## FLORIDA HIGH SCHOOLS COMPUTING COMPETITION '92
## BASIC PROGRAM SOLUTIONS

```
'1.1
' This program displays the company name: GTEDS.
'
PRINT "GGGGG   TTTTT   EEEEE"
PRINT "G         T     E"
PRINT "G GGG     T     EEEEE   DATA SERVICES"
PRINT "G   G     T     E"
PRINT "GGGGG     T     EEEEE"
```

```
'1.2
' This program will display the company name in a year.
'
INPUT "Enter year:"; YEAR
IF YEAR < 1920 THEN
  PRINT "RICHLAND CENTER TELEPHONE COMPANY"
ELSE
  IF YEAR < 1926 THEN
     PRINT "COMMONWEALTH TELEPHONE COMPANY"
  ELSE
    IF YEAR < 1935 THEN
      PRINT "ASSOCIATED TELEPHONE UTILITIES COMPANY"
    ELSE
      IF YEAR < 1959 THEN
         PRINT "GENERAL TELPHONE CORPORATION"
      ELSE
        IF YEAR < 1982 THEN
          PRINT "GENERAL TELPHONE & ELECTRONICS CORPORATION"
        ELSE
          PRINT "GTE CORPORATION"
        END IF
      END IF
    END IF
  END IF
END IF
```

```
'1.3
' This program will determine company's ranking in Forbes.
'
INPUT "Enter 1991 rank:"; RANK
INPUT "Enter number of places:"; PLACES
PRINT RANK - PLACES
```

```
'1.4
' This program will indent GTE's 6 operations.
'
INPUT "Enter number of spaces:"; X
PRINT "GTE TELPHONE OPERATIONS"
PRINT SPACE$(X); "GTE GOVERNMENT SYSTEMS"
PRINT SPACE$(X * 2); "GTE MOBILE COMMUNICATIONS"
PRINT SPACE$(X * 3); "GTE INFORMATION SERVICES"
PRINT SPACE$(X * 4); "GTE SPACENET"
PRINT SPACE$(X * 5); "GTE AIRFONE"
```

```
'1.5
' This program will display # of WHOLE YEARS GTEDS existed.
'
INPUT "Enter M, Y:"; M, Y
IF M < 10 THEN X = 1 ELSE X = 0
PRINT Y - 1967 - X; "YEARS"
```

```
'1.6
' This program will center a title and name in a box.
'
INPUT "Enter title:"; T$
INPUT "Enter name:"; N$
PRINT STRING$(24, "*")
PRINT "*"; SPACE$(22); "*"
L = LEN(T$) + LEN(N$) + 1
SP1 = INT((22 - L) / 2)
SP2 = (22 - L) - SP1
PRINT "*"; SPACE$(SP1); T$; " "; N$; SPACE$(SP2); "*"
PRINT "*"; SPACE$(22); "*"
PRINT STRING$(24, "*")
```

```
'1.7
' This program will display a 4-line statement for ISOP.
'
INPUT "Enter name:"; N$
INPUT "Enter title:"; T$
INPUT "Enter group:"; G$
PRINT N$; " IS A "; T$; " WITHIN THE"
PRINT G$; " GROUP AND"
PRINT "HAS BEEN SELECTED TO PARTICIPATE IN"
PRINT "THE ISOP."
```

```
'1.8
' This program will display a dollar sign next to an amount.
'
INPUT "Enter amount: "; AMOUNT$
A = VAL(AMOUNT$)
IF A >= 2000 THEN PRINT "$2000.00" ELSE PRINT "$"; AMOUNT$
```

```
'1.9
' This program will display an acronym for business words.
'
INPUT "Enter words: "; ST$
PRINT LEFT$(ST$, 1);
FOR I = 2 TO LEN(ST$) - 1
  IF MID$(ST$, I, 1) = " " THEN PRINT MID$(ST$, I + 1, 1);
NEXT I
```

```
'1.10
' This program will calculate QUALITY hours and minutes.
'
INPUT "Enter number of technicians, N:"; N
INPUT "Enter number of minutes, M:"; M
TOTAL = 50 * 5 * N * M
HOURS = INT(TOTAL / 60)
MIN = TOTAL - HOURS * 60
PRINT HOURS; "HOURS"; MIN; "MINUTES"
```

```
'2.1
' This program will display a speech indented.
'
I = 0
WHILE (LINE$(I) > "") OR (I = 0)
  I = I + 1
  INPUT "Enter line:"; LINE$(I)
WEND
FOR J = 1 TO I - 1
  CH$ = MID$(LINE$(J), 1, 1)
  IF CH$ = "I" THEN PRINT LINE$(J)
  IF CH$ >= "A" AND CH$ <= "H" THEN PRINT SPACE$(4); LINE$(J)
  IF VAL(CH$) > 0 THEN PRINT SPACE$(8); LINE$(J)
NEXT J
```

```
'2.2
' This program will display a number in words.
'
DIM WORDS$(27)
DATA ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,EIGHT,NINE,TEN
DATA ELEVEN,TWELVE,THIRTEEN,FOURTEEN,FIFTEEN,SIXTEEN
DATA SEVENTEEN,EIGHTEEN,NINETEEN,TWENTY,THIRTY,FOURTY
DATA FIFTY,SIXTY,SEVENTY,EIGHTY,NINETY
FOR I = 1 TO 27: READ WORDS$(I): NEXT I
INPUT "Enter number:"; NUM
IF NUM < 20 THEN PRINT WORDS$(NUM): END
TENS = INT(NUM / 10)
UNITS = NUM - TENS * 10
PRINT WORDS$(18 + TENS);
IF UNITS > 0 THEN PRINT "-"; WORDS$(UNITS)
```

```
'2.3
' This program will display selected items from a NRD menu.
'
DATA "DEMONSTRATED INTEREST IN INFORMATION MANAGMENT."
DATA "DEMONSTRATED LEADERSHIP SKILLS."
DATA "STRONG GPA/PERFORMANCE HISTORY."
DATA "AT LEAST TWO COURSES IN ANY PROGRAMMING LANGUAGE."
DATA "INTERNSHIP OR WORK EXPERIENCE."
DATA "EFFECTIVE ORAL AND WRITTEN COMMUNICATION SKILLS."
DATA "CAREER DEVELOPMENT POTENTIAL."
FOR I = 1 TO 7: READ CRIT$(I): NEXT I
INPUT "Enter name:"; NAM$
INPUT "Enter degree:"; DEGREE$
FOR I = 1 TO 7
  PRINT USING "#. "; I; : PRINT CRIT$(I)
NEXT I
PRINT
INPUT "Select up to 7 items:"; ITEMS$
CLS
PRINT NAM$: PRINT DEGREE$
NUM = 0
FOR I = 1 TO 7
  I$ = LTRIM$(STR$(I))
  IF INSTR(1, ITEMS$, I$) > 0 THEN
    NUM = NUM + 1
    PRINT : PRINT USING "#. "; NUM; : PRINT CRIT$(I)
  END IF
NEXT I

'2.4
' This program will rate a speech.
'
DATA SPEECH VALUE,PREPARATION,MANNER,ORGANIZATION
DATA OPENING,BODY OF SPEECH,CONCLUSION
FOR I = 1 TO 7: READ CAT$(I): NEXT I
DATA EXCELLENT,ABOVE AVERAGE,SATISFACTORY
DATA SHOULD IMPROVE,MUST IMPROVE
FOR I = 1 TO 5: READ VERBAL$(I): NEXT I
FOR I = 1 TO 7
  PRINT "Enter rating for "; CAT$(I);
  INPUT ": "; RATING$(I)
NEXT I
FOR I = 1 TO 7
  NUM = 1
  WHILE (RATING$(I) <> VERBAL$(NUM)) AND (NUM < 7)
    NUM = NUM + 1
  WEND
  PRINT CAT$(I); ":"; NUM
  TOTAL = TOTAL + NUM
NEXT I
200 PRINT
210 AVE = TOTAL / 7
220 PRINT USING "AVERAGE NUMERICAL RATING = #.#"; AVE
230 PRINT "SPEECH RATING = "; VERBAL$(INT(AVE + .5))
```

```
'2.5
' This program will format GTEDS MISSION statement.
'
DATA "BE THE CUSTOMER-ORIENTED LEADER AND PROVIDER-OF-CHOICE "
DATA "OF QUALITY INFORMATION PRODUCTS AND SERVICES IN THE "
DATA "TELECOMMUNICATIONS MARKETPLACE AND SELECTED OTHER "
DATA "RELATED MARKETS IN SUPPORT OF GTE'S TELOPS GOALS."
FOR I = 1 TO 4: READ ST$(I): NEXT I
INPUT "Enter N:"; N
STATE$ = ST$(1) + ST$(2) + ST$(3) + ST$(4)
FOR I = 1 TO LEN(STATE$)
  CH$ = MID$(STATE$, I, 1)
  WORD$ = WORD$ + CH$
  IF (CH$ = " " OR CH$ = "-" OR CH$ = ".") THEN
    NUMCH = LEN(LINE$) + LEN(WORD$)
    IF CH$ = " " THEN NUMCH = NUMCH - 1
    IF NUMCH > N THEN PRINT LINE$: LINE$ = WORD$
    IF NUMCH <= N THEN LINE$ = LINE$ + WORD$
    WORD$ = ""
  END IF
NEXT I
PRINT LINE$; WORD$
```

```
'2.6
' This program will change (.) to (?) at end of sentence.
'
DATA WHAT,WHY,HOW,WHO,WHERE
FOR I = 1 TO 5: READ QUEST$(I): NEXT I
INPUT "Enter paragraph:"; PAR$: PRINT
FIRSTW = -1
FOR I = 1 TO LEN(PAR$)
  CH$ = MID$(PAR$, I, 1)
  IF CH$ = " " AND LEN(FIRSTW$) > 0 THEN
    FIRSTW = 0
  ELSE
    IF (CH$ = "." OR CH$ = "!" OR CH$ = "?") THEN
      IF CH$ = "." THEN
        FOR J = 1 TO 5
          IF FIRSTW$ = QUEST$(J) THEN CH$ = "?"
        NEXT J
      END IF
      FIRSTW$ = "": FIRSTW = -1
    ELSE
      IF FIRSTW AND (CH$ <> " ") THEN FIRSTW$ = FIRSTW$ + CH$
    END IF
  END IF
  PRINT CH$;
NEXT I
```

```
'2.7
' This program will print names in the office at a beep.
'
DATA DAVID,0700,1600
DATA DON,0800,1700
DATA DOUG,0730,1630
DATA GRANDVILLE,1230,2100
DATA JAMES,1130,2200
DATA JIM,0900,1800
DATA JOHN,0700,1600
DATA LINDA,1230,2300
DATA MARIE,0700,1600
DATA MATT,1230,2300
DATA PAULA,0700,1600
DATA ROBERT,0800,1700
DATA SHELLEY,0630,1530
DATA TOM,1100,1930
DIM NAM$(14), START(14), QUIT(14)
FOR I = 1 TO 14
  READ NAM$(I), START(I), QUIT(I)
NEXT I
INPUT "Enter time:"; TIME
INPUT "Enter day:"; DAY$
FOR I = 1 TO 14
 IF (START(I) <= TIME) AND (TIME <= QUIT(I)) THEN
  IF (DAY$ <> "SUNDAY") AND (DAY$ <> "SATURDAY") THEN
   INOFFICE = -1
   IF (NAM$(I) = "JAMES") AND (DAY$ = "MONDAY") THEN INOFFICE = 0
   IF (NAM$(I) = "LINDA") AND (DAY$ = "FRIDAY") THEN INOFFICE = 0
   IF (NAM$(I) = "MATT") AND (DAY$ = "MONDAY") THEN INOFFICE = 0
   IF INOFFICE THEN
     NUM = NUM + 1
     IF NUM = 1 THEN PRINT NAM$(I);
     IF NUM > 1 THEN PRINT ", "; NAM$(I);
   END IF
  END IF
 END IF
NEXT I
IF NUM = 0 THEN PRINT "NONE"
```

```
'2.8
' This program will randomly assign titles to a team.
'
DATA WILL,DARLENE,JEFF,LIZ,LORI,MARY,PING
FOR I = 1 TO 7: READ NAM$(I): NEXT I
DATA AUTHOR,MODERATOR,READER,RECORDER,INSPECTOR
FOR I = 1 TO 5: READ TITLE$(I): NEXT I
RANDOMIZE TIMER
INPUT "Enter author's name:"; TNAME$(1)
' Choose moderator
IF TNAME$(1) = NAM$(1) THEN
  TNAME$(2) = NAM$(2)
ELSE
  IF TNAME$(1) = NAM$(2) THEN
    TNAME$(2) = NAM$(1)
  ELSE
    TNAME$(2) = NAM$(INT(RND(3) * 2) + 1)
  END IF
END IF
' Choose next 3 title names
FOR I = 3 TO 5
  VALID = 0
  WHILE NOT VALID
    VALID = -1
    X = INT(RND(3) * 7) + 1
    FOR J = 1 TO I
      IF NAM$(X) = TNAME$(J) THEN VALID = 0
    NEXT J
  WEND
  TNAME$(I) = NAM$(X)
NEXT I
' Display all 5 titles and names.
FOR I = 1 TO 5
  PRINT TITLE$(I); " - "; TNAME$(I)
NEXT I
```

```
'2.9
' This program will sort a list of names with area codes.
'
DIM NAM$(15)
INPUT "Enter two area codes:"; AREA1, AREA2
INPUT "Enter number of names:"; NUM
FOR I = 1 TO NUM
  INPUT "Enter name:"; NAM$(I)
NEXT I
FOR I = 1 TO NUM - 1
  FOR J = I + 1 TO NUM
    IF NAM$(I) > NAM$(J) THEN SWAP NAM$(I), NAM$(J)
  NEXT J
NEXT I
IF AREA1 > AREA2 THEN A = AREA1: AREA1 = AREA2: AREA2 = A
MID = INT((NUM + 1) / 2)
FOR I = 1 TO MID
  PRINT AREA1; "- "; NAM$(I)
NEXT I
FOR I = MID + 1 TO NUM
  PRINT AREA2; "- "; NAM$(I)
NEXT I
```

```
'2.10
' This program will adjust a golf score by handicap.
'
DATA 5,4,4,4,3,4,4,3,5
FOR I = 1 TO 9: READ PAR(I): NEXT I
INPUT "Enter handicap:"; HAND
PRINT "Enter gross scores:";
INPUT G(1), G(2), G(3), G(4), G(5), G(6), G(7), G(8), G(9)
PRINT "HOLE #:";
FOR I = 1 TO 9: PRINT USING "####"; I; : NEXT I
PRINT : PRINT "PAR:   ";
FOR I = 1 TO 9
  PRINT USING "####"; PAR(I);
  PARTOT = PARTOT + PAR(I)
NEXT I
PRINT : PRINT "GROSS: ";
FOR I = 1 TO 9
  PRINT USING "####"; G(I);
  GTOT = GTOT + G(I)
NEXT I
PRINT : PRINT "ADJUST:";
' Determine # of tripple and double bogeys allowed
IF HAND > 9 THEN BOG(3) = HAND - 9: BOG(2) = 9 - BOG(3)
IF HAND <= 9 THEN BOG(2) = HAND: BOG(1) = 9 - BOG(2)
' Adjust the gross scores by Handicap
FOR I = 1 TO 9
  DIFF = G(I) - PAR(I)
  ADJUSTED = 0
  B = 3
  WHILE NOT ADJUSTED AND (B > 0)
    IF (BOG(B) > 0) AND (DIFF >= B) THEN
      A(I) = PAR(I) + B
      BOG(B) = BOG(B) - 1
      ADJUSTED = -1
    END IF
    B = B - 1
  WEND
  IF NOT ADJUSTED THEN A(I) = G(I)
NEXT I
' Display the adjusted scores and totals
FOR I = 1 TO 9
  PRINT USING "####"; A(I);
  ATOT = ATOT + A(I)
NEXT I
PRINT : PRINT
PRINT "PAR TOTAL:"; PARTOT
PRINT "GROSS TOTAL:"; GTOT
PRINT "ADJUST TOTAL:"; ATOT
PRINT "ROUND HANDICAP:"; ATOT - PARTOT
```

```
'3.1
' This program will move a triangle of GTEDS around the screen.
'
DATA "           "
DATA "     G     "
DATA "    T T    "
DATA "   E   E   "
DATA "  D     D  "
DATA " SDETGTEDS "
DATA "           "
FOR I = 1 TO 7: READ A$(I): NEXT I
CLS
ROW = 9: COL = 34
WHILE CH$ <> CHR$(27)
  FOR I = 1 TO 7
    LOCATE ROW + I, COL: PRINT A$(I);
  NEXT I
  C$ = INKEY$: IF C$ > "" THEN CH$ = C$
  FOR I = 1 TO 100: NEXT I
  SELECT CASE UCASE$(CH$)
    CASE "I":  ROW = ROW - 1
    CASE "M":  ROW = ROW + 1
    CASE "J":  COL = COL - 1
    CASE "K":  COL = COL + 1
  END SELECT
  IF ROW = 0 THEN ROW = 1: CH$ = ""
  IF COL = 0 THEN COL = 1: CH$ = ""
  IF ROW = 18 THEN ROW = 17: CH$ = ""
  IF COL = 69 THEN COL = 68: CH$ = ""
WEND
```

```
'3.2
' This program will display a date in 1992 after # of days.
'
DIM MONTH(12), MNAME$(12)
DATA TUESDAY,WEDNESDAY,THURSDAY,FRIDAY,SATURDAY,MONDAY
FOR I = 1 TO 6: READ DAY$(I): NEXT I
DATA 31,29,31,30,31,30,31,31,30,31,30,31
FOR I = 1 TO 12: READ MONTH(I): NEXT I
DATA JANUARY,FEBRUARY,MARCH,APRIL,MAY,JUNE,JULY,AUGUST
DATA SEPTEMBER,OCTOBER,NOVEMBER,DECEMBER
FOR I = 1 TO 12: READ MNAME$(I): NEXT I
INPUT "Enter X:"; X
X = X + 1
D = (X MOD 6) + 1
PRINT DAY$(D); " ";
X = X + INT((X + 1) / 6)
I = 1
WHILE SUM + MONTH(I) < X
  SUM = SUM + MONTH(I):  I = I + 1
WEND
PRINT MNAME$(I); X - SUM
IF DAY$(D) <> "SATURDAY" THEN END
  X = X + 1
  WHILE SUM + MONTH(I) < X
    SUM = SUM + MONTH(I):  I = I + 1
  WEND
  PRINT "SUNDAY "; MNAME$(I); X - SUM
```

```
'3.3
' This program will release program modules for PWS.
'
WHILE NOT ALLDONE
  I = NUM + 1
  INPUT "Enter name, program:"; NAME$(I), PROG$(I)
'  Find previous Name/Prog or make addition
  J = 1
  NOTFOUND = (NAME$(J) <> NAME$(I) OR PROG$(J) <> PROG$(I))
  WHILE (J < I) AND NOTFOUND
    J = J + 1
    NOTFOUND = (NAME$(J) <> NAME$(I) OR PROG$(J) <> PROG$(I))
  WEND
  I = J
  IF I > NUM THEN NUM = I
  INPUT "Enter completed, release:"; COMP$(I), REL$(I)
  IF REL$(I) = "Y" THEN COMP$(I) = "Y"
  MODCOMP = (COMP$(I) = "Y")
' Check if Module completed by all, and at least 1 released
  IF MODCOMP THEN
    MODREL = 0
    FOR J = 1 TO NUM
      IF PROG$(J) = PROG$(I) THEN
        IF COMP$(J) <> "Y" THEN MODCOMP = 0
        IF REL$(J) = "Y" THEN MODREL = -1
      END IF
    NEXT J
'   If Module completed by all and 1 or more released
    IF (MODCOMP AND MODREL) THEN
      PRINT "MODULE "; PROG$(I); " HAS BEEN RELEASED"
      MODULE$ = PROG$(I)
      FOR J = 1 TO NUM
        IF PROG$(J) = MODULE$ THEN PROG$(J) = ""
      NEXT J
      ALLDONE = -1
      FOR J = 1 TO NUM
        IF PROG$(J) <> "" THEN ALLDONE = 0
      NEXT J
    END IF
  END IF
WEND
```

```
'3.4
' This program will produce acronyms for phone numbers.
'
DIM A$(18), B$(18)
DATA AGENT,SOAP,MONEY,JEWEL,BALL,LOANS,CARE,SAVE,CALL,PAVE
DATA KEEP,KINGS,KNIFE,KNOCK,JOINT,JUICE,LOBBY,RATE
FOR I = 1 TO 18: READ B$(I): A$(I) = B$(I): NEXT I
L1$ = " ADGJMPTW"
L2$ = " BEHKNRUX"
L3$ = " CFILOSVY"
' Sort the data alphabetically
FOR I = 1 TO 17
  FOR J = I + 1 TO 18
    IF A$(I) > A$(J) THEN SWAP A$(I), A$(J)
  NEXT J
NEXT I
'
INPUT "Enter phone #:"; PH$
P4$ = MID$(PH$, 5, 4): P5$ = MID$(PH$, 3, 1) + P4$
' Convert words to number strings
FOR I = 1 TO 18
  L = LEN(A$(I)): NUM$ = ""
  FOR J = 1 TO L
    K = 1: C$ = MID$(A$(I), J, 1): NOMATCH = -1
    WHILE NOMATCH
      K = K + 1
      IF MID$(L1$, K, 1) = C$ THEN NOMATCH = 0
      IF MID$(L2$, K, 1) = C$ THEN NOMATCH = 0
      IF MID$(L3$, K, 1) = C$ THEN NOMATCH = 0
    WEND
    NUM$ = NUM$ + CHR$(48 + K)
  NEXT J
  IF L = 4 AND NUM$ = P4$ THEN PRINT MID$(PH$, 1, 4); A$(I)
  IF L = 5 AND NUM$ = P5$ THEN
    PRINT MID$(PH$, 1, 2); MID$(A$(I), 1, 1); "-";
    PRINT MID$(A$(I), L - 3, 4)
  END IF
NEXT I
```

```
'3.5
' This program will find seven 7-digit squares in base 8.
'
NUM = 1242: SNUM = 0
WHILE SNUM < 7
  NUM1$ = MID$(STR$(NUM), 2)
' Convert NUM1$ to base 10 number NUM1V
  NUM1V = 0
  FOR I = 1 TO 4
    DIGIT = ASC(MID$(NUM1$, I, 1)) - ASC("0")
    POWER = 1
    FOR J = 1 TO LEN(NUM1$) - I
      POWER = POWER * 8
    NEXT J
    NUM1V = NUM1V + DIGIT * POWER
  NEXT I
  NUM1V = NUM1V * NUM1V
  SQUARE$ = "":  VALID = -1
  FOR I = 0 TO 7: DUP(I) = 0: NEXT I
' Convert Num1V to Base8 number
  J = INT(LOG(NUM1V) / LOG(8))
  WHILE (J >= 0) AND VALID
    POWER = 1
    FOR K = 1 TO J: POWER = POWER * 8: NEXT K
    X = INT(NUM1V / POWER)
'  Check for duplicate digits
    IF DUP(X) THEN
      VALID = 0
    ELSE
      DUP(X) = -1
      SQUARE$ = SQUARE$ + CHR$(48 + X)
      NUM1V = NUM1V - X * POWER
    END IF
    J = J - 1
  WEND
  IF VALID THEN SNUM = SNUM + 1: PRINT SQUARE$; " "; NUM
' Increment to next base 8 number
  NUM = NUM + 1: NUMST$ = LTRIM$(STR$(NUM))
  WHILE INSTR(1, NUMST$, "8") > 0 OR INSTR(1, NUMST$, "9") > 0
    NUM = NUM + 1
    NUMST$ = LTRIM$(STR$(NUM))
  WEND
WEND
```

```
'3.6
' This program will find 3 distinct integers that are pairwise
' relatively prime such that they sum to N.
'
INPUT "Enter N:"; N
X = 2 + (N MOD 2)
WHILE (X < INT(N / 3)) AND NOT FOUND
  Y = X + 1
  WHILE (Y < INT((N - X) / 2)) AND NOT FOUND
    Z = N - X - Y: FOUND = -1
    FOR I = 2 TO Y
      IF (X MOD I = 0) AND (Y MOD I = 0) THEN FOUND = 0
      IF (X MOD I = 0) AND (Z MOD I = 0) THEN FOUND = 0
      IF (Y MOD I = 0) AND (Z MOD I = 0) THEN FOUND = 0
    NEXT I
    IF FOUND THEN PRINT X; "+"; Y; "+"; Z; "="; N
    IF NOT FOUND THEN Y = Y + 1
  WEND
  Z = Z + 1
WEND
```

```
'3.7
' This program will print combinations of 6 soccer players.
'
DATA ANDY,DAN,DOUG,JACK,MIKE,YEHIA
FOR I = 1 TO 6: READ NAME$(I): NEXT I
INPUT "Enter number of substitutes:"; NUMOFSUB
L = 6 + NUMOFSUB
FOR I = 7 TO L
  INPUT "Enter name:"; NAME$(I)
NEXT I
' Sort names with substitutes
FOR I = 1 TO L - 1
  FOR J = I + 1 TO L
    IF NAME$(I) >= NAME$(J) THEN SWAP NAME$(I), NAME$(J)
  NEXT J
NEXT I
'
M = 6
FOR I = 1 TO M: A(I) = M - I + 1: NEXT I
N = 1: A(1) = A(1) - 1
WHILE N <= M
  A(N) = A(N) + 1
  IF N > 1 THEN
    FOR I = N - 1 TO 1 STEP -1: A(I) = A(I + 1) + 1: NEXT I
  END IF
  IF A(N) <= L - N + 1 THEN
    S = S + 1
    PRINT S; NAME$(A(M));
    FOR I = M - 1 TO 1 STEP -1
      PRINT ","; NAME$(A(I));
    NEXT I
    PRINT
    N = 0
    IF S MOD 24 = 0 THEN WHILE INKEY$ = "": WEND
  END IF
  N = N + 1
WEND
```

```
'3.8
' This program displays the Bill Date and the Due Date.
' January 1, 1992 was a Wednesday
'
DIM MNAME$(12), MON(12), MHOL(12), DHOL(12)
DATA JANUARY,FEBRUARY,MARCH,APRIL,MAY,JUNE,JULY,AUGUST
DATA SEPTEMBER,OCTOBER,NOVEMBER,DECEMBER
FOR I = 1 TO 12: READ MNAME$(I): NEXT I
DATA 31,29,31,30,31,30,31,31,30,31,30,31
FOR I = 1 TO 12: READ MON(I): NEXT I
DATA TUESDAY,WEDNESDAY,THURSDAY,FRIDAY,SATURDAY,SUNDAY,MONDAY
FOR I = 1 TO 7: READ DNAME$(I): NEXT I
INPUT "Enter month of bill:"; MNUM
INPUT "Enter cycle number:"; CYCLE
INPUT "Enter number of days:"; NUMDAYS
H = 1
INPUT "Enter holiday MM, DD:"; MHOL(H), DHOL(H)
WHILE MHOL(H) > 0
  H = H + 1
  INPUT "Enter holiday MM, DD:"; MHOL(H), DHOL(H)
WEND
H = H - 1: PRINT
DAYS(1) = 0
FOR I = 1 TO MNUM - 1
  DAYS(1) = DAYS(1) + MON(I)
NEXT I
DAY(1) = 3 * CYCLE - 2:  DAY(2) = DAY(1) + NUMDAYS
DAYS(2) = DAYS(1) + DAY(2)
DAYS(1) = DAYS(1) + DAY(1)
FOR T = 1 TO 2
  HOL = 1: WKEND = 1
' Decrement days counter if holiday or weekend
  WHILE (HOL = 1) OR (WKEND = 1)
    HOL = 0: WKEND = 0
    IF DAY(T) > MON(MNUM) THEN
      DAY(T) = DAY(T) - MON(MNUM)
      MNUM = MNUM + 1
    END IF
    FOR I = 1 TO H
      IF MHOL(I) = MNUM AND DHOL(I) = DAY(T) THEN
        DAY(T) = DAY(T) + 1
        DAYS(T) = DAYS(T) + 1: HOL = 1
      END IF
    NEXT I
    X = DAYS(T) MOD 7
    IF (X = 4) OR (X = 5) THEN
'     Saturday or Sunday
      DAY(T) = DAY(T) + 1
      DAYS(T) = DAYS(T) + 1: WKEND = 1
    END IF
  WEND
  IF T = 1 THEN PRINT "BILL ";  ELSE PRINT "DUE ";
  PRINT "DATE: "; DNAME$(X + 1); " "; MNAME$(MNUM); DAY(T)
NEXT T
```

```
'3.9
' This program will calculate the area of a polygon room.
'
INPUT "Enter number of sides:"; SIDES
FOR I = 1 TO SIDES
  INPUT "Enter movement:"; MOV$
  DIR$(I) = MID$(MOV$, 1, 1)
  L = LEN(MOV$)
  MOV$ = MID$(MOV$, 2, L - 1)
  DIST(I) = VAL(MOV$)
' Subtract Down and Left directions
  IF DIR$(I) = "D" OR DIR$(I) = "L" THEN DIST(I) = -DIST(I)
NEXT I
' Multiply length by width to obtain rectangle area,
' then add or subtract area from overall area.
I = 1: SUM = 0: AREA = 0
WHILE (I <= SIDES)
  SUM = SUM + DIST(I)
  AREA = AREA + (SUM * DIST(I + 1))
  I = I + 2
WEND
PRINT "AREA ="; ABS(AREA); "SQUARE FEET"
```

```
'3.10
' This program will display the reasons a Rubik's Cube is
' unsolvable.  Input is to be separated by a space (not a ,)
'
DATA "TOP:   ","FRONT: ","RIGHT: ","BACK:  ","LEFT:  "
DATA "BOTTOM:"
FOR I = 1 TO 6: READ SIDE$(I): NEXT I
EDGES$ = "T2P2 T6R2 T8F2 T4L2 F4L6 F6R4 "
EDGES$ = EDGES$ + "R6P4 P6L4 F8B2 R8B6 P8B8 L8B4"
FOR I = 1 TO 6
  PRINT "Enter colors on "; SIDE$(I);
  INPUT COLORS$
  FOR J = 1 TO 9
    COL$(I, J) = MID$(COLORS$, J * 2 - 1, 1)
  NEXT J
NEXT I
'
MIDUNIQUE = -1
FOR I = 1 TO 5
  FOR J = I + 1 TO 6
    IF COL$(I, 5) = COL$(J, 5) THEN MIDUNIQUE = 0
  NEXT J
NEXT I
'
IF NOT MIDUNIQUE THEN
  PRINT "COLORS ON MIDDLE SQUARES ARE NOT UNIQUE"
END IF
FOR K = 1 TO 12
  S1 = INSTR(1, "TFRPLB", MID$(EDGES$, K * 5 - 4, 1))
  N1 = ASC(MID$(EDGES$, K * 5 - 3, 1)) - ASC("0")
  S2 = INSTR(1, "TFRPLB", MID$(EDGES$, K * 5 - 2, 1))
  N2 = ASC(MID$(EDGES$, K * 5 - 1, 1)) - ASC("0")
  IF COL$(S1, N1) = COL$(S2, N2) THEN ENUM = ENUM + 1
NEXT K
PRINT "NUMBER OF EDGE PIECES HAVING SAME COLOR:"; ENUM
```