

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: <b>MAX</b>	INPUT: Enter name: <b>ANDRIA</b>
Enter type: <b>MAINFRAME</b>	Enter type: <b>MICRO</b>
Enter name: <b>MIKE</b>	Enter name: <b>LISA</b>
Enter type: <b>MICRO</b>	Enter type: <b>MAINFRAME</b>
Enter name: <b>MILTON</b>	Enter name: <b>KIM</b>
Enter type: <b>MINI</b>	Enter type: <b>MINI</b>

OUTPUT: **MIKE**  
**MILTON**  
**MAX**

OUTPUT: **ANDRIA**  
**KIM**  
**LISA**

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

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

1.9 INPUT: Enter command: <b>ADD</b>	INPUT: Enter command: <b>ADD</b>
Enter integer: <b>11</b>	Enter integer: <b>6</b>
Enter command: <b>TAKE</b>	Enter command: <b>ADD</b>
OUTPUT: <b>11</b>	Enter integer: <b>7</b>
INPUT: Enter command: <b>ADD</b>	Enter command: <b>TAKE</b>
Enter integer: <b>22</b>	OUTPUT: <b>6</b>
Enter command: <b>TAKE</b>	INPUT: Enter command: <b>ADD</b>
OUTPUT: <b>22</b>	Enter integer: <b>4</b>
INPUT: Enter command: <b>QUIT</b>	Enter command: <b>TAKE</b>
OUTPUT: (program terminates)	OUTPUT: <b>7</b>
	INPUT: Enter command: <b>TAKE</b>
	OUTPUT: <b>4</b>
	INPUT: Enter command: <b>QUIT</b>
	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**



## 2.3 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

INPUT: Enter n: 3

Enter vertex: 1,1  
Enter vertex: 5,5  
Enter vertex: -1,-4

OUTPUT: AREA = 6.00

OUTPUT: AREA = 41.5

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                    INPUT: Enter N: 6  
          Enter #: 523                    Enter #: -1.009  
          Enter #: 321                    Enter #: 54.32  
          Enter #: 899.6                   Enter #: 81.4  
          Enter #: 66.79                   Enter #: -8.8  
  Enter #: 7.3456  
  Enter #: -6.7  
  
OUTPUT: 66.79                            OUTPUT: -6.7  
          899.6                            -1.009  
          523                              -8.8  
          321                              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 cubel diagonal point1: 2, -3, 6  
          Enter cubel 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 cubel diagonal point1: 4, 5, 6  
          Enter cubel 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:  $P(Q(X)) = 64X^{**4} + -104X^{**3} + 68X^{**2} + -20X^{**1} + 2X^{**0}$   
 $Q(P(X)) = 8X^{**4} + 6X^{**3} + 4X^{**2} + 2X^{**1} + -1X^{**0}$

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:  $P(Q(X)) = -2X^{**0}$   
 $Q(P(X)) = -1X^{**0}$

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:  $P(Q(X)) = -16X^{**4} + 0X^{**3} + 16X^{**2} + 0X^{**1} + -1X^{**0}$   
 $Q(P(X)) = 4X^{**4} + 0X^{**3} + -24X^{**2} + 0X^{**1} + 34X^{**0}$