

# CONTROL STATEMENTS

(IF..THEN, IF..THEN..ELSE, GOTO,  
ON..GOTO, ON..GOSUB)

## 18.1 The IF... THEN Statement

1. This statement is used for decision making.
2. IF the condition given is satisfactory, then it does the task after THEN statement.
3. When the condition is not satisfied, then BASIC executes the line next to it.

## 18.2 The IF... THEN .. ELSE Statement

This is a control flow statement that allows conditional execution or branching, based on the evaluation of an expression that must be either true or false. IF..THEN..ELSE is the variation of IF..THEN. If condition is satisfied do the following task, otherwise do the next task. The syntax of IF... THEN.. ELSE statement is:

### Block Syntax: IF...THEN...ELSE

IF booleanexpression1 THEN

[statement block-1]

[ELSEIF booleanexpression2 THEN

[statementblock-2]

.....

[ELSE

[statement block-n]]

END IF

Here, *Boolean expression* is an expression that must return non-zero (true) or zero (false).

*Statement block* consists of any number of statements on one or more lines.

### Single-line Syntax: IF...THEN...ELSE

IF boolean expression THEN *then part* [ELSE *else part*]

Here, *boolean expression* is an expression that must return non-zero (true) or zero (false).

*then part* consists of the statements or branches performed when *booleanexpression* is true.

*else part* consists of the statements or branches performed when *booleanexpression* is false.

### Single Line Example

(a) IF (Hours% > RegHrs%) THEN GOTO 1800 ELSE GOTO 2800

(b) IF (Hours% > RegHrs%) THEN OverTime ELSE RegTime

(c) IF (I = Svalue) THEN

I = I^2: Svalue = SQR(Svalue):PP=Svalue+I: ELSE 400

## Skeleton of Example:

```
IF... THEN ... ELSE ..... END IF

IF (OutStanding! <= CreditLimit!) THEN
    GOSUB AccoutsUPdate
    GOSUB LedgerEntry
    GOSUB JE
    ELSE
        GOSUB OverDueAccts
        GOSUB PaybleAgind
    END IF

IF Switch = 1 THEN
    GOSUB Step1: GOSUB Step23
    CALL Evaluate
ELSEIF Switch =2 THEN
    GOSUB Step2: GOSUB Step24
    CALL Evaluate
ELSEIF Switch = 3 THEN
    GOSUB Step3: GOSUB Step25
END IF
```

**Note:** The outline sketch of the program shown above is just a skeleton. It does not contain detail in this section.

Single line IF..THEN statement is illustrated with the help of the following program. Note that the program contains some advanced statements, which are not discussed yet. But you can test the program in your computer.

## Example 1

```
CLS
PRINT "This is a guessing game."
PRINT "I have a number in mind. You try to guess what it is."
RANDOMIZE TIMER
Guess% = INT(10-1)*RND+1
PRINT "You have three guesses..."
Cnt = 1
DO WHILE (Cnt <=3)
    INPUT "Your guess", InG%
    IF InG% > Guess% THEN PRINT "Too high..."
    IF InG% < Guess% THEN PRINT "Too low..."
    IF InG% = Guess% THEN PRINT "Bingo !":Cnt =4
    Cnt=Cnt+1
LOOP
PRINT "My number was";Guess%; "and your number was";InG%
END
```

**Note:** A new statement of QBASIC is included. They are: RANDOMIZE TIMER, DO WHILE.. LOOP

Single line IF..THEN statement can be understood by running the following program. This is a modification from GWBASIC.

### Example 2

```
CLS
10: INPUT "Enter marks scored in Nepali";N
    IF N<1 OR N>99 THEN PRINT "You are wrong to enter marks":GOTO 10
20: INPUT "Enter marks scored in Science";S
    IF S<1 OR S>99 THEN PRINT "You are wrong to enter marks":GOTO 20
30: INPUT "Enter marks in Mathematics";M
    IF M<1 OR M>99 THEN PRINT "You are wrong to enter marks": GOTO 30
40: INPUT "Enter marks in English";E
    IF E<1 OR E>99 THEN PRINT "You are wrong to enter marks":GOTO 40
    S1 = N+S+M+E
    A = S1/4
    PRINT:PRINT:PRINT
    PRINT "Your result"
    PRINT "Marks in Nepali:";N
    PRINT "Marks in Science:";S
    PRINT "Marks in Mathematics:";M
    PRINT "Marks in English:";E
    PRINT "TOTAL MARKS:";S1
    PRINT "AVERAGE MARKS:";A
END
```

### Example 3

```
CLS
REM ***** THIS IS NEW TEST *****
PRINT "THIS IS ONE IMPORTANT COMMAND"
PRINT "ENTER YOUR NAME"
INPUT N$
PRINT "YOUR NAME IS:";N$
END
```

**Example 4:** Write a program to calculate the area of a square.

```
CLS
INPUT "Enter the length of a square";L
LET A=L^2
CLS
PRINT "The length of the square is:";L
PRINT "Area of the square is:";A
END
```

**Example 5:** Write a program to check eligibility of a person participating in an election.

```
PRINT "THIS IS TEST PROGRAM"
PRINT "THIS PROGRAM PRINTS STRING, NUMBERS AND VARIABLES"
INPUT "Enter your name";N$
INPUT "Enter your age";A
IF A<0 OR A> 110 THEN PRINT "YOU DID NOT ENTER CORRECT NUMBER":END
IF A>=18 THEN PRINT "You are eligible to cast vote"
PRINT "Your age is";A
PRINT "Your name is";N$
END
```

- Press **Alt+R** (Run) or press **F5**
- Select **Start**
- Press a key to continue.

**Example 6**

```
CLS
PRINT "THIS IS LET STATEMENT TEST"
INPUT "Enter a number";N
INPUT "Enter second number";M
LET P = N*M
LET D = N-M
LET S = N+M
PRINT "Sum of the two numbers is";S
PRINT "Difference of the two numbers is";D
PRINT "Product of the two numbers is";P
END
```

**Example 7**

```
CLS
PRINT "THIS IS TEST PROGRAM"
20:
INPUT "ENTER YOUR MONEY DEPOSITED";P
IF P<=0 OR P>50000000 THEN PRINT "RE-ENTER":GOTO 20
40:
INPUT "ENTER FOR HOW MANY YEARS";N
    IF N<=0 OR N>=20 THEN PRINT "RE-ENTER THE NUMBER":GOTO 40
    IF N<=5 THEN R = 10
    IF N>5 OR N<=10 THEN R = 15
    IF N>10 THEN R = 18
        I = (P*N*R)/100
        A = P+I
        PRINT:PRINT
PRINT "***** OUTPUT *****"
PRINT "INTEREST IS";I
PRINT "COMPOUNDED MONEY IS";A
PRINT:PRINT:PRINT
PRINT "***** THANK YOU FOR VISITING US ****"
END
```

## 18.3 The GOTO Statement

GOTO is a control flow statement that branches unconditionally to the specified line.

### Syntax

`GOTO {linelabel / linenumbers}`

Here,

*linelabel* or *linenumber* is the label of the line to execute next.

This line must be in the same procedure or subroutine as the GOTO statement makes a jump to the main module only.

### Example 8:

Write a program to test the goto statement.

```
CLS
PRINT "THIS IS GOTO TEST PROGRAM"
GOTO 60
40:
PRINT "THIS IS SECOND GOTO STATEMENT"
GOTO 100
60:
PRINT "THIS IS ALSO GOTO"
GOTO 40
100:
PRINT "THIS IS END"
END
```

Here, line numbers are treated as **line labels**. Instead of line numbers, you can name them. The names thus given are called **label name**. GOTO statement is followed with a line label or label name. The *label name* or *line label* has a colon sign where from the program begins to execute as it encounters GOTO statement.

**EXIT:** This is a control flow statement that exists in DEF FNfunction, DO...LOOP, FOR...NEXT loop, FUNCTION or SUB Statements.

### Syntax

`EXIT {DEF / DO / FOR / FUNCTION / SUB}`

### Example 9:

Write a program to find the area of a square.

```
CLS
PRINT "THIS PROGRAM CALCULATES AREA OF A SQUARE"
INPUT "Enter length of square";L
IF L <=1 OR L>50 THEN END
A = L^2
PRINT "The area of square is";A
END
```

**Example 10:** Write a program to find the circumference of a circle.

```
CLS
Start:
    PRINT "This program calculates circumference"
    INPUT R
    COUNT = 0
    IF R<=0 OR R>100 THEN PRINT "Wrong entry - Retry":GOTO Start
    P = 3.1416
    A =2*P*R
    PRINT "Area of circle ";A
    COUNT = COUNT +1
    IF COUNT >10 THEN PRINT "OK":END
    GOTO Start
END IF
```

**Example 11:** Write a program to classify the triangles as the user enters the length of three sides.

```
REM Program for classifying triangles.
INPUT "Enter first side of triangle"; a
INPUT "Enter second side of triangle"; b
INPUT "Enter third side of triangle"; c
IF (a + b < c) OR (a + c < b) OR (b + c < a) THEN
    PRINT "No triangle is possible"
ELSEIF a = b AND b = c THEN
    PRINT "Equilateral triangle"
ELSEIF (a = b AND a <> c) OR (a = b AND b <> c) OR (b = c AND a <> b) THEN
    PRINT "Triangle is isosceles"
ELSE PRINT "Triangle is scalene"
END IF
```

## 18.4 The ON .. GOTO Statement

ON..GOTO is a branching statement. ON...GOTO is followed with a line label or label name.

**Syntax:** ON *expression* GOTO *line1, line2*

The expression component is any numeric expression yielding an integer result. When result is not an integer, the result is rounded to an integer value before the GOTO statement. ON..GOTO resemble a multilevel statement like IF..THEN .. ELSE statement or a CASE SELECT statement, only on more compact.

ON..GOTO performs a conditional jump on a particular expression. More than one line can be stated after GOTO statement. After the evaluation of the expression, if the condition is true, the control pointer moves the first line number (line label) or label name. For each keyboard entry, the expression is evaluated and the pointer is redirected to the line that matches the condition. The skeleton of ON..GOTO can be expressed as;

ON X GOTO 500, 600, 700

[ON expression GOTO line number, line number, line number]

ON X+Y-Z GOTO 770, 880, 990

OX 7\*A-3\*8 GOTO 100, 2000, 2200

### Example 12

CLS

Start:

PRINT "CHOOSE 9 TO QUIT"

PRINT "PROGRAM TO FIND 7 WONDERS OF THE WORLD"

INPUT "WHICH WONDER DO YOU WANT";W

ON W GOTO 10, 20, 30, 40, 50, 60, 70, 80, 90 PRINT "SELECTION LIMITED TO ONLY 1 THROUGH 9"

IF W<1 OR W > 9 THEN GOTO Start

10:

PRINT "MT. EVEREST NEPAL"

GOTO 81

20:

PRINT "VICTORIA FALLS (ZIMBABWE)"

GOTO 81

30:

PRINT "GRAND CANYON (COLORADO RIVER)"

GOTO 81

40:

PRINT "GREAT BARRIER REEF (AUSTRALIA)"

GOTO 81

50:

PRINT "CAVES IN FRANCE AND SPAIN"

GOTO 81

60:

PRINT "PARICUTION (VOLCANO IN MEXICO)"

GOTO 81

70:

PRINT "HARBOR AT RIO DE JANEIRO"

GOTO 81

80:

PRINT "GRAND COULEE DAM...?"

GOTO 81

81:

PRINT:PRINT

INPUT "WHEN FINISHED TOUCH ENTER";Q\$

GOTO Start

90:

END

## 18.5 The ON.. GOSUB Statement

This is a branching statement to a specified line number or label depending on the value of the given expression.

**Syntax:** ON *expression* GOSUB *line1, line2*

Here, *Line1, line2* are the line numbers followed with colon or line labels. The *expression* component is any numeric valid expression yielding an integer result. The program branches to the *line1* if the expression yields as 1, and to *line2* if the expression yields a 2 and so on. An error message “Illegal function call” is generated when the expression yields a negative number or a number larger than 255.

**Example 13:** To accept a number as temperature and convert.

```
CLS
PRINT : PRINT "Centrigade to Fahrenheit to Centigrade..."
PRINT : INPUT "1-> C to F; 2-> F to C"; Fc
ON Fc GOSUB C2F, F2C
END
C2F:
INPUT "Enter temperature"; Cent
PRINT
PRINT "Temperature in Centigrade"; Cent; "in Fahrenheit"; (9 * Cent + 160) / 5
RETURN

F2C:
INPUT "Enter temperature in fahrenheit"; Farhn
PRINT
PRINT "Temperature in Fahrenheit"; Farhn; "Celsius"; (5 * Farhn - 160) / 9
RETURN
```

**Note:** Entire program should be written in the same screen. Do not use SUB..END SUB here.

**Example 14**

```
CLS
REM ** THIS IS INPUT MODULE **
FOR I = 1 TO 3
    READ A,B,C
NEXT I
GOSUB First
GOTO Second
```

First:

```
REM *** TECHNICAL SUBROUTINE ***
S = A+B+C
P = A*B*C
D = A-B-C
S1 = A^2+B^2+C^2
C1 = A^3+B^3+C^3
PRINT A;B;C; "RESULTS";S;D;P;S1;C1
RETURN
REM *** DATA MODULE ***
```

Second:

```
DATA 4.2, 3.9, 1.7
DATA 3.7,1.4,5.4
DATA 2.1, 5.4, 3.6
```

## EXERCISES

1. Write a program to find the roots of a quadratic equation  $ax^2+bx+c = 0$ , if the values of a, b and c are given.
2. An examination board declares a student passed if:
  - (a) he gets 32 out of 100 in each of the five subjects (English, Nepali, Mathematics, Science and Computer) and
  - (b) he gets 35 percent in aggregate.  
Write a program to find whether the student passed or not, if the marks in five subjects (out of 100) are given.
3. Write a program to find the average of 15 given numbers, using only three variables.
4. The lengths of three sides of a triangle are given. Write a program to find the area of the triangle.
5. The lengths of three sides of a triangle are given. Write a program to test whether it is an isosceles triangle.
6. The lengths of three sides of a triangle are given. Write a program to test whether it is a right-angled triangle.
7. The lengths of three sides of a triangle are given. Write a program to test whether it is an equilateral triangle.
8. You are given three steel bars of different lengths. Write a program to test whether a triangle can be formed.
9. Write a program to accept 3 numbers from the keyboard and determine the smallest and largest among them.
10. Write a program to find the sum of squares of first 10 natural numbers.