

FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '85
BASIC PROGRAM SOLUTIONS

```
'1.1
' This program will simulate a Last-In-First-Out stack.
'
INPUT "Enter command:"; A$
WHILE A$ <> "QUIT"
  IF A$ = "ADD" THEN S = S + 1: INPUT "Enter number:"; A(S)
  IF A$ = "TAKE" THEN PRINT A(S): S = S - 1
  INPUT "Enter command:"; A$
WEND
```

```
'1.2
' This program will determine which number was erased.
'
INPUT "Enter N, AV: "; N, AV
FOR I = 1 TO N: S = S + I: NEXT I
FOR I = 1 TO N
  T = S - I
  IF ABS((T / (N - 1)) - AV) < .0001 THEN
    PRINT "NUMBER ERASED WAS"; I: END
  END IF
NEXT I
```

```
'1.3
' This program will print the square root of N.
'
INPUT "Enter N, D: "; N, D: Q = SQR(N)
S = INT(Q / 10 ^ D + .5) * 10 ^ D
A$ = MID$(STR$(S), 2)
FOR I = 1 TO LEN(A$)
  T = T + VAL(MID$(A$, I, 1))
NEXT I
PRINT USING "S=####.####"; S
PRINT "SUM="; LTRIM$(STR$(T))
```

```
'1.4
' This program will simulate a time dial.
'
CLS : Y = 1985
J = 3000      'About one second on a 386-computer
WHILE Y <= 2345
  LOCATE 12, 38: PRINT Y
  IF J > 30 THEN J = J - 30
  FOR K = 1 TO J: NEXT K
  Y = Y + 1
WEND
```

'1.5

' This program will determine # of tennis games and byes.

,

```
INPUT "Enter N: "; N
WHILE N > 1
  G = INT(N / 2)
  IF G * 2 = N THEN B = 0 ELSE B = 1
  R = R + 1
  PRINT USING "ROUND #  ## GAMES"; R; G;
  IF B = 1 THEN PRINT "  1 BYE" ELSE PRINT
  TG = TG + G: BY = BY + B: N = G + B
WEND
PRINT USING "TOTAL      ## GAMES ## BYES"; TG; BY
```

'1.6

' This program will find smallest, largest, and sum of #s.

,

```
INPUT "Enter N, M: "; N, M
IF M > 999 THEN M = 999
IF N < 100 THEN N = 100
FOR I = N TO M
  NUM = I
  H = INT(NUM / 100): NUM = NUM - H * 100
  T = INT(NUM / 10):  U = NUM - T * 10
  IF NOT (H = 0 OR T = 0 OR U = 0 OR H = T OR T = U OR H = U) THEN
    S = S + I: L = I: IF S = I THEN PRINT "SMALLEST ="; S
  END IF
NEXT I
PRINT "LARGEST ="; L
PRINT "SUM ="; S
```

'1.7

' This program will print a bill for Bob's Cycle shop.

,

```
INPUT "Enter Customer name: "; N$
INPUT "Enter part#: "; P$
INPUT "Enter labor time: "; LT
WHILE P$ <> A$: READ A$, B$, C: WEND
' Print Invoice
PRINT
PRINT "CUSTOMER NAME: "; N$
PRINT "PART #: "; P$
PRINT "DESCRIPTION: "; B$
PRINT USING "PART COST:  ###.##"; C
LC = LT * 10
PRINT USING "LABOR COST: ###.##"; LC
TAX = C * .05: TAX = INT(TAX * 100 + .5) / 100
PRINT USING "5% TAX:      ###.##"; TAX
TOT = LC + C + TAX
PRINT USING "TOTAL:      ###.##"; INT(TOT * 100 + .5) / 100
```

```
DATA S193,10 INCH SPROCKET,13.95
DATA S867,30 INCH CHAIN,27.50
DATA F234,BLITZ MAG FRAME,119.00
DATA S445,COMPUTCYCLE COMPUTER,33.95
DATA C492,JET BRAKE SET,29.98
DATA J273,27 INCH WHEEL,32.00
DATA T100,27x1 INCH TIRE TUBE,12.50
```

'1.8

' This program will display labels alphabetically.

```
,
INPUT "Enter # of lines on label:"; H
S = 1: READ A$(S), B$(S)
WHILE A$(S) <> "*END*"
  L = LEN(A$(S)): I = 1
  WHILE MID$(A$(S), I, 1) <> " ": I = I + 1: WEND
  R$ = RIGHT$(A$(S), L - I): L$ = LEFT$(A$(S), I)
  C$(S) = R$ + ", " + L$
  S = S + 1
  READ A$(S), B$(S)
WEND
S = S - 1
FOR I = 1 TO S - 1
  FOR J = I + 1 TO S
    IF C$(I) > C$(J) THEN
      X$ = C$(I): C$(I) = C$(J): C$(J) = X$
      X$ = B$(I): B$(I) = B$(J): B$(J) = X$
    END IF
  NEXT J
NEXT I
FOR I = 1 TO S
  PRINT : PRINT C$(I): PRINT B$(I)
  FOR J = 1 TO H - 3: PRINT : NEXT J
NEXT I
DATA LISA SPINXS,987-6543
DATA BOB SIMON,123-4455
DATA BILL SIMON,123-4567
DATA HARRY TROUTMAN,876-2174
DATA HARRY PARKER,222-3333
DATA *END*,0
```

```
'1.9
' This program will guess secret letter in 5x5 matrix.
,
CLS : DIM A(25): S = 11
RANDOMIZE TIMER
FOR I = 1 TO 5
  FOR J = 1 TO 5
    DO: X = INT(RND(3) * 25): LOOP UNTIL A(X) = 0
    B$(I, J) = CHR$(X + 65): LOCATE I, 13 + J * 2: PRINT B$(I, J)
    A(X) = 1
  NEXT J
NEXT I
LOCATE 2, 30: PRINT "SCORE="; S
I = 0
WHILE A$ <> "Y"
  I = I + 1
  LOCATE 10, 10: PRINT "IS THE LETTER IN ROW"; I; : INPUT A$
  LOCATE 2, 30: S = S - 1: PRINT "SCORE="; S: LOCATE 10, 10
WEND: A$ = "": J = 0
WHILE A$ <> "Y"
  J = J + 1
  LOCATE 12, 10: PRINT "IS THE LETTER IN COL"; J; : INPUT A$
  LOCATE 2, 30: S = S - 1: PRINT "SCORE="; S
WEND
IF S > 0 THEN LOCATE 14, 1: PRINT "YOUR LETTER IS "; B$(I, J)
```

```

'1.10
' This program will display squares relative to cursor and #.
' Key movements: I-up, J-left, K-right, M-down
'
CLS : R = 5: C = 5
WHILE VAL(A$) = 0
  LOCATE R, C: PRINT "#": A$ = ""
  WHILE A$ = "": A$ = INKEY$: WEND
  IF VAL(A$) = 0 THEN
    LOCATE R, C: PRINT " "
    IF A$ = "I" THEN R = R - 1
    IF A$ = "M" THEN R = R + 1
    IF A$ = "J" THEN C = C - 1
    IF A$ = "K" THEN C = C + 1
  END IF
WEND
X = VAL(A$)
IF X = 1 THEN A = 1: B = 0
IF X = 2 THEN A = 1: B = -1
IF X = 3 THEN A = -1: B = -1
IF X = 4 THEN A = -1: B = 0
IF R + 5 * A > 24 OR R + 5 * A < 1 THEN
  LOCATE 21, 1: PRINT "OFF THE SCREEN": END
END IF
IF C + 9 * B > 80 OR C + 9 * B < 1 THEN
  LOCATE 21, 1: PRINT "OFF THE SCREEN": END
END IF
LOCATE R + 1 * A, C + 8 * B: PRINT "*****"
LOCATE R + 2 * A, C + 8 * B: PRINT "*      *"
LOCATE R + 3 * A, C + 8 * B: PRINT USING "*      #      *"; X
LOCATE R + 4 * A, C + 8 * B: PRINT "*      *"
LOCATE R + 5 * A, C + 8 * B: PRINT "*****"

```

'2.1

' This program will outline screen with random letters.

,

RANDOMIZE TIMER

DO

CLS

FOR I = 1 TO 11

X = INT(RND * 26): A\$ = CHR\$(65 + X)

LOCATE I, I

FOR J = I TO 80 - I: PRINT A\$; : NEXT J

FOR J = I + 1 TO 23 - I

LOCATE J, I: PRINT A\$: LOCATE J, 80 - I: PRINT A\$

NEXT J

LOCATE 23 - I, I

FOR J = I TO 80 - I: PRINT A\$; : NEXT J

B\$ = "": WHILE B\$ = "": B\$ = INKEY\$: WEND

NEXT I

LOOP UNTIL B\$ = CHR\$(27)

'2.2

' This program will print the longest sequence of letters.

,

INPUT "Enter N: "; N: DIM A\$(N)

FOR I = 1 TO N: INPUT "Enter letter: "; A\$(I): NEXT I

I = N: FOUND = 0

WHILE (I >= 2) AND NOT FOUND

FOR J = 1 TO N - I + 1

ONE = -1

FOR K = 0 TO I - 2

IF A\$(J + K) >= A\$(J + K + 1) THEN ONE = 0

NEXT K

IF ONE THEN

FOR K = 0 TO I - 1: PRINT A\$(J + K); " "; : NEXT K

PRINT : FOUND = -1

END IF

NEXT J

I = I - 1

WEND

'2.3

' This program will change the margins of a given text.

```

,
INPUT "Enter text: "; A$: A$ = A$ + " ": L = LEN(A$)
LW = 5: PRINT SPACE$(10);
FOR I = 1 TO L
  C$ = MID$(A$, I, 1)
  IF C$ <> " " THEN
    W$ = W$ + C$
  ELSE
    LL = LEN(W$)
    IF LW + LL > 30 THEN PRINT : PRINT SPACE$(5); : LW = 0
    IF LL > 0 THEN PRINT W$; " "; : LW = LW + LL + 1: W$ = ""
    IF LL = 0 AND LW > 0 THEN PRINT " "; : LW = LW + 1
  END IF
NEXT I

```

'2.4

' This program will print word with consonants alphabeitized.

```

,
INPUT "Enter word: "; A$: L = LEN(A$): V$ = "AEIOU"
DIM V$(L), C$(L), A(L)
FOR I = 1 TO L
  B$ = MID$(A$, I, 1): J = 1
  WHILE (J < 5) AND (MID$(V$, J, 1) <> B$): J = J + 1: WEND
  IF MID$(V$, J, 1) <> B$ THEN
    C = C + 1: C$(C) = B$
  ELSE
    V = V + 1: A(I) = 1: V$(V) = B$
  END IF
NEXT I
FOR I = 1 TO V - 1
  FOR J = I + 1 TO V
    IF V$(I) > V$(J) THEN X$ = V$(I): V$(I) = V$(J): V$(J) = X$
  NEXT J
NEXT I
FOR I = 1 TO C - 1
  FOR J = I + 1 TO C
    IF C$(I) > C$(J) THEN X$ = C$(I): C$(I) = C$(J): C$(J) = X$
  NEXT J
NEXT I
FOR I = 1 TO L
  IF A(I) = 1 THEN VV = VV + 1: PRINT V$(VV);
  IF A(I) = 0 THEN CC = CC + 1: PRINT C$(CC);
NEXT I

```

```
'2.5
' This program will print common letters and line up words.
,
INPUT "Enter N: "; N
FOR I = 1 TO N: INPUT "Enter word: "; A$(I): NEXT I
FOR I = 1 TO 26
  X$ = CHR$(64 + I): CO = -1: J = 1
  WHILE (J <= N) AND CO
    CO = INSTR(A$(J), X$)
    J = J + 1
  WEND
  IF CO THEN PRINT X$; " "; : FOUND = -1
NEXT I
IF NOT FOUND THEN PRINT "NO COMMON LETTERS": END
PRINT : INPUT "Choose letter: "; A$
FOR I = 1 TO N
  J = 1: WHILE MID$(A$(I), J, 1) <> A$: J = J + 1: WEND
  PRINT SPACE$(10 - J); A$(I)
NEXT I
```



```
'2.6
' This program will keep score for a double dual race.
,
CLS : DIM IN$(21)
FOR I = 1 TO 21
  PRINT "Place "; I; ":"; : INPUT IN$(I)
  IF I > 1 THEN
    J = 1
    WHILE J <= TN AND INIT$(J) <> IN$(I): J = J + 1: WEND
  END IF
  IF (INIT$(J) <> IN$(I)) OR (I = 1) THEN
    TN = TN + 1: INIT$(TN) = IN$(I)
  END IF
NEXT I
' Assert TEAM$(1, 2, 3) = 3 unique team INITIALS
FOR I = 1 TO 2
  FOR J = I + 1 TO 3
    PL = 0: T1 = 0: T2 = 0: T1PL = 0: T2PL = 0
    FOR K = 1 TO 21
      IF IN$(K) = INIT$(I) THEN
        PL = PL + 1: T1 = T1 + PL: T1PL = T1PL + 1
        TEAM1(T1PL) = PL
      END IF
      IF IN$(K) = INIT$(J) THEN
        PL = PL + 1: T2 = T2 + PL: T2PL = T2PL + 1
        TEAM2(T2PL) = PL
      END IF
    NEXT K
    T1 = T1 - TEAM1(6) - TEAM1(7)
    T2 = T2 - TEAM2(6) - TEAM2(7)
    PRINT "TEAM "; INIT$(I); ":"; T1; " POINTS"
    PRINT "TEAM "; INIT$(J); ":"; T2; " POINTS"
    IF (T1 < T2) OR (T1 = T2 AND TEAM1(6) < TEAM2(6)) THEN
      PRINT "TEAM "; INIT$(I);
    ELSE
      PRINT "TEAM "; INIT$(J);
    END IF
    PRINT " WINS!": PRINT
  NEXT J
NEXT I
```

'2.7

' This program will allow manipulation of 3x3 array of data.

```
,
DATA 10.11, 20.22, 30.33
DATA 11.1, 22.2, 33.3
DATA 10, 20, 30
FOR I = 1 TO 3: FOR J = 1 TO 3: READ A(I, J): NEXT J, I
WHILE C$ <> "C"
  CLS
  PRINT "A. EDIT OR CHANGE A VALUE"
  PRINT "B. DISPLAY THE RESULTS"
  PRINT "C. QUIT."
  INPUT "Enter option: "; C$
  IF C$ = "A" THEN
    INPUT "Enter row, col: "; ROW, COL
    INPUT "Enter number: "; A(ROW, COL)
  ELSEIF C$ = "B" THEN
    FOR I = 1 TO 3: A(I, 4) = 0: NEXT I
    FOR J = 1 TO 3: A(4, J) = 0: NEXT J: TOT = 0
    FOR I = 1 TO 3
      FOR J = 1 TO 3
        PRINT USING "###.## "; A(I, J); : TOT = TOT + A(I, J)
        A(4, J) = A(4, J) + A(I, J): A(I, 4) = A(I, 4) + A(I, J)
      NEXT J
      PRINT USING "###.##"; A(I, 4)
    NEXT I
    FOR J = 1 TO 3: PRINT USING "###.## "; A(4, J); : NEXT J
    PRINT USING "###.##"; TOT
  END IF
  IF C$ <> "C" THEN
    PRINT : PRINT "Press any key: ";
    A$ = "": WHILE A$ = "": A$ = INKEY$: WEND
  END IF
WEND
```

'2.8

' This program will print all combinations of 4 digits.

```
,
FOR A = 1 TO 8
  FOR B = A + 1 TO 9
    P = A * B
    IF P >= 10 THEN
      P$ = MID$(STR$(P), 2)
      C = VAL(MID$(P$, 1, 1)): D = VAL(MID$(P$, 2, 1))
      IF NOT (A = C OR A = D OR B = C OR B = D) THEN
        PRINT A; B; C; D, A; "X"; B; "= "; P$: S = S + 1
      END IF
    END IF
  NEXT B
NEXT A
PRINT " TOTAL ="; S
```

```
'2.9
' This program will select words, given a string w/wildcard.
'
INPUT "Enter N: "; N: DIM A$(N)
FOR I = 1 TO N: INPUT "Enter word: "; A$(I): NEXT I
DO
  INPUT "Enter string: "; A$: L = LEN(A$): W = 0
  I = INSTR(A$, "*"): IF I = 0 THEN END
  ' Asterisk is position I
  L$ = LEFT$(A$, I - 1): R$ = RIGHT$(A$, L - I)
  FOR J = 1 TO N
    IF LEFT$(A$(J), I - 1) = L$ THEN
      IF RIGHT$(A$(J), L - I) = R$ THEN
        PRINT A$(J): W = 1
      END IF
    END IF
  NEXT J
  IF W = 0 THEN PRINT "NO WORDS FOUND"
  PRINT
LOOP UNTIL I = 0
```

```
'2.10
' This program will maintain air conditioning in 3 rooms.
'
INPUT "Enter last 5-minutes: "; LM
CLS : OF = 72: CO = 65: DR = 79
PRINT " OF CO DS OFFICE COMP. DRY. MIN:SEC"
DO
  IF ((M MOD 5 = 0) AND S = 0) OR CH = 1 THEN
    PRINT O; C; D; " ";
    PRINT USING "##.#"; OF; : PRINT " ";
    PRINT USING "##.#"; CO; : PRINT " ";
    PRINT USING "##.#"; DR; : PRINT " ";
    PRINT USING "###: "; M;
    IF S > 0 THEN PRINT USING "##"; S ELSE PRINT "00"
    CH = 0
  END IF
  S = S + 15: IF S = 60 THEN M = M + 1: S = 0
  OF = OF + .1 - OFAIR
  CO = CO + .2 - COAIR
  DR = DR + .025 - DRAIR
  IF OF > 78 AND O = 0 THEN O = 1: CH = 1
  IF CO > 70 AND C = 0 THEN C = 1: CH = 1
  IF DR > 85 AND D = 0 THEN D = 1: CH = 1
  IF OF < 72 AND O = 1 THEN O = 0: CH = 1
  IF CO < 65 AND C = 1 THEN C = 0: CH = 1
  IF DR < 75 AND D = 1 THEN D = 0: CH = 1
  AIR = (O + C + D) * 2
  IF AIR = 0 THEN
    OFAIR = 0: COAIR = 0: DRAIR = 0
  ELSE
    OFAIR = O / AIR: COAIR = C / AIR: DRAIR = D / AIR
  END IF
LOOP UNTIL (M = LM) AND (S > 0)
```

```
'3.1
' This program will display the sides of a die.
'
' 6 ways to represent die (each with different top)
' DATA Top, Front, Right, Back, Left, (Bottom derived from top)
DATA 1, 5, 4, 2, 3
DATA 6, 5, 3, 2, 4
DATA 5, 1, 3, 6, 4
DATA 2, 1, 4, 6, 3
DATA 3, 5, 1, 2, 6
DATA 4, 5, 6, 2, 1
INPUT "Enter top, front: "; T, F
' Determine which data set to use (based on top #)
WHILE A <> T
  READ A
  FOR J = 1 TO 4: READ B(J): NEXT J
WEND
' Rotate sides till a side matches the front #
J = 1
WHILE B(J) <> F:
  J = J + 1
WEND
IF J = 4 THEN J = 0
R = J + 1
'Generate rest of sides, sum of opposites sides = 7
PRINT USING "TOP=# FRONT=# RIGHT=#"; T; F; B(R)
PRINT USING "BACK=# LEFT=# BOTTOM=#"; 7 - F; 7 - B(R); 7 - T
```

```
'3.2
' This program will factor a quadratic equation.
,
INPUT "Enter A, B, C: ", A, B, C
IF A < 0 THEN A = -A: B = -B: C = -C
IF A > 1 THEN
  FOR I = A TO 2 STEP -1
    IF (A MOD I = 0) AND (B MOD I = 0) AND (C MOD I = 0) THEN
      A = A / I: B = B / I: C = C / I
      PRINT LTRIM$(STR$(I));
    END IF
  NEXT I
END IF
S = B * B - 4 * A * C
IF S < 0 THEN PRINT "CANNOT BE FACTORED": END
H = INT(SQR(S) + .1): E = 2 * A
R(1) = -B + H: R(2) = -B - H
FOR K = 1 TO 2
  D = E: N = R(K)
  I = D
  WHILE (I > 0) AND ((N MOD I <> 0) OR (D MOD I <> 0))
    I = I - 1
  WEND
  N = N / I: D = D / I
  PRINT "("; : IF D > 1 THEN PRINT LTRIM$(STR$(D));
  PRINT "X";
  IF N < 0 THEN PRINT "+"; LTRIM$(STR$((-N)); ")";
  IF N > 0 THEN PRINT "-"; LTRIM$(STR$(N)); ")";
NEXT K
```

'3.3

' This program will simulate a calculator.

,

```
INPUT "Enter expression: "; A$: L = LEN(A$)
FOR I = 1 TO L
  B$ = MID$(A$, I, 1)
  IF ASC(B$) >= ASC("0") THEN
    C$ = C$ + B$
  ELSE
    J = J + 1: A(J) = VAL(C$): C$ = ""
    B(J) = INSTR("+-*/", B$)
  END IF
NEXT I
J = J + 1: A(J) = VAL(C$): K = 1
FOR I = 1 TO J - 1
  IF B(I) < 3 THEN
    B(K) = B(I): K = K + 1: A(K) = A(I + 1)
  ELSE
    IF B(I) = 3 THEN
      A(K) = A(K) * A(I + 1)
    ELSE
      A(K) = A(K) / A(I + 1)
    END IF
  END IF
NEXT I
S = A(1)
IF K > 1 THEN
  FOR I = 1 TO K - 1
    IF B(I) = 2 THEN S = S - A(I + 1) ELSE S = S + A(I + 1)
  NEXT I
END IF
PRINT USING "###.###"; S
```

'3.4

' This program will compute all digits of N factorial.

,

```
INPUT "Enter N: "; N: DIM A(3 * N)
D = 1: A(1) = 1
FOR I = 1 TO N
  FOR J = 1 TO D
    A(J) = A(J) * I + C: C = INT(A(J) / 10)
    A(J) = A(J) - 10 * C
  NEXT J
  WHILE C > 0
    CC = INT(C / 10): D = D + 1: A(D) = C - 10 * CC: C = CC
  WEND
NEXT I
FOR I = D TO 1 STEP -1: PRINT MID$(STR$(A(I)), 2); : NEXT I
```

```

'3.5
' This program will sum and subtract 2 big decimals.
,
DIM A(30), B(30), C(30), D(30): CLS
INPUT "Enter #1: "; A$: INPUT "Enter #2: "; B$
A = LEN(A$): B = LEN(B$)
FOR I = A TO 1 STEP -1
  IF MID$(A$, I, 1) = "." THEN
    X = I
  ELSE
    S = S + 1: A(S) = VAL(MID$(A$, I, 1))
  END IF
NEXT I: S = 0
FOR I = B TO 1 STEP -1
  IF MID$(B$, I, 1) = "." THEN
    Y = I
  ELSE
    S = S + 1: B(S) = VAL(MID$(B$, I, 1))
  END IF
NEXT I
' Align decimal point
G = A - X: H = B - Y
IF G > H THEN L = G ELSE L = H
Z = G - H
IF Z > 0 THEN
' Second # is smaller, so place leading 0s and align decimal
  FOR I = B - 1 TO 1 STEP -1
    B(I + Z) = B(I): B(I) = 0
  NEXT I
  B = B + Z
ELSEIF Z < 0 THEN
' First # is smaller, so place leading 0s and align decimal
  FOR I = A - 1 TO 1 STEP -1
    A(I - Z) = A(I): A(I) = 0
  NEXT I
  A = A - Z
END IF
IF A > B THEN Y = A - 1 ELSE Y = B - 1
' Add and subtract
FOR I = 1 TO Y
  C(I) = A(I) + B(I) + C: C = INT(C(I) / 10)
  C(I) = C(I) - C * 10
  D(I) = A(I) - B(I) - D
  IF D(I) < 0 THEN D = 1 ELSE D = 0
  D(I) = D(I) + D * 10
NEXT I
PRINT "SUM = "; : IF C > 0 THEN PRINT LTRIM$(STR$(C));
FOR I = Y TO 1 STEP -1
  IF I = L THEN PRINT ".";
  PRINT LTRIM$(STR$(C(I)));
NEXT I
PRINT : PRINT "DIFFERENCE = ";
FOR I = Y TO 1 STEP -1
  IF I = L THEN PRINT ".";
  PRINT LTRIM$(STR$(D(I)));

```

NEXT I

'3.6

' This program will control the movements of a snake.

,

CLS : DIM A(25, 81)

V = 12: H = 8: LOCATE V, H

FOR I = 8 TO 32

PRINT "*"; : A(V, I) = 1

A\$ = A\$ + "12": B\$ = B\$ + RIGHT\$(STR\$(I), 2)

NEXT I

WHILE D\$ = "": D\$ = INKEY\$: WEND: C\$ = D\$

DO UNTIL C\$ = CHR\$(27)

FOR I = 1 TO 100

D\$ = INKEY\$: IF D\$ <> "" THEN C\$ = D\$

NEXT I

IF C\$ = "I" THEN V = V - 1

IF C\$ = "M" THEN V = V + 1

IF C\$ = "J" THEN H = H - 1

IF C\$ = "K" THEN H = H + 1

IF A(V, H) OR V = 0 OR V = 25 OR H = 0 OR H = 81 THEN END

A(V, H) = 1: LOCATE V, H: PRINT "*"

X = VAL(RIGHT\$(A\$, 2)): Y = VAL(RIGHT\$(B\$, 2))

LOCATE X, Y: PRINT " "

A(X, Y) = 0

A\$ = LEFT\$(A\$, 24 * 2): B\$ = LEFT\$(B\$, 24 * 2)

A\$ = RIGHT\$(STR\$(V), 2) + A\$

B\$ = RIGHT\$(STR\$(H), 2) + B\$

LOOP


```
'3.7
' This program will print 3 permutations of a word.
'
INPUT "Enter word: "; A$: INPUT "Enter K: "; KK: L = LEN(A$)
FOR I = 1 TO L: A$(I) = MID$(A$, I, 1): NEXT I
' Alphabetize letters
FOR I = 1 TO L - 1
  FOR J = I + 1 TO L
    IF A$(I) > A$(J) THEN X$ = A$(I): A$(I) = A$(J): A$(J) = X$
  NEXT J
NEXT I
' Produce factorials F(I) = (I-1)!
FOR I = 1 TO L
  F = 1
  FOR J = 1 TO I - 1: F = F * J: NEXT J
  F(I) = F
NEXT I
FOR T = 1 TO 3
  K = KK * T - 1
  ' Generate Kth permutation
  FOR I = L TO 1 STEP -1
    X = INT(K / F(I))
    FOR J = 1 TO L
      IF A(J) = 0 THEN
        S = S + 1: IF S > X THEN A(J) = 1: PRINT A$(J); : J = L
      END IF
    NEXT J
    S = 0: K = K - F(I) * X
  NEXT I
  FOR I = 1 TO L: A(I) = 0: NEXT I
  PRINT " ";
NEXT T
```

'3.8

' This program will display N pennies on board.

```

'
INPUT "Enter N: "; N: DIM A(N): CLS
PRINT "TOTAL ="; N
IF N = 8 THEN SP = 1 ' 8 and 14 are special cases
J = N MOD 2: J = 2 - J: S = J
IF N = 14 THEN S = J + 2
PRINT " ";
FOR I = 1 TO N: PRINT USING "##"; I MOD 10; : NEXT I
PRINT
FOR I = 1 TO N: PRINT USING "#"; I MOD 10: NEXT I
FOR I = 1 TO N
  A(I) = S
  IF N = 14 AND I = 14 THEN S = 2: A(I) = S
  LOCATE 2 + I, 2 * S + 1: PRINT "*"
  S = S + 2 + SP
  IF S > N THEN
    IF SP THEN S = S - N ELSE S = (N MOD 2) + 1
  END IF
NEXT I
FOR I = 1 TO N
  LOCATE I + 2, 2 * N + 4: PRINT "(";
  PRINT LTRIM$(STR$(I)); ", "; LTRIM$(STR$(A(I))); " ) ";
  PRINT "SUM ="; I + A(I)
NEXT I

```

'3.9

' This program will determine # of moves made to a stack.

```

'
' 1 block - 1 move (obvious)
' 2 blocks- 3 moves (Move 1 stack, move #2, move 1 stack)
' 3 blocks- 7 moves (Move 2 stack, move #3, move 2 stack on #3)
' (3 moves + 1 move + 3 moves)
' 4 blocks- 15 moves (Move 3 stack, move #4, move 3 stack on #4)
' (7 moves + 1 move + 7 moves)
' N blocks- 2^N - 1 moves
DIM A(16)
INPUT "Enter N: "; N: A(1) = 1
FOR I = 2 TO N: A(I) = A(I - 1) * 2 + 1: NEXT I
PRINT A(N)

```

```

'3.10
' This program will find set of #s P, Q, R (P = Q x R).
,
INPUT "Enter S:"; S
Q = S
DO
  DO
    Q = Q + 1: X1 = INT(Q / 10): Y1 = Q - X1 * 10
  LOOP UNTIL X1 <> Y1
  NU = INT(10000 / Q)
  FOR R = NU TO 999
    X2 = INT(R / 100): C = R - X2 * 100
    Y2 = INT(C / 10): Z2 = C - Y2 * 10
    IF X2 <> Y2 AND Y2 <> Z2 AND X2 <> Z2 THEN
      IF X1 <> X2 AND X1 <> Y2 AND X1 <> Z2 THEN
        IF Y1 <> X2 AND Y1 <> Y2 AND Y1 <> Z2 THEN
          A(X1) = 1: A(Y1) = 1: A(X2) = 1: A(Y2) = 1: A(Z2) = 1
          P$ = STR$(Q * R)
          FOR I = 2 TO 6
            X = VAL(MID$(P$, I, 1)): IF A(X) THEN DUPL = -1
          NEXT I
          FOR I = 2 TO 5
            FOR J = I + 1 TO 6
              IF MID$(P$, I, 1) = MID$(P$, J, 1) THEN DUPL = -1
            NEXT J
          NEXT I
          IF NOT DUPL THEN
            PRINT "P ="; P$;
            PRINT USING " Q = ## R = ###"; Q; R: END
          END IF
          FOR I = 0 TO 9: A(I) = 0: NEXT I: DUPL = 0
        END IF
      END IF
    END IF
  NEXT R
LOOP UNTIL Q > 99

```