**
Enter #: **78.901**

OUTPUT:

```
1234.5
 123.45
  12.345
   1.23
  78.901
 ---------
 1450.4260
```

2.3 INPUT: Enter statement: **IF A >= B OR C > D**

OUTPUT: **IF A IS NOT LESS THAN B OR C IS GREATER THAN D**

INPUT: Enter statement: **IF C >< B AND A = D**

OUTPUT: **IF C IS NOT EQUAL TO B AND A IS EQUAL TO D**

2.4 INPUT: Enter N: **6**
Enter team: **PAR FOUR**
Enter wins, losses: **25, 25**
Enter team: **STRIKES**
Enter wins, losses: **35, 15**
Enter team: **WINNERS**
Enter wins, losses: **35, 15**
Enter team: **RUNNERS**
Enter wins, losses: **25, 25**
Enter team: **FORCE**
Enter wins, losses: **15, 35**
Enter team: **HIGH ROLLERS**
Enter wins, losses: **35, 15**

OUTPUT:
```
1 HIGH ROLLERS  35 , 15
1 STRIKES       35 , 15
1 WINNERS       35 , 15

4 PAR FOUR      25 , 25
4 RUNNERS       25 , 25

6 FORCE         15 , 35
```

2.5 RUN PROGRAM:

```
OUTPUT: GUESS 1: 64
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 2: 32
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 3: 16
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 4: 8
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 5: 4
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 6: 6
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 7: 7
 INPUT: Enter H, L, or R: R
OUTPUT: (program terminates)
```

RUN PROGRAM:

```
OUTPUT: GUESS 1: 64
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 2: 32
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 3: 48
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 4: 56
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 5: 52
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 6: 50
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 7: 49
 INPUT: Enter H, L, or R: R
OUTPUT: (program terminates)
```

RUN PROGRAM:

```
OUTPUT: GUESS 1: 64
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 2: 96
 INPUT: Enter H, L, or R: H
OUTPUT: GUESS 3: 112
 INPUT: Enter H, L, or R: L
OUTPUT: GUESS 4: 104
 INPUT: Enter H, L, or R: R
OUTPUT: (program terminates)
```

2.6 INPUT: Enter text: **EACH SUCCEEDING LINE CONTAINS THE LEAST AMOUNT OF WORDS THAT CAUSES IT TO EXCEED THE PREVIOUS LINE BY AT LEAST 2 CHARACTERS.**

OUTPUT:

EACH
SUCCEEDING
LINE CONTAINS
THE LEAST AMOUNT
OF WORDS THAT CAUSES
IT TO EXCEED THE PREVIOUS
LINE BY AT LEAST 2 CHARACTERS.

INPUT: Enter text:
EACH LINE IS TO BE CENTERED WITH RESPECT TO COLUMN 20.

OUTPUT:

EACH
LINE IS
TO BE CENTERED
WITH RESPECT TO COLUMN
20.

2.7 INPUT: Enter length, width: **10, 5**

OUTPUT: (the following appears in the center of the screen):

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

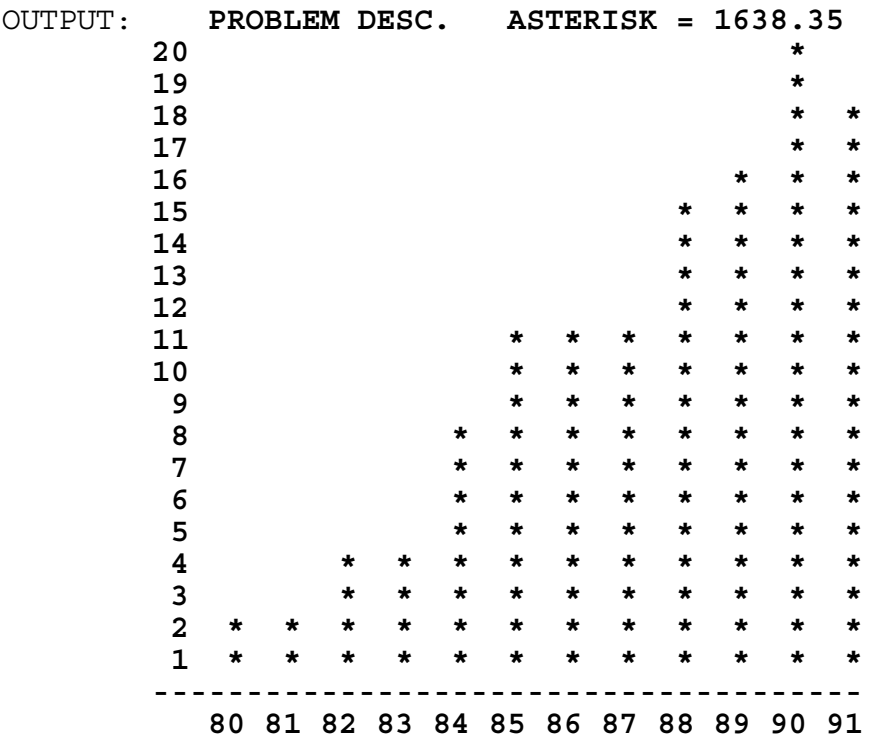
INPUT: Enter length, width: **15, 6**

OUTPUT: (the following appears in the center of the screen):

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

2.8 INPUT: Enter title: **PROBLEM DESC.**
Enter # for 1980: **3982**
Enter # for 1981: **3910**
Enter # for 1982: **7599**
Enter # for 1983: **7723**
Enter # for 1984: **14485**
Enter # for 1985: **20144**
Enter # for 1986: **19762**
Enter # for 1987: **21133**
Enter # for 1988: **26066**
Enter # for 1989: **26932**
Enter # for 1990: **32767**
Enter # for 1991: **29726**

OUTPUT:



2.9 INPUT: Enter # of entries in yesterday's file: **4**
Enter ID: **AB12**
Enter item: **D**
Enter ID: **CH39**
Enter item: **R**
Enter ID: **CH40**
Enter item: **D**
Enter ID: **CR11**
Enter item: **A**

INPUT: Enter # of entries in today's file: **5**
Enter ID: **AB12**
Enter item: **C**
Enter ID: **CH39**
Enter item: **R**
Enter ID: **CH41**
Enter item: **D**
Enter ID: **CR11**
Enter item: **B**
Enter ID: **DE12**
Enter item: **F**

OUTPUT: **ADDED**
CR41 D
DE12 F

CHANGED
AB12 D C
CR11 A B

DELETED
CH40 D

TOTAL ADDED = 2
TOTAL CHANGED = 2
TOTAL DELETED = 1

2.10 INPUT: Enter year: **1980**

OUTPUT: **FHS80-1.PRB**
FHS80-2.PRB
FHS80-3.PRB
FHS80-1.JDG
FHS80-2.JDG
FHS80-3.JDG
FHS80-1.PG1
FHS80-2.PG1
FHS80-3.PG1
FHS80-1.PG2
FHS80-2.PG2
FHS80-3.PG2
ONE1T80.BAS
ONE2T80.BAS
ONE3T80.BAS
ONE4T80.BAS
ONE5T80.BAS
ONE6T80.BAS
ONE7T80.BAS
ONE8T80.BAS
INPUT: (press any key)
OUTPUT: **ONE9T80.BAS**
ONE10T80.BAS
TWO1T80.BAS
TWO2T80.BAS
TWO3T80.BAS
TWO4T80.BAS
TWO5T80.BAS
TWO6T80.BAS
TWO7T80.BAS
TWO8T80.BAS
TWO9T80.BAS
TWO10T80.BAS
THR1T80.BAS
THR2T80.BAS
THR3T80.BAS
THR4T80.BAS
THR5T80.BAS
THR6T80.BAS
THR7T80.BAS
THR8T80.BAS
INPUT: (press any key)
OUTPUT: (continued on next page)

```
OUTPUT: THR9T80.BAS
        THR10T80.BAS
        THR11T80.BAS
        THR12T80.BAS
        ONE1T80.PAS
        ONE2T80.PAS
        ONE3T80.PAS
        ONE4T80.PAS
        ONE5T80.PAS
        ONE6T80.PAS
        ONE7T80.PAS
        ONE8T80.PAS
        ONE9T80.PAS
        ONE10T80.PAS
        TWO1T80.PAS
        TWO2T80.PAS
        TWO3T80.PAS
        TWO4T80.PAS
        TWO5T80.PAS
        TWO6T80.PAS
 INPUT: (press any key)
OUTPUT: TWO7T80.PAS
        TWO8T80.PAS
        TWO9T80.PAS
        TWO10T80.PAS
        THR1T80.PAS
        THR2T80.PAS
        THR3T80.PAS
        THR4T80.PAS
        THR5T80.PAS
        THR6T80.PAS
        THR7T80.PAS
        THR8T80.PAS
        THR9T80.PAS
        THR10T80.PAS
        THR11T80.PAS
        THR12T80.PAS
```

3.1 RUN PROGRAM: (twice)

OUTPUT: (Each run is random, but should be SIMILAR
to the following baseball game results.
Check that the score is correctly added.
99% of the time this program will have:
- each score in an inning less than 10,
- total # of strikes between 211 and 280,
- total # of balls between 290 and 470,
- total # of walks between 69 and 111.)

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**SCORE**
TEAM A !	2	3	0	0	0	1	0	0	3	9
TEAM B !	2	0	1	2	3	0	0	0	2	10

TOTAL # OF STRIKES: 247
TOTAL # OF BALLS: 403
TOTAL # OF WALKS: 92
TOTAL # OF STRIKE OUTS: 54

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**SCORE**
TEAM A !	0	2	0	1	0	2	0	0	1	6
TEAM B !	0	0	0	0	0	0	0	1	0	1

TOTAL # OF STRIKES: 239
TOTAL # OF BALLS: 337
TOTAL # OF WALKS: 76
TOTAL # OF STRIKE OUTS: 54

3.2 INPUT: Enter A, X: **2, 345**
Enter B, Y: **9, 876**
Enter C, Z: **11, 234**

OUTPUT: **4**

INPUT: Enter A, X: **6, 123**
Enter B, Y: **8, 456**
Enter C, Z: **14, 321**

OUTPUT: **6**

3.3 INPUT: Enter X, Y: **7, 89**

OUTPUT: (resulting digits should appear on one line on screen)
16357825134744349084771609590778780110077149747549
9697974493805316003428960 7

INPUT: Enter X, Y: **37, 73**

OUTPUT: (resulting digits will wrap around on the screen)
30111047009373532386032375415313753434442016552707
88531783697471044984243073342239158015955057079636
461132367224197

3.4 INPUT: Enter name: **DON A CHANG**
Enter name: **FRED B COOK**
Enter name: **DON RING**
Enter name: **FRED B CORN**
Enter name: **DAVID ALFRED CHANG**
Enter name: **END**

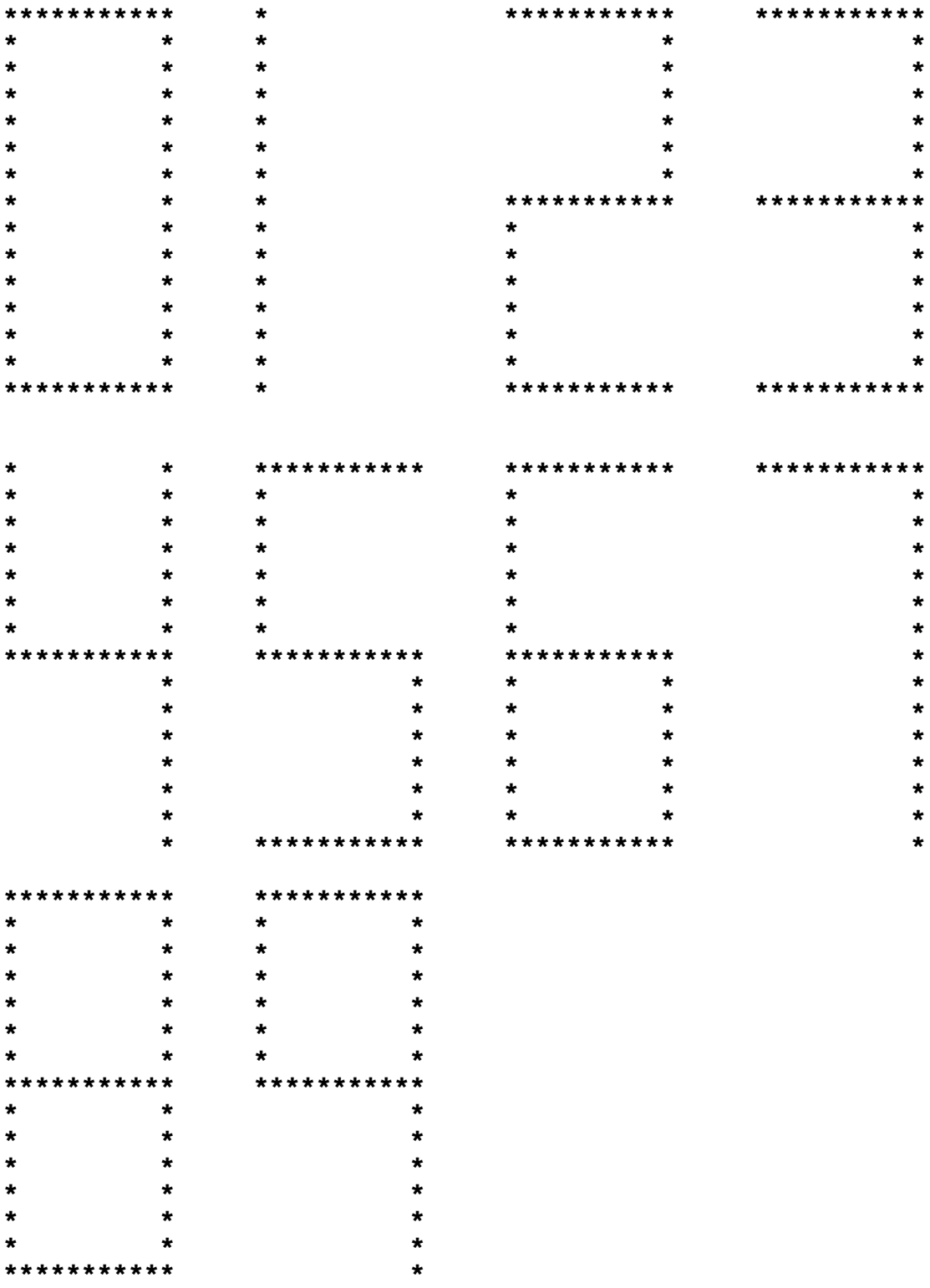
OUTPUT:
```
DON A CHANG         SDD2C1
FRED B COOK         SDF1C1
DON RING            SDDXR1
FRED B CORN         SDF2C1
DAVID ALFRED CHANG  SDD1C1
```

INPUT: Enter name: **DOUG E WOOLLEY**
Enter name: **DAVE E WEAVER**
Enter name: **BOB R JONES**
Enter name: **PAUL SIMON SMITH**
Enter name: **DON ENGLAND WANG**
Enter name: **END**

OUTPUT:
```
DOUG E WOOLLEY      SDD3W1
DAVE E WEAVER       SDD2W1
BOB R JONES         SDBRJ1
PAUL SIMON SMITH    SDPSS1
DON ENGLAND WANG    SDD1W1
```

3.5 RUN PROGRAM:

OUTPUT: (The numbers 0 through 9 appear, one at a time in enlarged block format [15 characters high by 11 characters wide]. Each number must be displayed for approximately 1 second before clearing the screen and displaying the next number.)



3.6 INPUT: Enter expression: **(((5-4)-9+3)-6+1)+2**
OUTPUT: **-8**

INPUT: Enter expression: **7-(8+9)+(6+4)-((1-9)+(2-8))**
OUTPUT: **14**

3.7 INPUT: Enter holiday MM, DD: **7, 4**
Enter holiday MM, DD: **9, 2**
Enter holiday MM, DD: **9, 30**
Enter holiday MM, DD: **10, 14**
Enter holiday MM, DD: **11, 28**
Enter holiday MM, DD: **12, 25**
Enter holiday MM, DD: **12, 31**
Enter holiday MM, DD: **0, 0**

INPUT: Enter month #: **9**

OUTPUT: **FRIDAY SEPTEMBER 13**
FRIDAY SEPTEMBER 27

INPUT: Enter month #: **12**

OUTPUT: **FRIDAY DECEMBER 13**
MONDAY DECEMBER 30

INPUT: Enter month #: **8**

OUTPUT: **THURSDAY AUGUST 15**
FRIDAY AUGUST 30

INPUT: Enter month #: **0**

OUTPUT: (program terminates)

3.8 INPUT: Enter digit: **1**
Enter row, col: **2, 3**

OUTPUT: **4 3 8**
9 5 1
2 7 6

2 7 6
9 5 1
4 3 8

(Note: The two magic squares may appear in either order.)

INPUT: Enter digit: **4**
Enter row, col: **1, 3**

OUTPUT: **2 9 4**
7 5 3
6 1 8

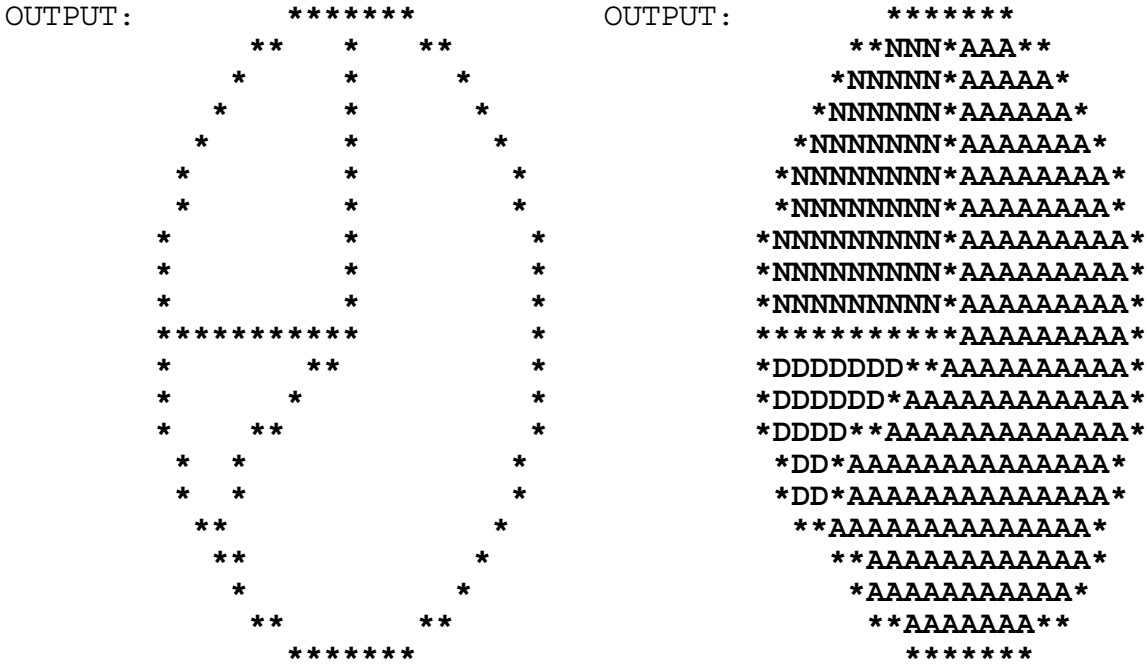
8 3 4
1 5 9
6 7 2

INPUT: Enter digit: **1**
Enter row, col: **3, 3**

OUTPUT: **NO SOLUTION**

3.9 INPUT: Enter 3 percentages: **64, 11, 25**

OUTPUT:

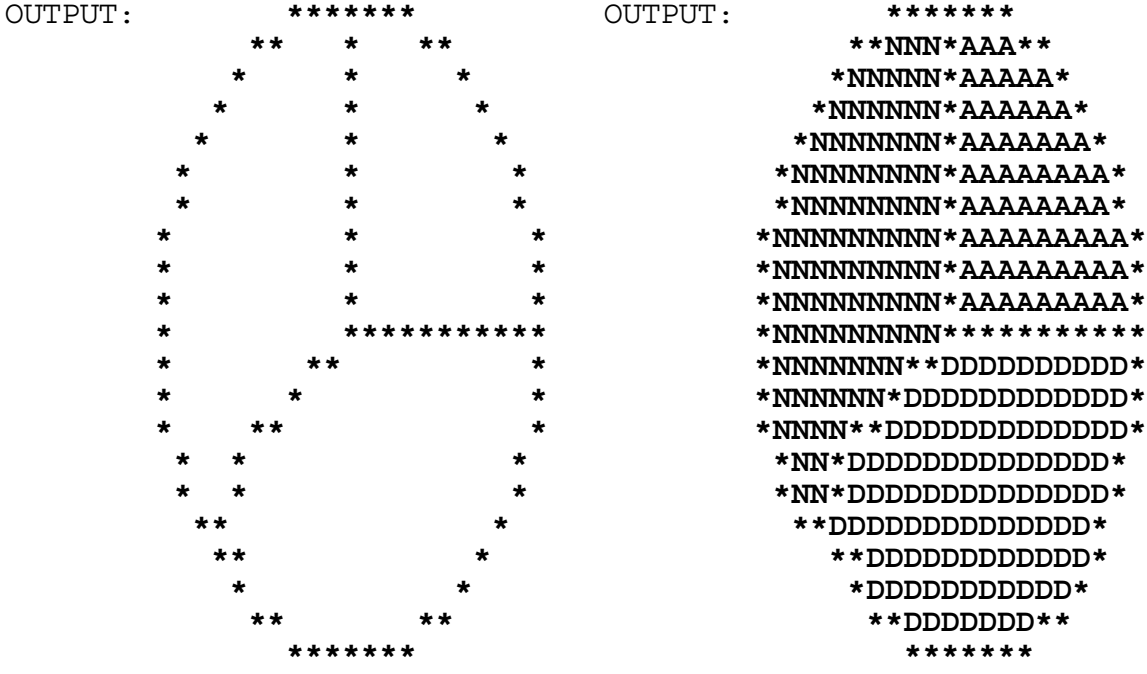


INPUT: (press any key)

Note: Although the output should look very similar to the judging criteria, minor variations will be accepted. After pressing a key to fill the regions, all regions must be at least 90% filled. No letters may replace any of the asterisks.

INPUT: Enter 3 percentages: **25, 39, 36**

OUTPUT:



INPUT: (press any key)

3.10 INPUT: Enter numeral: **3BCDEF0123456789ABCDEF0123456789**
Enter base M: **16**
Enter base N: **4**

OUTPUT:
323303132330001020310111213202122233031323300010203101112132021

INPUT: Enter numeral: **32101230321012303210**
Enter base M: **4**
Enter base N: **8**

OUTPUT: **16215471066344**

INPUT: Enter numeral: **10110011100011110000111110000010**
Enter base M: **2**
Enter base N: **8**

OUTPUT: **26343607602**

INPUT: Enter numeral: **765**
Enter base M: **8**
Enter base N: **16**

OUTPUT: **1F5**

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '92
JUDGING CRITERIA

1.1 RUN PROGRAM:

```
OUTPUT: GGGGG   TTTTT   EEEEE
        G         T     E
        G GGG     T     EEEEE   DATA SERVICES
        G   G     T     E
        GGGGG     T     EEEEE
```

1.2 INPUT: Enter year: **1992**
OUTPUT: **GTE CORPORATION**

INPUT: Enter year: **1919**
OUTPUT: **RICHLAND CENTER TELEPHONE COMPANY**

INPUT: Enter year: **1926**
OUTPUT: **ASSOCIATED TELEPHONE UTILITIES COMPANY**

1.3 INPUT: Enter 1991 rank: **7**
Enter number of places: **6**

OUTPUT: **1**

INPUT: Enter 1991 rank: **50**
Enter number of places: **21**

OUTPUT: **29**

1.4 INPUT: Enter number of spaces: **4**

OUTPUT:
```
GTE TELEPHONE OPERATIONS
    GTE GOVERNMENT SYSTEMS
        GTE MOBILE COMMUNICATIONS
            GTE INFORMATION SERVICES
                GTE SPACENET
                    GTE AIRFONE
```

INPUT: Enter number of spaces: **1**

OUTPUT:
```
GTE TELEPHONE OPERATIONS
 GTE GOVERNMENT SYSTEMS
  GTE MOBILE COMMUNICATIONS
   GTE INFORMATION SERVICES
    GTE SPACENET
     GTE AIRFONE
```

1.5 INPUT: Enter M, Y: **11, 1982**

OUTPUT: **15 YEARS**

INPUT: Enter M, Y: **8, 1990**

OUTPUT: **22 YEARS**

1.6 INPUT: Enter title: **V.P.**
Enter name: **BOESCHENSTEIN**

OUTPUT:
```
**************************
*                        *
*  V.P. BOESCHENSTEIN    *
*                        *
**************************
```

INPUT: Enter title: **DIRECTOR**
Enter name: **WALTERS**

OUTPUT:
```
**************************
*                        *
*   DIRECTOR WALTERS     *
*                        *
**************************
```

1.7 INPUT: Enter name: **SCOTT**
Enter title: **SENIOR SYSTEMS ANALYST**
Enter group: **SERVICE ORDER DEVELOPMENT**

OUTPUT: **SCOTT IS A SENIOR SYSTEMS ANALYST WITHIN THE SERVICE ORDER DEVELOPMENT GROUP AND HAS BEEN SELECTED TO PARTICIPATE IN THE ISOP.**

INPUT: Enter name: **MIKE**
Enter title: **PROJECT LEADER**
Enter group: **BILLING SUPPORT**

OUTPUT: **MIKE IS A PROJECT LEADER WITHIN THE BILLING SUPPORT GROUP AND HAS BEEN SELECTED TO PARTICIPATE IN THE ISOP.**

1.8 INPUT: Enter amount: **1234.56**

OUTPUT: **$1234.56**

INPUT: Enter amount: **2345.67**

OUTPUT: **$2000.00**

INPUT: Enter amount: **1900.00**

OUTPUT: **$1900.00**

1.9 INPUT: Enter words: **REVENUE ACCOUNTING OFFICE**

OUTPUT: **RAO**

INPUT: Enter words: **CUSTOMER BILLING SERVICES SYSTEM**

OUTPUT: **CBSS**

1.10 INPUT: Enter number of technicians, N: **6**
Enter number of minutes, M: **10**

OUTPUT: **250 HOURS 0 MINUTES**

INPUT: Enter number of technicians, N: **10**
Enter number of minutes, M: **31**

OUTPUT: **1291 HOURS 40 MINUTES**

2.1 INPUT: Enter line: **I. SUCCESS**
Enter line: **A. OPPORTUNITY**
Enter line: **1. TEACHERS**
Enter line: **2. SCHOOL**
Enter line: **3. YOUR COUNTY**
Enter line: **B. HARD WORK**
Enter line: **1. HOURS OF PRACTICE**
Enter line: **2. SELF-MOTIVATION**
Enter line: (press the return key)

OUTPUT: **I. SUCCESS**
A. OPPORTUNITY
1. TEACHERS
2. SCHOOL
3. YOUR COUNTY
B. HARD WORK
1. HOURS OF PRACTICE
2. SELF-MOTIVATION

INPUT: Enter line: **I. SUCCESS**
Enter line: **A. VISION**
Enter line: **B. PLAN**
Enter line: **C. ACTION**
Enter line: (press the return key)

OUTPUT: **I. SUCCESS**
A. VISION
B. PLAN
C. ACTION

2.2 INPUT: Enter number: **36**

OUTPUT: **THIRTY-SIX**

INPUT: Enter number: **72**

OUTPUT: **SEVENTY-TWO**

INPUT: Enter number: **18**

OUTPUT: **EIGHTEEN**

INPUT: Enter number: **1**

OUTPUT: **ONE**

2.3 INPUT: Enter name: **MIKE**
Enter degree: **BS IN COMPUTER SCIENCE**

OUTPUT: **1. DEMONSTRATED INTEREST IN INFORMATION MANAGEMENT.**
2. DEMONSTRATED LEADERSHIP SKILLS.
3. STRONG GPA/PERFORMANCE HISTORY.
4. AT LEAST TWO COURSES IN ANY PROGRAMMING LANGUAGE.
5. INTERNSHIP OR WORK EXPERIENCE.
6. EFFECTIVE ORAL AND WRITTEN COMMUNICATION SKILLS.
7. CAREER DEVELOPMENT POTENTIAL.

INPUT: Select up to 7 items: **314**

OUTPUT: (screen is cleared)
MIKE
BS IN COMPUTER SCIENCE

1. DEMONSTRATED INTEREST IN INFORMATION MANAGEMENT.

2. STRONG GPA/PERFORMANCE HISTORY.

3. AT LEAST TWO COURSES IN ANY PROGRAMMING LANGUAGE.

INPUT: Enter name: **TANIA**
Enter degree: **BS IN ACCOUNTING**

OUTPUT: **1. DEMONSTRATED INTEREST IN INFORMATION MANAGEMENT**
2. DEMONSTRATED LEADERSHIP SKILLS.
3. STRONG GPA/PERFORMANCE HISTORY.
4. AT LEAST TWO COURSES IN ANY PROGRAMMING LANGUAGE.
5. INTERNSHIP OR WORK EXPERIENCE.
6. EFFECTIVE ORAL AND WRITTEN COMMUNICATION SKILLS.
7. CAREER DEVELOPMENT POTENTIAL.

INPUT: Select up to 7 items: **761**

OUTPUT: (screen is cleared)
TANIA
BS IN ACCOUNTING

1. DEMONSTRATED INTEREST IN INFORMATION MANAGEMENT.

2. EFFECTIVE ORAL AND WRITTEN COMMUNICATION SKILLS.

3. CAREER DEVELOPMENT POTENTIAL.

2.4 INPUT: Enter rating for speech value: **EXCELLENT**
Enter rating for preparation: **SATISFACTORY**
Enter rating for manner: **SHOULD IMPROVE**
Enter rating for organization: **EXCELLENT**
Enter rating for opening: **MUST IMPROVE**
Enter rating for body of speech: **ABOVE AVERAGE**
Enter rating for conclusion: **EXCELLENT**

OUTPUT: **SPEECH VALUE: 1**
PREPARATION: 3
MANNER: 4
ORGANIZATION: 1
OPENING: 5
BODY OF SPEECH: 2
CONCLUSION: 1

AVERAGE NUMERICAL RATING = 2.4
SPEECH RATING = ABOVE AVERAGE

INPUT: Enter rating for speech value: **MUST IMPROVE**
Enter rating for preparation: **SHOULD IMPROVE**
Enter rating for manner: **SATISFACTORY**
Enter rating for organization: **EXCELLENT**
Enter rating for opening: **MUST IMPROVE**
Enter rating for body of speech: **ABOVE AVERAGE**
Enter rating for conclusion: **MUST IMPROVE**

OUTPUT: **SPEECH VALUE: 5**
PREPARATION: 4
MANNER: 3
ORGANIZATION: 1
OPENING: 5
BODY OF SPEECH: 2
CONCLUSION: 5

AVERAGE NUMERICAL RATING = 3.6
SPEECH RATING = SHOULD IMPROVE

2.5 INPUT: Enter N: **38**

OUTPUT: **BE THE CUSTOMER-ORIENTED LEADER AND PROVIDER-OF-CHOICE OF QUALITY INFORMATION PRODUCTS AND SERVICES IN THE TELECOMMUNICATIONS MARKETPLACE AND SELECTED OTHER RELATED MARKETS IN SUPPORT OF GTE'S TELOPS GOALS.**

INPUT: Enter N: **24**

OUTPUT: **BE THE CUSTOMER-ORIENTED LEADER AND PROVIDER-OF-CHOICE OF QUALITY INFORMATION PRODUCTS AND SERVICES IN THE TELECOMMUNICATIONS MARKETPLACE AND SELECTED OTHER RELATED MARKETS IN SUPPORT OF GTE'S TELOPS GOALS.**

2.6 Note: Input and Output will wrap around the screen. Two spaces separate sentences; one space separates the other words.

INPUT: Enter paragraph: **WHO WILL WIN. MY TEAM! WHY! BECAUSE WE WILL. HOW WILL THEY WIN. WHAT ARE THE PRIZES. I DON'T KNOW?**

OUTPUT: **WHO WILL WIN? MY TEAM! WHY! BECAUSE WE WILL. HOW WILL THEY WIN? WHAT ARE THE PRIZES? I DON'T KNOW?**

INPUT: Enter paragraph: **WHAT ARE THE RULES. WHOM DO YOU REPRESENT. WHY DO YOU THINK YOU WILL WIN. HOW COME! WHERE WILL YOU GO.**

OUTPUT: **WHAT ARE THE RULES? WHOM DO YOU REPRESENT. WHY DO YOU THINK YOU WILL WIN? HOW COME! WHERE WILL YOU GO?**

2.7 INPUT: Enter time: **1930**
Enter day: **FRIDAY**

OUTPUT: **GRANDVILLE, JAMES, MATT, TOM**

INPUT: Enter time: **0700**
Enter day: **TUESDAY**

OUTPUT: **DAVID, JOHN, MARIE, PAULA, SHELLEY**

INPUT: Enter time: **1130**
Enter day: **SATURDAY**

OUTPUT: **NONE**

INPUT: Enter time: **2250**
Enter day: **MONDAY**

OUTPUT: **LINDA**

2.8 Note: Run this program three times for each set of INPUT below, and look for random names (replacing ??? below) without duplication among the valid names: JEFF, LIZ, LORI, MARY, PING, and possibly DARLENE and WILL (for the third set of input).

INPUT: Enter author's name: **WILL**

OUTPUT: **AUTHOR - WILL**
MODERATOR - DARLENE
READER - ???
RECORDER - ???
INSPECTOR - ???

INPUT: Enter author's name: **DARLENE**

OUTPUT: **AUTHOR - DARLENE**
MODERATOR - WILL
READER - ???
RECORDER - ???
INSPECTOR - ???

INPUT: Enter author's name: **JEFF**

OUTPUT: **AUTHOR - JEFF**
MODERATOR - ??? (DARLENE or WILL only, 50% chance)
READER - ???
RECORDER - ???
INSPECTOR - ???

2.9 INPUT: Enter two area codes: **515, 519**
Enter number of names: **9**
Enter name: **JENNIFER**
Enter name: **JACKIE**
Enter name: **BYRON**
Enter name: **ESTHER**
Enter name: **JOHN**
Enter name: **BONNIE**
Enter name: **PAM**
Enter name: **THERESA**
Enter name: **CHARLOTTE**

OUTPUT: **515 - BONNIE**
515 - BYRON
515 - CHARLOTTE
515 - ESTHER
515 - JACKIE
519 - JENNIFER
519 - JOHN
519 - PAM
519 - THERESA

INPUT: Enter two area codes: **805, 803**
Enter number of names: **4**
Enter name: **MARCELLE**
Enter name: **RICK**
Enter name: **MIKE**
Enter name: **PAM**

OUTPUT: **803 - MARCELLE**
803 - MIKE
805 - PAM
805 - RICK

2.10 INPUT: Enter handicap: **13**
Enter gross scores: **9,8,5,11,5,6,8,7,4**

OUTPUT:

HOLE #:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
PAR:	**5**	**4**	**4**	**4**	**3**	**4**	**4**	**3**	**5**
GROSS:	**9**	**8**	**5**	**11**	**5**	**6**	**8**	**7**	**4**
ADJUST:	**8**	**7**	**5**	**7**	**5**	**6**	**7**	**5**	**4**

PAR TOTAL: 36
GROSS TOTAL: 63
ADJUST TOTAL: 54
ROUND HANDICAP: 18

INPUT: Enter handicap: **6**
Enter gross scores: **7,3,7,6,7,8,7,6,7**

OUTPUT:

HOLE #:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
PAR:	**5**	**4**	**4**	**4**	**3**	**4**	**4**	**3**	**5**
GROSS:	**7**	**3**	**7**	**6**	**7**	**8**	**7**	**6**	**7**
ADJUST:	**7**	**3**	**6**	**6**	**5**	**6**	**6**	**4**	**6**

PAR TOTAL: 36
GROSS TOTAL: 58
ADJUST TOTAL: 49
ROUND HANDICAP: 13

3.1 RUN PROGRAM:

OUTPUT:

```
      G
     T T
    E   E
   D     D
  SDETGTEDS
```

Note: The above figure is displayed in the approximate center of the screen.

Use the keys: I, J, K, and M to move the triangle up, left, right, and down respectively. Move the triangle along the perimeter of the screen.

Once a valid directional key is pressed the triangle continuously shifts one column (or row) in the designated direction until either: 1) another valid directional key is pressed, causing the triangle to shift in another direction, or 2) the triangle's edge is about to go past the perimeter of the screen, in which case the triangle is to remain stationary until another directional key is pressed to send it away from (or along) the perimeter.

3.2 INPUT: Enter X: **200**

OUTPUT: **FRIDAY AUGUST 21**

INPUT: Enter X: **261**

OUTPUT: **SATURDAY OCTOBER 31**
SUNDAY NOVEMBER 1

INPUT: Enter X: **105**

OUTPUT: **SATURDAY MAY 2**
SUNDAY MAY 3

INPUT: Enter X: **100**

OUTPUT: **MONDAY APRIL 27**

3.3 INPUT: Enter name, program: **MARIE,TN10**
Enter completed, release: **Y,N**

INPUT: Enter name, program: **DERRIL,TU03**
Enter completed, release: **N,Y**

OUTPUT: **MODULE TU03 HAS BEEN RELEASED**

INPUT: Enter name, program: **DERRIL,TU01**
Enter completed, release: **Y,Y**

OUTPUT: **MODULE TU01 HAS BEEN RELEASED**

INPUT: Enter name, program: **DOUG,TT00**
Enter completed, release: **N,N**

INPUT: Enter name, program: **LARRY,TT00**
Enter completed, release: **Y,Y**

INPUT: Enter name, program: **DOUG,TN10**
Enter completed, release: **Y,Y**

OUTPUT: **MODULE TN10 HAS BEEN RELEASED**

INPUT: Enter name, program: **MIKE,TT00**
Enter completed, release: **Y,Y**

INPUT: Enter name, program: **DOUG,TT00**
Enter completed, release: **N,Y**

OUTPUT: **MODULE TT00 HAS BEEN RELEASED**

3.4 INPUT: Enter phone #: **555-6625**

OUTPUT: **55K-NOCK**

INPUT: Enter phone #: **555-7283**

OUTPUT: **555-PAVE**
555-RATE
555-SAVE

INPUT: Enter phone #: **555-6229**

OUTPUT: **55L-OBBY**

3.5 RUN PROGRAM:

```
OUTPUT: 1567204  1242
        2436051  1463
        3645021  1751
        5460231  2273
        5612704  2316
        5716420  2334
        6532471  2453
```

3.6 INPUT: Enter N: **19**

OUTPUT: **3 + 5 + 11 = 19**

INPUT: Enter N: **500**

OUTPUT: **2 + 5 + 493 = 500**

INPUT: Enter N: **32525**

OUTPUT: **3 + 13 + 32509 = 32525**

3.7 INPUT: Enter number of substitutes: **1**
Enter name: **DEAN**

OUTPUT: **1 ANDY,DAN,DEAN,DOUG,JACK,MIKE**
2 ANDY,DAN,DEAN,DOUG,JACK,YEHIA
3 ANDY,DAN,DEAN,DOUG,MIKE,YEHIA
4 ANDY,DAN,DEAN,JACK,MIKE,YEHIA
5 ANDY,DAN,DOUG,JACK,MIKE,YEHIA
6 ANDY,DEAN,DOUG,JACK,MIKE,YEHIA
7 DAN,DEAN,DOUG,JACK,MIKE,YEHIA

INPUT: Enter number of substitutes: **2**
Enter name: **TONY**
Enter name: **DEAN**

OUTPUT: **1 ANDY,DAN,DEAN,DOUG,JACK,MIKE**
2 ANDY,DAN,DEAN,DOUG,JACK,TONY
3 ANDY,DAN,DEAN,DOUG,JACK,YEHIA
4 ANDY,DAN,DEAN,DOUG,MIKE,TONY
5 ANDY,DAN,DEAN,DOUG,MIKE,YEHIA
6 ANDY,DAN,DEAN,DOUG,TONY,YEHIA
7 ANDY,DAN,DEAN,JACK,MIKE,TONY
8 ANDY,DAN,DEAN,JACK,MIKE,YEHIA
9 ANDY,DAN,DEAN,JACK,TONY,YEHIA
10 ANDY,DAN,DEAN,MIKE,TONY,YEHIA
11 ANDY,DAN,DOUG,JACK,MIKE,TONY
12 ANDY,DAN,DOUG,JACK,MIKE,YEHIA
13 ANDY,DAN,DOUG,JACK,TONY,YEHIA
14 ANDY,DAN,DOUG,MIKE,TONY,YEHIA
15 ANDY,DAN,JACK,MIKE,TONY,YEHIA
16 ANDY,DEAN,DOUG,JACK,MIKE,TONY
17 ANDY,DEAN,DOUG,JACK,MIKE,YEHIA
18 ANDY,DEAN,DOUG,JACK,TONY,YEHIA
19 ANDY,DEAN,DOUG,MIKE,TONY,YEHIA
20 ANDY,DEAN,JACK,MIKE,TONY,YEHIA
21 ANDY,DOUG,JACK,MIKE,TONY,YEHIA
22 DAN,DEAN,DOUG,JACK,MIKE,TONY
23 DAN,DEAN,DOUG,JACK,MIKE,YEHIA
24 DAN,DEAN,DOUG,JACK,TONY,YEHIA
25 DAN,DEAN,DOUG,MIKE,TONY,YEHIA
26 DAN,DEAN,JACK,MIKE,TONY,YEHIA
27 DAN,DOUG,JACK,MIKE,TONY,YEHIA
28 DEAN,DOUG,JACK,MIKE,TONY,YEHIA

Note: A key will need to be pressed after the screen is filled with the first set of OUTPUT before displaying the rest of the 28 lines of OUTPUT.

Note: INPUT/OUTPUT is continued on next page for 3.7

(Judging Criteria for 3.7 Continued)

```
 INPUT: Enter number of substitutes: 3
        Enter name: SEAN
        Enter name: PAUL
        Enter name: ROB

OUTPUT: 1 ANDY,DAN,DOUG,JACK,MIKE,PAUL
        2 ANDY,DAN,DOUG,JACK,MIKE,ROB
        3 ANDY,DAN,DOUG,JACK,MIKE,SEAN
        4 ANDY,DAN,DOUG,JACK,MIKE,YEHIA
        5 ANDY,DAN,DOUG,JACK,PAUL,ROB
        6 ANDY,DAN,DOUG,JACK,PAUL,SEAN
        7 ANDY,DAN,DOUG,JACK,PAUL,YEHIA
        8 ANDY,DAN,DOUG,JACK,ROB,SEAN
        9 ANDY,DAN,DOUG,JACK,ROB,YEHIA
       10 ANDY,DAN,DOUG,JACK,SEAN,YEHIA
       11 ANDY,DAN,DOUG,MIKE,PAUL,ROB
       12 ANDY,DAN,DOUG,MIKE,PAUL,SEAN
       13 ANDY,DAN,DOUG,MIKE,PAUL,YEHIA
       :
       :
       :
       37 ANDY,DOUG,JACK,MIKE,PAUL,SEAN
       38 ANDY,DOUG,JACK,MIKE,PAUL,YEHIA
       39 ANDY,DOUG,JACK,MIKE,ROB,SEAN
       40 ANDY,DOUG,JACK,MIKE,ROB,YEHIA
       41 ANDY,DOUG,JACK,MIKE,SEAN,YEHIA
       42 ANDY,DOUG,JACK,PAUL,ROB,SEAN
       43 ANDY,DOUG,JACK,PAUL,ROB,YEHIA
       44 ANDY,DOUG,JACK,PAUL,SEAN,YEHIA
       45 ANDY,DOUG,JACK,ROB,SEAN,YEHIA
       46 ANDY,DOUG,MIKE,PAUL,ROB,SEAN
       47 ANDY,DOUG,MIKE,PAUL,ROB,YEHIA
       48 ANDY,DOUG,MIKE,PAUL,SEAN,YEHIA
       :
       :
       76 DAN,JACK,PAUL,ROB,SEAN,YEHIA
       77 DAN,MIKE,PAUL,ROB,SEAN,YEHIA
       78 DOUG,JACK,MIKE,PAUL,ROB,SEAN
       79 DOUG,JACK,MIKE,PAUL,ROB,YEHIA
       80 DOUG,JACK,MIKE,PAUL,SEAN,YEHIA
       81 DOUG,JACK,MIKE,ROB,SEAN,YEHIA
       82 DOUG,JACK,PAUL,ROB,SEAN,YEHIA
       83 DOUG,MIKE,PAUL,ROB,SEAN,YEHIA
       84 JACK,MIKE,PAUL,ROB,SEAN,YEHIA
```

Note: lines 14 through 36 and lines 49 through 75 are not shown. For judging purposes, spot check only those lines shown. The actual program will have 84 lines of output, allowing a key to be pressed after a screen full of lines has been displayed.

3.8 INPUT: Enter month of bill: **2**
Enter cycle number: **10**
Enter number of days: **15**
Enter holiday MM, DD: **1, 1**
Enter holiday MM, DD: **3, 17**
Enter holiday MM, DD: **0, 0**

OUTPUT: **BILL DATE: FRIDAY FEBRUARY 28**
DUE DATE: MONDAY MARCH 16

INPUT: Enter month of bill: **12**
Enter cycle number: **3**
Enter number of days: **18**
Enter holiday MM, DD: **11, 26**
Enter holiday MM, DD: **12, 25**
Enter holiday MM, DD: **0, 0**

OUTPUT: **BILL DATE: MONDAY DECEMBER 7**
DUE DATE: MONDAY DECEMBER 28

3.9 INPUT: Enter number of sides: **8**
Enter movement: **L3**
Enter movement: **U10**
Enter movement: **R5**
Enter movement: **U7**
Enter movement: **R3**
Enter movement: **D10**
Enter movement: **L5**
Enter movement: **D7**

OUTPUT: **AREA = 66 SQUARE FEET**

INPUT: Enter number of sides: **10**
Enter movement: **R5**
Enter movement: **D12**
Enter movement: **L5**
Enter movement: **U2**
Enter movement: **L2**
Enter movement: **D2**
Enter movement: **L6**
Enter movement: **U5**
Enter movement: **R8**
Enter movement: **U7**

OUTPUT: **AREA = 96 SQUARE FEET**

3.10 INPUT: Enter colors on top: **G,B,R,B,W,B,Y,G,R**
Enter colors on front: **W,G,W,Y,Y,Y,Y,O,Y**
Enter colors on right: **G,W,B,Y,O,R,R,G,G**
Enter colors on back: **Y,Y,O,W,G,G,R,B,O**
Enter colors on left: **O,B,O,O,R,O,W,W,B**
Enter colors on bottom: **W,O,W,R,B,R,G,R,B**

OUTPUT: **NUMBER OF EDGE PIECES HAVING SAME COLOR: 4**

INPUT: Enter colors on top: **B,Y,B,O,G,R,R,W,O**
Enter colors on front: **Y,W,B,B,O,O,R,G,G**
Enter colors on right: **G,B,R,B,G,W,O,G,O**
Enter colors on back: **G,B,O,Y,R,W,Y,W,W**
Enter colors on left: **Y,Y,B,O,Y,G,Y,Y,W**
Enter colors on bottom: **G,R,W,R,B,O,R,W,R**

OUTPUT: **COLORS ON MIDDLE SQUARES ARE NOT UNIQUE**
NUMBER OF EDGE PIECES HAVING SAME COLOR: 2

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '93
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: **GTEDS GTEDS GTEDS GTEDS GTEDS GTEDS**
GTEDS GTEDS GTEDS GTEDS GTEDS
GTEDS GTEDS GTEDS GTEDS
GTEDS GTEDS GTEDS
GTEDS GTEDS
GTEDS

1.2 INPUT: Enter N: **20**
Enter M: **13**

OUTPUT: **287 PROGRAMMERS**

INPUT: Enter N: **13**
Enter M: **5**

OUTPUT: **190 PROGRAMMERS**

1.3 INPUT: Enter N: **25.06**

OUTPUT: **25,060,000 ACCESS LINES**

INPUT: Enter N: **1.2**

OUTPUT: **1,200,000 ACCESS LINES**

1.4 INPUT: Enter # at Tampa: **27318**
Enter # at St. Petersburg: **3009**
Enter # at Fort Myers: **1328**
Enter # at Lakeland: **855**
Enter # at Sarasota: **1845**

OUTPUT: **34355 STUDENTS**

INPUT: Enter # at Tampa: **20000**
Enter # at St. Petersburg: **3000**
Enter # at Fort Myers: **1000**
Enter # at Lakeland: **900**
Enter # at Sarasota: **1845**

OUTPUT: **26745 STUDENTS**

1.5 INPUT: Enter name: **MIKE**
Enter level: **3**
Enter desire: **NO**

OUTPUT: **MIKE IS NOT A POSSIBLE CANDIDATE FOR ISOP**

INPUT: Enter name: **MARK**
Enter level: **5**
Enter desire: **YES**

OUTPUT: **MARK IS A POSSIBLE CANDIDATE FOR ISOP**

INPUT: Enter name: **DIANA**
Enter level: **6**
Enter desire: **YES**

OUTPUT: **DIANA IS A POSSIBLE CANDIDATE FOR ISOP**

INPUT: Enter name: **LINDA**
Enter level: **6**
Enter desire: **NO**

OUTPUT: **LINDA IS NOT A POSSIBLE CANDIDATE FOR ISOP**

1.6 INPUT: Enter curriculum: **C/UNIX**

OUTPUT: **C**
UNIX
ANSI SQL
OSF/MOTIF
SHELL PROGRAMMING

INPUT: Enter curriculum: **MVS/COBOL**

OUTPUT: **COBOL**
JCL
MVS/ESA
TSO/ISPF
VSAM
ANSI SQL
DB2
IMS

1.7 INPUT: Enter N: **3**
OUTPUT: **ABC**

INPUT: Enter N: **20**
OUTPUT: **ABCDEFGHIJKLMNOPQRST**

1.8 INPUT: Enter salary: **28500**
Enter rating: **ABOVE AVERAGE**

OUTPUT: **NEW SALARY = $30495.00**

INPUT: Enter salary: **23456.78**
Enter rating: **EXCELLENT**

OUTPUT: **NEW SALARY = $25802.46**

INPUT: Enter salary: **65432.11**
Enter rating: **GOOD**

OUTPUT: **NEW SALARY = $68703.72**

1.9 INPUT: Enter order: **R**
OUTPUT: **RECORDS**

INPUT: Enter order: **OUT**
OUTPUT: **O**

INPUT: Enter order: **TO**
OUTPUT: **T**

INPUT: Enter order: **C**
OUTPUT: **CHANGE**

1.10 INPUT: Enter grade: **B**
Enter grade: **W**
Enter grade: **D**
Enter grade: **M**
Enter grade: **W**

OUTPUT: **GPA = 1.333**

INPUT: Enter grade: **A**
Enter grade: **F**
Enter grade: **I**
Enter grade: **C**
Enter grade: **A**

OUTPUT: **GPA = 2.000**

2.1 For following inputs, the user may enter 2 numbers on the second line with or without a comma. Since the program will randomly generate N numbers between X and Y inclusive, the following outputs will vary:

INPUT: Enter N: **10**
Enter X, Y: **3, 6**

Possible OUTPUT: 4 5 3 6 4 5 6 6 3 4

(Note: Verify that 10 random #s (between 3 and 6 inclusive) are displayed with a space between each, and each of the numbers 3, 4, 5, and 6 appear at least once. Run this criteria again and look for a new set of random numbers.)

INPUT: Enter N: **8**
Enter X, Y: **80, -70**

Possible OUTPUT: 7 -23 -34 0 16 -55 2 63

(Note: Verify that 8 random #s (between -70 and 80 inclusive) are displayed with a space between each. Verify that each number is displayed at most once. Verify that there are at least 2 positive numbers and at least 2 negative numbers. Run this criteria again and look for a new set of random numbers.)

2.2
INPUT: Enter N: **7**
Enter name: **MARK**
Enter title: **SA**
Enter name: **CINDY**
Enter title: **SSE**
Enter name: **BOBBY**
Enter title: **PA**
Enter name: **JIM**
Enter title: **SSE**
Enter name: **ANITA**
Enter title: **SSE**
Enter name: **JOHN**
Enter title: **SASE**
Enter name: **DAVE**
Enter title: **SASE**

OUTPUT: **DAVE - SASE**
JOHN - SASE
ANITA - SSE
CINDY - SSE
JIM - SSE
MARK - SA
BOBBY - PA

INPUT: Enter N: **2**
Enter name: **DAVE**
Enter title: **SA**
Enter name: **WILL**
Enter title: **SE**

OUTPUT: **WILL - SE**
DAVE - SA

2.3 INPUT: Enter field: **01 WS-NAME PIC X(10).**
Enter field: **01 WS-ADDRESS PIC X(56).**
Enter field: (press the **Enter** key)

OUTPUT: **01 WS-NAME PIC X(10).**
01 WS-ADDRESS PIC X(56).

INPUT: Enter field: **01 WS-NAME.**
Enter field: **05 WS-FIRST-N.**
Enter field: **10 WS-FIRST-N-1 PIC X(01).**
Enter field: **10 WS-FIRST-N-REST PIC X(14).**
Enter field: **07 WS-MIDDLE.**
Enter field: **10 WS-MIDDLE-INIT PIC X(01).**
Enter field: **05 WS-LAST-N.**
Enter field: **12 WS-LAST-N-1 PIC X(01).**
Enter field: **12 WS-LAST-N-REST PIC X(14).**
Enter field: **01 WS-ADDRESS.**
Enter field: **07 WS-STREET PIC X(20).**
Enter field: **08 WS-CITY-ST-ZIP PIC X(46).**
Enter field: (Press the **Enter** key)

OUTPUT:
```
01 WS-NAME.
    05 WS-FIRST-N.
        10 WS-FIRST-N-1 PIC X(01).
        10 WS-FIRST-N-REST PIC X(14).
    07 WS-MIDDLE.
        10 WS-MIDDLE-INIT PIC X(01).
    05 WS-LAST-N.
        12 WS-LAST-N-1 PIC X(01).
        12 WS-LAST-N-REST PIC X(14).
01 WS-ADDRESS.
    07 WS-STREET PIC X(20).
        08 WS-CITY-ST-ZIP PIC X(46).
```

2.4 INPUT: Enter word: **COMPUTER**

OUTPUT: **NUMBER = 31513162120518**
BLOCKS = 6

INPUT: Enter word: **PRINTER**

OUTPUT: **NUMBER = 161891420518**
BLOCKS = 8

INPUT: Enter word: **COMPETITION**

OUTPUT: **NUMBER = 315131652092091514**
BLOCKS = 8

2.5

```
  INPUT: Enter N: 3
         Enter #: 1231231234
         Enter #: 1234561234
         Enter #: 9876543210

 OUTPUT: 123-123-1234

         123-456-1234  TOTAL FOR NPA OF 123 = 2


         987-654-3210  TOTAL FOR NPA OF 987 = 1


  INPUT: Enter N: 8
         Enter #: 1234567890
         Enter #: 1234568907
         Enter #: 1235678901
         Enter #: 1235679012
         Enter #: 1235679999
         Enter #: 2345678901
         Enter #: 3456789012
         Enter #: 3457890123

 OUTPUT: 123-456-7890
         123-456-8907

         123-567-8901
         123-567-9012
         123-567-9999  TOTAL FOR NPA OF 123 = 5


         234-567-8901  TOTAL FOR NPA OF 234 = 1


         345-678-9012

         345-789-0123  TOTAL FOR NPA OF 345 = 2
```

2.6 INPUT: Enter product: **H**
Enter price: **7.89**
Enter product: **I**
Enter price: **6.78**
Enter product: **J**
Enter price: **5.00**
Enter product: **H**
Enter price: **7.89**
Enter product: **A**
Enter price: **1.23**
Enter product: **J**
Enter price: **5.00**
Enter product: **9**

Enter coupon: **I**
Enter discount: **0.50**
Enter coupon: **H**
Enter discount: **0.89**
Enter coupon: **J**
Enter discount: **0.55**
Enter coupon: **J**
Enter discount: **0.95**
Enter coupon: **B**
Enter discount: **0.70**
Enter coupon: **J**
Enter discount: **0.75**
Enter coupon: **9**

OUTPUT: **TOTAL = $30.70**

INPUT: Enter product: **A**
Enter price: **2.00**
Enter product: **9**

Enter coupon: **A**
Enter discount: **0.50**
Enter coupon: **A**
Enter discount: **0.65**
Enter coupon: **9**

OUTPUT: **TOTAL = $1.35**

2.7 INPUT: Enter format: **AMERICAN**
Enter date: **01-31-1993**

OUTPUT: **ISO = 1993-01-31**
EUROPEAN = 31-01-1993

INPUT: Enter format: **ISO**
Enter date: **1993-02-04**

OUTPUT: **AMERICAN = 02-04-1993**
EUROPEAN = 04-02-1993

INPUT: Enter format: **EUROPEAN**
Enter date: **29-02-1996**

OUTPUT: **ISO = 1996-02-29**
AMERICAN = 02-29-1996

2.8 INPUT: Enter sentence: **THIS IS A GOOD COMPUTER CONTEST.**

OUTPUT: **CONTEST COMPUTER GOOD A IS THIS.**

INPUT: Enter sentence: **WE WILL WIN. I THINK WE WILL.**

OUTPUT: **WIN WILL WE. WILL WE THINK I.**

2.9 Note: numbers input may be separated without a comma.

INPUT: Enter row 1: **8, 6, 4, 2**
Enter row 2: **1, 9, 5, 7**
Enter row 3: **9, 7, 5, 4**
Enter row 4: **1, 2, 3, 4**

OUTPUT: **1. SMALLEST = 1 OCCURS AT (2,1), (4,1)**
2. SMALLEST = 2 OCCURS AT (1,4), (4,2)
3. SMALLEST = 3 OCCURS AT (4,3)
4. SMALLEST = 4 OCCURS AT (1,3), (3,4), (4,4)

INPUT: Enter row 1: **9, 8, 7, 6**
Enter row 2: **2, 3, 7, 6**
Enter row 3: **4, 6, 4, 8**
Enter row 4: **8, 9, 9, 3**

OUTPUT: **1. SMALLEST = 2 OCCURS AT (2,1)**
2. SMALLEST = 3 OCCURS AT (2,2), (4,4)
3. SMALLEST = 4 OCCURS AT (3,1), (3,3)
4. SMALLEST = 6 OCCURS AT (1,4), (2,4), (3,2)

2.10 INPUT: Enter month: **11**
Enter day: **27**
Enter year: **1967**

OUTPUT: **33 DAYS**

INPUT: Enter month: **10**
Enter day: **25**
Enter year: **1992**

OUTPUT: **9132 DAYS**

INPUT: Enter month: **2**
Enter day: **21**
Enter year: **1998**

OUTPUT: **11077 DAYS**

3.1 INPUT: (**Move cursor** to middle of the screen and press **1**)

OUTPUT:
```
#
G T E D S
T       D
E   1   E
D       T
S D E T G
```

INPUT: (**Move cursor** to bottom left corner and press **4**)

OUTPUT:
```
G T E D S
T       D
E   4   E
D       T
S D E T G
#
```

INPUT: (**Move cursor** to top right corner and press **3**)

OUTPUT: **OFF THE SCREEN**
(Note: message must appear on the top line of screen)

INPUT: (**Move cursor** to bottom right corner and press **4**)

OUTPUT: **OFF THE SCREEN**
(Note: message must appear on the top line of screen)

3.2
```
INPUT: Enter value: X
       Enter symbol: =
       Enter value: 5
       Enter symbol: +
       Enter value: 9
```
OUTPUT: **X = 14**

```
INPUT: Enter value: 7
       Enter symbol: =
       Enter value: X
       Enter symbol: -
       Enter value: 20
```
OUTPUT: **X = 27**

```
INPUT: Enter value: 15
       Enter symbol: *
       Enter value: 30
       Enter symbol: =
       Enter value: X
```
OUTPUT: **X = 450**

```
INPUT: Enter value: 111
       Enter symbol: /
       Enter value: X
       Enter symbol: =
       Enter value: 37
```
OUTPUT: **X = 3**

3.3 INPUT: Enter digits: **231765**
Enter sum: **25**

OUTPUT: **3+7+6+5 = 25** or OUTPUT: **2+1+7+6+5 = 25**
2+1+7+6+5 = 25 **3+7+6+5 = 25**

INPUT: Enter digits: **1357**
Enter sum: **17**

OUTPUT: **3+5+7 = 17**

INPUT: Enter digits: **31245**
Enter sum: **12**

OUTPUT: **3+1+2+4 = 12**
3+2+5 = 12
1+4+5 = 12

(Note: The three lines may appear in a different order.)

3.4 INPUT: Enter number:
4486111541039808345623155281556563464276142307

OUTPUT: **67^25**

INPUT: Enter number:
160869384745105719903453909002800919499225145597738902647 2923

OUTPUT: **3^5 * 7^10 * 23^15 * 29 * 31^3 * 79 * 83^11**

3.5 INPUT: Enter word: **BASIC**
OUTPUT: **FIRST LETTER: (10, 6)**
LAST LETTER: (10, 2)

INPUT: Enter word: **MICROCHIP**
OUTPUT: **FIRST LETTER: (1, 11)**
LAST LETTER: (9, 11)

INPUT: Enter word: **COBOL**
OUTPUT: **FIRST LETTER: (10, 2)**
LAST LETTER: (6, 2)

INPUT: Enter word: **DATA**
OUTPUT: **FIRST LETTER: (1, 1)**
LAST LETTER: (1, 4)

```
3.6  INPUT: Enter equation 1: X<1
            Enter logical op: AND
            Enter equation 2: X>1
     OUTPUT: NO SOLUTION

      INPUT: Enter equation 1: X<6
            Enter logical op: AND
            Enter equation 2: X>0
     OUTPUT: 1,2,3,4,5

      INPUT: Enter equation 1: X<1
            Enter logical op: OR
            Enter equation 2: X>5
     OUTPUT: ...-2,-1,0   6,7,8...

       INPUT: Enter equation 1: X>5
            Enter logical op: OR
            Enter equation 2: X<8
     OUTPUT: ALL INTEGERS

      INPUT: Enter equation 1: X<3
            Enter logical op: AND
            Enter equation 2: X<0
     OUTPUT: ...-3,-2,-1

      INPUT: Enter equation 1: X<9
            Enter logical op: AND
            Enter equation 2: X>0
     OUTPUT: 1,2,3...6,7,8
```

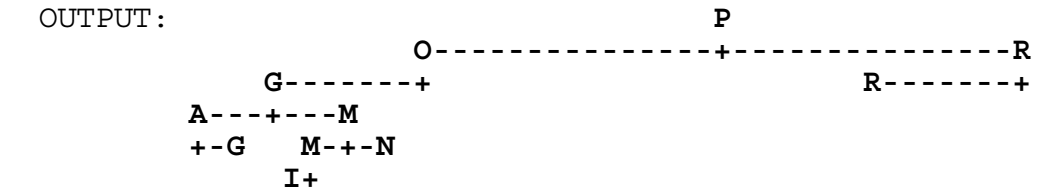
```
3.7  INPUT: Enter Mat1 (1,1): AB      Enter Mat2 (1,1): FE
            Enter Mat1 (1,2): CD      Enter Mat2 (1,2): 8
            Enter Mat1 (1,3): EF      Enter Mat2 (1,3): 9
            Enter Mat1 (2,1): 1A      Enter Mat2 (2,1): 10
            Enter Mat1 (2,2): 2B      Enter Mat2 (2,2): A
            Enter Mat1 (2,3): 3C      Enter Mat2 (2,3): B
            Enter Mat1 (3,1): 4D      Enter Mat2 (3,1): FF
            Enter Mat1 (3,2): 5E      Enter Mat2 (3,2): AA
            Enter Mat1 (3,3): 6F      Enter Mat2 (3,3): BB

     OUTPUT: SUM =   1A9    D5    F8
                      2A    35    47
                     14C   108   12A

             PRODUCT = 1A48B  AC10  BD67
                        5840  2A56  2E97
                        C0D7  4FCA  57D4
```

3.8 RUN PROGRAM:

OUTPUT:
```
149 + 257 + 863 = 1269
149 + 263 + 857 = 1269
239 + 587 + 641 = 1467
241 + 367 + 859 = 1467
257 + 419 + 683 = 1359
263 + 419 + 587 = 1269
283 + 457 + 619 = 1359
```

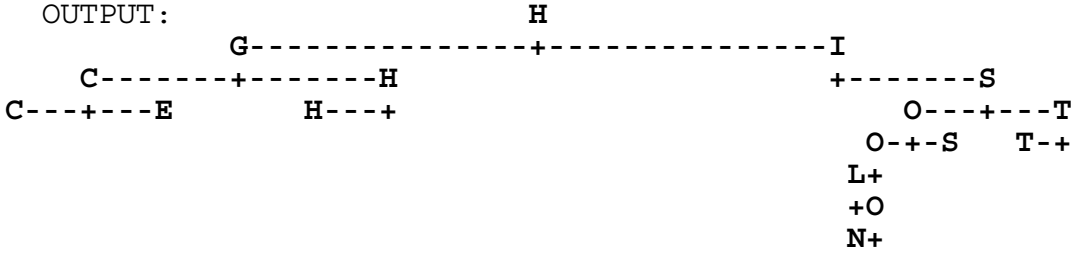
3.9 INPUT: Enter word(s): **PROGRAMMING**

OUTPUT:



INPUT: Enter word(s): **HIGH SCHOOL CONTEST**

OUTPUT:



3.10 RUN PROGRAM:

OUTPUT:
```
MINIMUM VALUE: F(X) = 0.368 OCCURS WHEN K = 0.066
MAXIMUM VALUE: F(X) = 2.7   OCCURS WHEN K = 1.44467
```

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '94
JUDGING CRITERIA

1.1 RUN PROGRAM:

OUTPUT: (The following lines are displayed, each beginning at the left most column of the screen):

FHSCC '94 IS SPONSORED BY:

GTEDS GTEDS GTEDS GTEDS GTEDS
GTEDS GTEDS GTEDS GTEDS GTEDS
GTEDS GTEDS GTEDS GTEDS GTEDS
GTEDS GTEDS GTEDS GTEDS GTEDS

USF CENTER FOR EXCELLENCE
USF CENTER FOR EXCELLENCE
USF CENTER FOR EXCELLENCE
USF CENTER FOR EXCELLENCE

FLORIDA DEPARTMENT OF EDUCATION
FLORIDA DEPARTMENT OF EDUCATION
FLORIDA DEPARTMENT OF EDUCATION
FLORIDA DEPARTMENT OF EDUCATION

1.2 INPUT: Entrance requirement: **PASSED**
Plans to accept or reject offer: **REJECT**

OUTPUT: **APPLICANT WILL NOT BE HIRED**

INPUT: Entrance requirement: **PASSED**
Plans to accept or reject offer: **ACCEPT**

OUTPUT: **APPLICANT WILL BE HIRED**

INPUT: Entrance requirement: **FAILED**
Plans to accept or reject offer: **ACCEPT**

OUTPUT: **APPLICANT WILL NOT BE HIRED**

1.3 INPUT: Enter current number: **130000**
Enter number hiring: **4321**
Enter number leaving: **5678**

OUTPUT: **128643 EMPLOYEES**

1.4 INPUT: Enter number of accounts: **2 MILLION**
Enter number of accounts: **2.1 MILLION**
Enter number of accounts: **2.4 MILLION**
Enter number of accounts: **1.5 MILLION**
Enter number of accounts: **1 MILLION**
Enter number of accounts: **-999**

OUTPUT: **9 MILLION ACCOUNTS CONVERTED TO CBSS**

INPUT: Enter number of accounts: **3.1 MILLION**
Enter number of accounts: **0.4 MILLION**
Enter number of accounts: **0.3 MILLION**
Enter number of accounts: **4 MILLION**
Enter number of accounts: **-999**

OUTPUT: **7.8 MILLION ACCOUNTS CONVERTED TO CBSS**

1.5 INPUT: Enter hours, rate: **40, 7.50**

OUTPUT: **GROSS WAGES ARE $300.00**

INPUT: Enter hours, rate: **50, 6.10**

OUTPUT: **GROSS WAGES ARE $335.50**

1.6 INPUT: Enter number of area codes: **4**
Enter area code: **912**
Enter area code: **706**
Enter area code: **208**
Enter area code: **404**

OUTPUT: **TOTAL NUMBER OF ACCOUNTS BEING SOLD = 339321**

INPUT: Enter number of area codes: **2**
Enter area code: **605**
Enter area code: **706**

OUTPUT: **TOTAL NUMBER OF ACCOUNTS BEING SOLD = 183776**

1.7 INPUT: Enter cost $: **76**
Enter phase: **DESIGN**

OUTPUT: **COST IS $380 TO FIX PROBLEM IN DESIGN PHASE**

INPUT: Enter cost $: **66**
Enter phase: **SYSTEM TEST**

OUTPUT: **COST IS $1320 TO FIX PROBLEM IN SYSTEM TEST PHASE**

INPUT: Enter cost $: **99**
Enter phase: **CODING**

OUTPUT: **COST IS $990 TO FIX PROBLEM IN CODING PHASE**

1.8 INPUT: Enter logical record length: **2348**

OUTPUT: **BLOCKSIZE = 21132 BYTES**

INPUT: Enter logical record length: **600**

OUTPUT: **BLOCKSIZE = 23400 BYTES**

1.9 INPUT: Enter kilowatt hours: **15**

OUTPUT: **THE CUSTOMER'S ELECTRIC BILL IS $92.38**

INPUT: Enter kilowatt hours: **7.5**

OUTPUT: **THE CUSTOMER'S ELECTRIC BILL IS $40.47**

INPUT: Enter kilowatt hours: **99.5**

OUTPUT: **THE CUSTOMER'S ELECTRIC BILL IS $637.77**

1.10 INPUT: Enter row: **1, 2, 3, 4, 5**
Enter row: **2, 3, 5, 6, 8**
Enter row: **3, 5, 0, 7, 0**
Enter row: **4, 6, 7, 4, 1**
Enter row: **5, 8, 0, 1, 5**

OUTPUT: **MATRIX IS SYMMETRIC**

INPUT: Enter row: **1, 2, 3, 4, 5**
Enter row: **2, 3, 4, 5, 6**
Enter row: **3, 4, 5, 6, 7**
Enter row: **4, 5, 8, 9, 1**
Enter row: **5, 6, 7, 1, 2**

OUTPUT: **MATRIX IS NOT SYMMETRIC**

2.1 INPUT: Enter jobs/CK: **OA OC OJ CK OE CK OO ON CK SG SK CK**

OUTPUT: **OA**
OC
OJ
EVERYTHING OK?
INPUT: **N**
OUTPUT: **OA**
OC
OJ
EVERYTHING OK?
INPUT: **Y**
OUTPUT: **OE**
EVERYTHING OK?
INPUT: **N**
OUTPUT: **OE**
EVERYTHING OK?
INPUT: **N**
OUTPUT: **OE**
EVERYTHING OK?
INPUT: **Y**
OUTPUT: **OO**
ON
EVERYTHING OK?
INPUT: **Y**
OUTPUT: **SG**
SK
EVERYTHING OK?
INPUT: **Y**
OUTPUT: (program ends)

2.2 RUN PROGRAM:
OUTPUT: (The screen is cleared and a randomly chosen letter is displayed in random locations until a key is pressed, displaying the letters slowly (about 10 per second))
INPUT: Enter letter: **R**
OUTPUT: (The program clears the screen and continuously displays the letter **R** in random locations until a key is pressed)
INPUT: Enter letter: (press space bar)
OUTPUT: (The program clears the screen and displays a randomly chosen letter in random locations until a key is pressed)
INPUT: Enter letter: **Y**
OUTPUT: (The program clears the screen and continuously displays the letter **Y** in random locations until a key is pressed)
INPUT: Enter letter: (press space bar)
OUTPUT: (The program clears the screen and displays a randomly chosen letter in random locations until a key is pressed)
INPUT: Enter letter: **3**
OUTPUT: (program terminates)

2.3 INPUT: Enter letters: **TSADHE RESH ALEPH HE**

OUTPUT: **H)RTS**

INPUT: Enter letters: **DALETH BETH AYIN TAW**

OUTPUT: **T(BD**

INPUT: Enter letters: **NUN HETH TETH**

OUTPUT: **TCHN**

2.4 INPUT: Enter account number: **123456789**

OUTPUT: **1234567890**

INPUT: Enter account number: **2309849123**

OUTPUT: **ERROR - INCORRECT LENGTH**

INPUT: Enter account number: **9876543**

OUTPUT: **98765431**

INPUT: Enter account number: **98765432A**

OUTPUT: **ERROR - NON-NUMERIC**

INPUT: Enter account number: **123ABC**

OUTPUT: **ERROR - INCORRECT LENGTH**
ERROR - NON-NUMERIC

2.5 INPUT: Enter last page number: **2109**
Enter M: **13**

OUTPUT: **0 APPEARS 572 TIMES**
1 APPEARS 1504 TIMES
2 APPEARS 674 TIMES
3 APPEARS 573 TIMES
4 APPEARS 574 TIMES
5 APPEARS 572 TIMES
6 APPEARS 573 TIMES
7 APPEARS 574 TIMES
8 APPEARS 573 TIMES
9 APPEARS 574 TIMES

DIGIT(S) APPEARING THE MOST: 1
DIGIT(S) APPEARING THE LEAST: 0 5

2.6 INPUT: Enter coefficients A, B, C: **2, 10, -12**

OUTPUT: **THE ROOTS ARE REAL**
THE ROOTS ARE 1 AND -6

INPUT: Enter coefficients A, B, C: **1, 6, 9**

OUTPUT: **THE ROOTS ARE REAL**
THE ONLY ROOT IS -3

INPUT: Enter coefficients A, B, C: **3, 12, 24**

OUTPUT: **THE ROOTS ARE COMPLEX**
THE ROOTS ARE -2 + 2I AND -2 - 2I

2.7 INPUT: Enter seed used last: **420001233**

OUTPUT: **4243000123**
4253000126
4263000129
4273000121
4283000124
4293000127
4204000126
4214000129
4224000121
4234000124
4244000127
4264000122
4274000125
4284000128
4294000120

2.8 INPUT: Enter speed, distance: **632.1, 0**
Enter time: **03:05C**

OUTPUT: **DISTANCE = 1949.0 MILES**

INPUT: Enter speed, distance: **120.9, 59.9**
Enter time: **0**

OUTPUT: **TIME = 0.50 HOURS**

INPUT: Enter speed, distance: **0, 999.9**
Enter time: **70M**

OUTPUT: **SPEED = 857.1 MPH**

INPUT: Enter speed, distance: **0, 432.0**
Enter time: **7.2H**

OUTPUT: **SPEED = 60.0 MPH**

2.9 INPUT: Enter reported date: **11/07/95**
Enter reported time: **09:05**
Enter cleared date: **11/09/95**
Enter cleared time: **19:15**

OUTPUT: **RESPONSE TIME WAS 1555 MINUTES**

INPUT: Enter reported date: **08/03/94**
Enter reported time: **05:35**
Enter cleared date: **08/04/94**
Enter cleared time: **14:25**

OUTPUT: **RESPONSE TIME WAS 925 MINUTES**

INPUT: Enter reported date: **02/05/94**
Enter reported time: **23:59**
Enter cleared date: **02/16/94**
Enter cleared time: **00:12**

OUTPUT: **RESPONSE TIME WAS 5400 MINUTES**

2.10 INPUT: Enter originating number: **8135558530**
Enter number called: **4075551234**
Handicapped person?: **NO**
Enter length of call: **8**
Enter cost of call $: **11.44**

OUTPUT: **THE PLAN A CHARGE WOULD BE $9.72**
THE PLAN C CHARGE WOULD BE $10.04
THIS PERSON WOULD RECEIVE PLAN A

INPUT: Enter originating number: **4075558530**
Enter number called: **4075551212**
Handicapped person?: **NO**
Enter length of call: **10**
Enter cost of call $: **1.23**

OUTPUT: **THIS PERSON DOES NOT QUALIFY FOR ANY PLANS**

INPUT: Enter originating number: **8135558530**
Enter number called: **4075551212**
Handicapped person?: **YES**
Enter length of call: **4**
Enter cost of call $: **2.34**

OUTPUT: **THE PLAN B CHARGE WOULD BE $2.11**
THE PLAN C CHARGE WOULD BE $2.05
THIS PERSON WOULD RECEIVE PLAN C

3.1 INPUT: Enter transliteration: **PHILADELPHIA**

OUTPUT: **PHI IOTA LAMBDA ALPHA DELTA EPSILON LAMBDA PHI IOTA ALPHA**
NUMERICAL SUM = 1091

INPUT: Enter transliteration: **EKPSUCHO**

OUTPUT: **EPSILON KAPPA PSI UPSILON CHI OMEGA**
NUMERICAL SUM = 2525

INPUT: Enter transliteration: **CHTHES**

OUTPUT: **CHI THETA EPSILON SIGMA**
NUMERICAL SUM = 814

INPUT: Enter transliteration: **PHOTIZO**

OUTPUT: **PHI OMEGA TAU IOTA ZETA OMEGA**
NUMERICAL SUM = 2417

3.2 INPUT: Enter starting position: **X,7**
Enter direction: **S**
OUTPUT: **TAXI LOCATION IS X,8**
INPUT: Enter direction: **S**
OUTPUT: **LOCATION IS OUTSIDE CITY LIMITS**
INPUT: Enter direction: **W**
OUTPUT: **TAXI LOCATION IS W,8**
INPUT: Enter direction: **W**
OUTPUT: **TAXI LOCATION IS V,8**
INPUT: Enter direction: **W**
OUTPUT: **LOCATION IS TOO FAR WEST**
INPUT: Enter direction: **N**
OUTPUT: **TAXI LOCATION IS V,7**
INPUT: Enter direction: **N**
OUTPUT: **TAXI LOCATION IS V,6**
INPUT: Enter direction: **N**
OUTPUT: **TAXI LOCATION IS V,5**
INPUT: Enter direction: **N**
OUTPUT: **LOCATION IS TOO FAR NORTH**
INPUT: Enter direction: **Q**
OUTPUT: (program terminates)

INPUT: Enter starting position: **Y,1**
Enter direction: **N**
OUTPUT: **LOCATION IS OUTSIDE CITY LIMITS**
INPUT: Enter direction: **E**
OUTPUT: **TAXI LOCATION IS Z,1**
INPUT: Enter direction: **Q**
OUTPUT: (program terminates)

3.3 INPUT: Enter number of words: **5**
Enter word: **REACT**
Enter word: **MASTER**
Enter word: **CRATE**
Enter word: **STREAM**
Enter word: **PEACH**

OUTPUT: **ANAGRAMS: CRATE, REACT**
MASTER, STREAM

INPUT: Enter number of words: **9**
Enter word: **PEACH**
Enter word: **RESTING**
Enter word: **SHORE**
Enter word: **HORSE**
Enter word: **MANGER**
Enter word: **STINGER**
Enter word: **CHEAP**
Enter word: **GERMAN**
Enter word: **MANAGER**

OUTPUT: **ANAGRAMS: CHEAP, PEACH**
GERMAN, MANGER
HORSE, SHORE
RESTING, STINGER

INPUT: Enter number of words: **2**
Enter word: **BEARD**
Enter word: **BEAR**

OUTPUT: **NO ANAGRAMS IN LIST**

3.4 INPUT: Enter amount of money: **15**

OUTPUT: **TAKE 1 2 3 9 AND DISPERSE 8 DOLLARS TO MAKE 2 3 9 1**
TAKE 1 2 4 8 AND DISPERSE 7 DOLLARS TO MAKE 2 4 8 1
TAKE 1 2 5 7 AND DISPERSE 6 DOLLARS TO MAKE 2 5 7 1
TAKE 1 3 4 7 AND DISPERSE 6 DOLLARS TO MAKE 3 4 7 1
TAKE 1 3 5 6 AND DISPERSE 5 DOLLARS TO MAKE 3 5 6 1
TAKE 2 3 4 6 AND DISPERSE 4 DOLLARS TO MAKE 3 4 6 2
TOTAL NUMBER OF SOLUTIONS = 6

INPUT: Enter amount of money: **19**

OUTPUT: **TAKE 1 2 3 13 AND DISPERSE 12 DOLLARS TO MAKE 2 3 13 1**
TAKE 1 2 4 12 AND DISPERSE 11 DOLLARS TO MAKE 2 4 12 1
TAKE 1 2 5 11 AND DISPERSE 10 DOLLARS TO MAKE 2 5 11 1
TAKE 1 2 6 10 AND DISPERSE 9 DOLLARS TO MAKE 2 6 10 1
TAKE 1 2 7 9 AND DISPERSE 8 DOLLARS TO MAKE 2 7 9 1
TAKE 1 3 4 11 AND DISPERSE 10 DOLLARS TO MAKE 3 4 11 1
TAKE 1 3 5 10 AND DISPERSE 9 DOLLARS TO MAKE 3 5 10 1
TAKE 1 3 6 9 AND DISPERSE 8 DOLLARS TO MAKE 3 6 9 1
TAKE 1 3 7 8 AND DISPERSE 7 DOLLARS TO MAKE 3 7 8 1
TAKE 1 4 5 9 AND DISPERSE 8 DOLLARS TO MAKE 4 5 9 1
TAKE 1 4 6 8 AND DISPERSE 7 DOLLARS TO MAKE 4 6 8 1
TAKE 1 5 6 7 AND DISPERSE 6 DOLLARS TO MAKE 5 6 7 1
TAKE 2 3 4 10 AND DISPERSE 8 DOLLARS TO MAKE 3 4 10 2
TAKE 2 3 5 9 AND DISPERSE 7 DOLLARS TO MAKE 3 5 9 2
TAKE 2 3 6 8 AND DISPERSE 6 DOLLARS TO MAKE 3 6 8 2
TAKE 2 4 5 8 AND DISPERSE 6 DOLLARS TO MAKE 4 5 8 2
TAKE 2 4 6 7 AND DISPERSE 5 DOLLARS TO MAKE 4 6 7 2
TAKE 3 4 5 7 AND DISPERSE 4 DOLLARS TO MAKE 4 5 7 3
TOTAL NUMBER OF SOLUTIONS = 18

3.5 INPUT: Enter Gregorian or Julian: **GREGORIAN**
Enter date: **12/31/95**
OUTPUT: **JULIAN DATE = 95365**

INPUT: Enter Gregorian or Julian: **GREGORIAN**
Enter date: **03/31/92**
OUTPUT: **JULIAN DATE = 92091**

INPUT: Enter Gregorian or Julian: **JULIAN**
Enter date: **94334**
OUTPUT: **GREGORIAN DATE = 11/30/94**

INPUT: Enter Gregorian or Julian: **JULIAN**
Enter date: **96023**
OUTPUT: **GREGORIAN DATE = 01/23/96**

3.6 INPUT: Enter base of first number: **4**
Enter number: **32103210**
Enter base of output: **16**

OUTPUT: **4E4E**

INPUT: Enter base of first number: **16**
Enter number: **5BCDEF**
Enter base of output: **10**

OUTPUT: **5946106**

INPUT: Enter base of first number: **12**
Enter number: **5B43**
Enter base of output: **14**

OUTPUT: **D5A3**

3.7 INPUT: Enter seed X(0): **8765**

OUTPUT: **1000TH NUMBER = 130877**
2000TH NUMBER = 270865
3000TH NUMBER = 403565
4000TH NUMBER = 536721
5000TH NUMBER = 657665
6000TH NUMBER = 780405
7000TH NUMBER = 914329
8000TH NUMBER = 1048317

3.8 INPUT: Enter N: **99**
Enter radius: **10**

OUTPUT: **4188.790204786390984616857844372670512262892532500141**
094633259456410421875048278664837379767122822757309

INPUT: Enter N: **89**
Enter radius: **100**

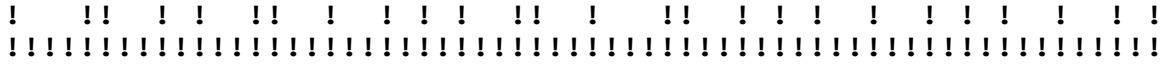
OUTPUT: **4188790.204786390984616857844372670512262892532500141**
094633259456410421875048278664837379767122822

INPUT: Enter N: **85**
Enter radius: **55**

OUTPUT: **696909.970321335800065629723857503056477738745094710974619608542060283939461157362862319058 7**

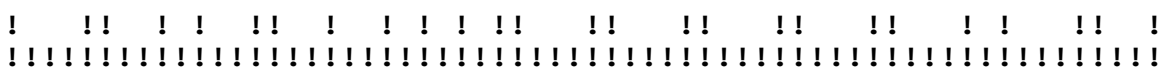
3.9 INPUT: Enter address 1: **P.O. BOX 6789**
Enter address 2: **SOMEWHERE FL 12345**

OUTPUT: **DELIVERY POINT BAR CODE = 123456789898**



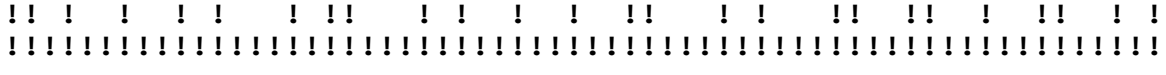
INPUT: Enter address 1: **9 GTE ST**
Enter address 2: **THISTOWN FL 12345**

OUTPUT: **DELIVERY POINT BAR CODE = 123450000096**



INPUT: Enter address 1: **5234 GTE ST**
Enter address 2: **COMPUTERVILLE FL 98765-4321**

OUTPUT: **DELIVERY POINT BAR CODE = 987654321348**



3.10 INPUT: Enter first number: **5**
Enter increment: **4**

Enter number: **9**
Enter row, col: **1, 1**

Enter number: **13**
Enter row, col: **2, 3**

OUTPUT: **9 37 17**
29 21 13
25 5 33

MAGIC NUMBER = 63

INPUT: Enter first number: **77**
Enter increment: **2**

Enter number: **89**
Enter row, col: **2, 3**

Enter number: **91**
Enter row, col: **3, 1**

OUTPUT: **83 93 79**
81 85 89
91 77 87

MAGIC NUMBER = 255