# UNIT 3 : THE CONTROL FLOW

Advanced Program

Nguyen Thi Thu Huong-SoICT-HUST

# FUNCTIONS PRINTF, SCANF

- Input by using the scanf() function
- Output by using the printf() function
- It is required to declare the header <stdio.h>

# THE SYNTAX OF PRINTF() FUNCTION

**printf("[string]"[,list of arguments]);**

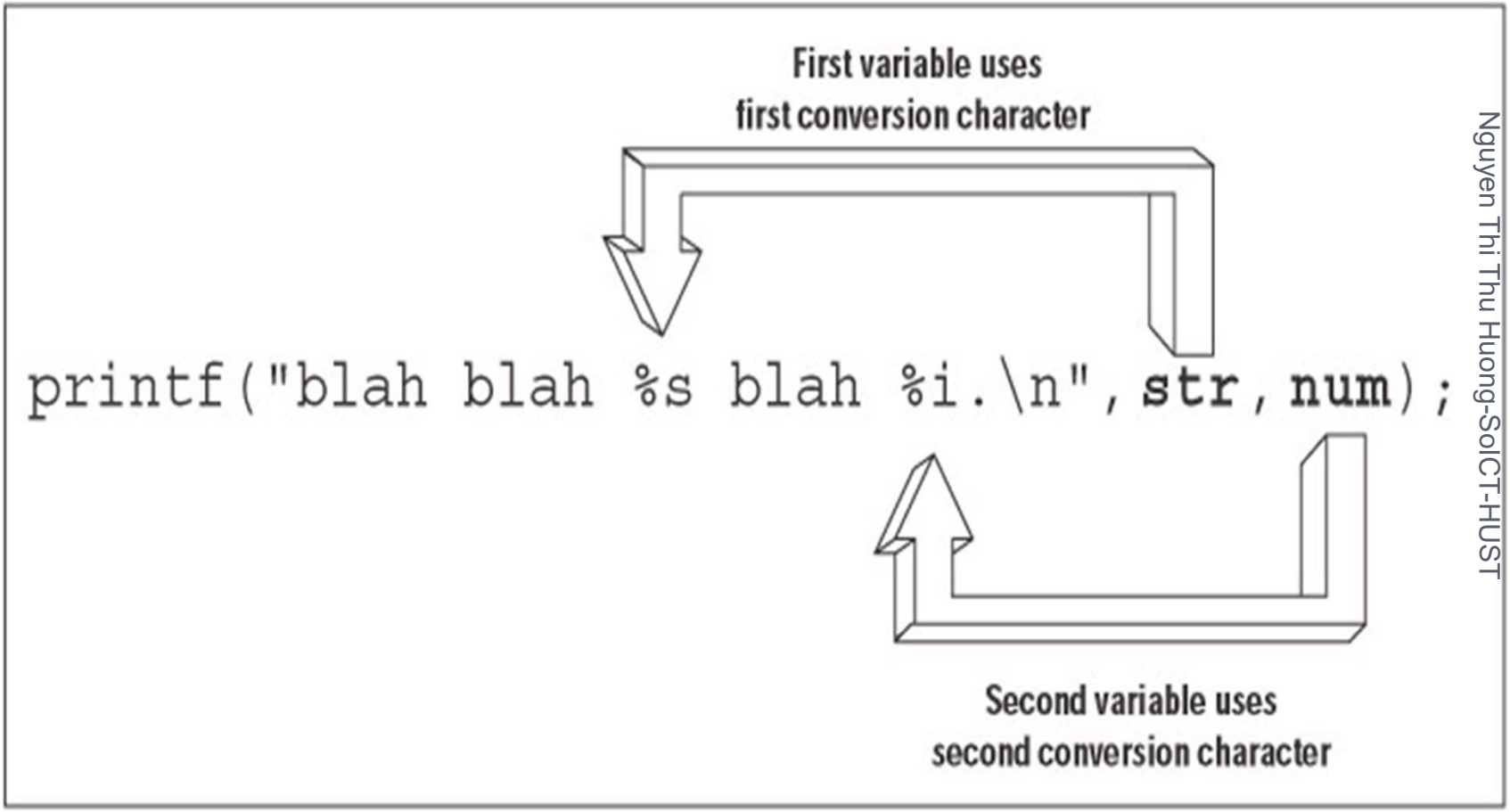
**List of arguments** : expressions, separated by commas.

The **string** is usually called the *control string* or the *format string*.

Action:

- Scan the string from left to right
- Prints on the screen any characters it encounters - except when it reaches a % character

# How printf function works



# THE % FORMAT SPECIFIERS

| | Usual variable type | Display |
|---|---|---|
| %c | char | single character |
| %d (%i) | int | signed integer |
| %e (%E) | float or double | exponential format |
| %f | float or double | signed decimal |
| %g (%G) | float or double | use %f or %e as required |
| %o | int | unsigned octal value |
| %s | array of char | sequence of characters |
| %u | int | unsigned decimal |
| %x (%X) | int | unsigned hex value |

# THE SYNTAX OF SCANF() FUNCTION

**scanf("control string",list of addresses)**

- **List of addresses** : addresses of variables, separated by commas
- The format string:
  - Has some extra items to cope with the problems of reading data in
  - Specifies how strings of characters
- All of the specifiers listed in connection with printf can be used with scanf.
- Action:
- Change the values stored in the parts of memory that is associated with variables.
- Simple variables have to be passed with a preceding &.
- Process the control string from left to right
- Each time it reaches a specifier : interpret what has been typed as a value.
- Input multiple values : separated by white space (spaces, newline or tabs)

# EXAMPLE 1

- In the two-dimensional Cartesian system, a point A is represented by a pair of numbers (x,y)
- Calculate the distance between to points A and B based on their coordinates

# PROGRAM

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
main()
{
        float xa,ya,xb,yb,d;
        printf("\nInput co-ordinates of point A");
        scanf("%f %f",&xa,&ya);
        printf("\nInput co-ordinates of point B");
        scanf("%f %f",&xb,&yb);
        d= sqrt((xa-xb)*(xa-xb)+(ya-yb)*(ya-yb));
        printf("\nDistance between A and B: %f",d);
        getch();
}
```

# Other Input and Output Functions

***getch***

Reads a single character from standard input.

It requires the user to press enter after entering

***putch***

writes a single character to standard output.

***gets***

reads a line of input into a character array.

gets(name of string)

***puts***

Writes a line of output to standard output.

puts(name of string)

# C CONTROL STRUCTURES

- **Selection**

  **if**

  **if . . . else**

  **switch**

- **Repetition**

  **for loop**

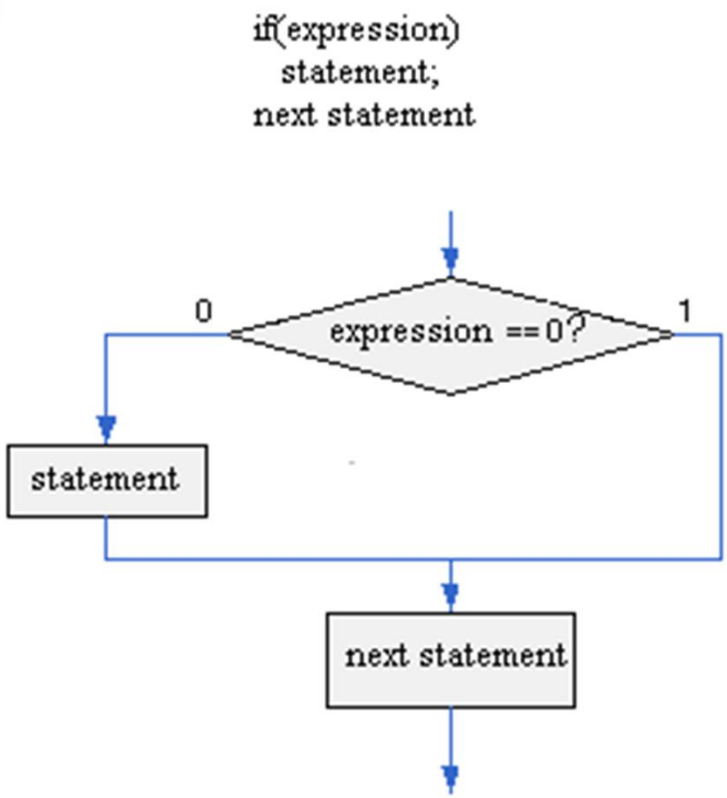
  **while loop**

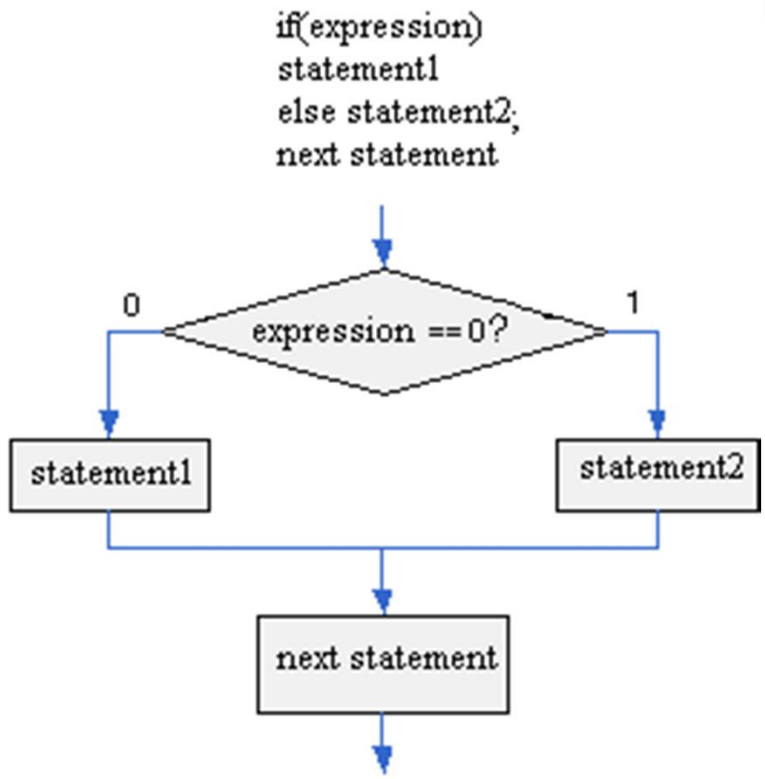
  **do . . . while loop**

# Statements and Blocks

- An expression such as x = 0 or i++ or printf(. . .) becomes a statement when it is followed by a semicolon.
- Block (*compound statement):* group any number of data definitions, declarations, and statements into one statement. Use a block wherever a single statement is allowed.
- We use braces {} to build compound - statements

# IF STATEMENT



# LOGICAL EXPRESSIONS

**Logical expressions may include:**

***6 Relational Operators***

**<  <=  >  >=  ==  !=**

***3 Logical Operators***

**!  &&  ||**

# EXAMPLE 2

- Check whether the year you enter is a leap year or not
- In the Gregorian calendar, A leap year is a year containing one extra day
- Most years that are evenly divisible by 4 are leap years
- Years are evenly divisible by 100 are not leap year unless they are evenly divisible by 400

# EXAMPLE 1

- Write a program to find all the roots of a quadratic equation $ax^2+bx+c=0$
- User interface

| | |
|---|---|
| Enter the coefficient of $x^2$ here | 1 |
| Enter the coefficient of x here | 5 |
| Enter the constant here | 6 |

Root 1: -2
Root 2: -3

Solve Reset

- What happen if the user type 0,0,1 or even 0,0,0?

```
#include <stdio.h>
#include<conio.h>
#include <math.h>
main()
{  float a, b, c, delta, x1,x2;
   puts("Enter coefficients a b c : ");
   scanf("%f%f%f", &a, &b, &c);
     if( a==0)
          {printf("Equation is not quadratic.Become linear eqution %fx+%f=0",b,c);
          if(b==0)
               if (c==0)puts ("Inconsistent liner equation. No root.");
               else puts ("Every number is a root of the equation!");
             else printf ("\nThe unique root of linear equqtion: x= %f",-c/b);
          }
          else
               {delta=b*b-4*a*c;
                if (delta < 0)
                                puts("");
                               else
                                 if (delta == 0)
                                                {x1= -b/(2*a);
                               printf("Quadratic equation has unique root x = %f", x1);
                               }
                               else
                               {x1=(-b + sqrt(delta))/(2*a);
                               x2=(-b - sqrt(delta))/(2*a);
                                    printf("Quadratic equation has 2 roots\n  x1=%f \n  x2=%f",x1,x2 );
                                                }
                               }
}
```

# User screen

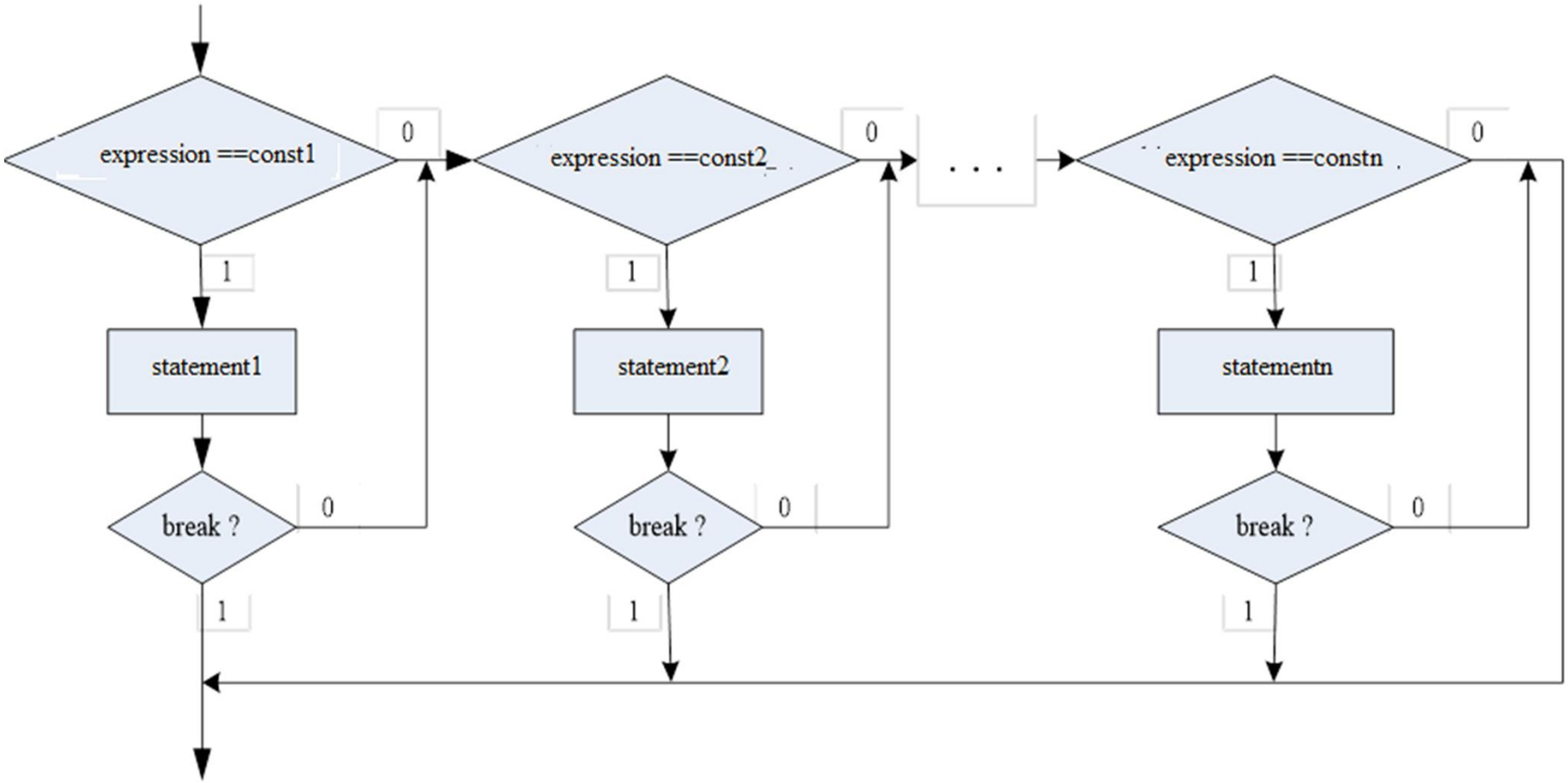
C:\HuongNT\Untitled1.exe

```
Enter coefficients a b c :
1 5 6
Quadratic equation has 2 roots
  x1=-2.000000
  x2=-3.000000
--------------------------------
Process exited after 11.44 seconds with return value 0
Press any key to continue . . .
```

# THE SWITCH STATEMENT

```
switch (expression){
case const1:    statements
case const2:    statements
. . . .
default:        statements
}
```

# THE SWITCH STATEMENT



# EXAMPLE 1

Write a program that asks the user to type a number between 1 and 7 and print the day of week corresponding to that number (1 for Sunday… 7 for Saturday)

# PROGRAM AND USER SCREEN

```
#include <stdio.h>
main(){
        int a;
        printf("\nEnter a number value for a weekday (from 1 to 7): "); scanf("%d",&a);
        switch(a)  {
                case 1: printf("Sunday"); break;
                case 2: printf("Monday"); break;
                case 3: printf("Tuesday"); break;
                case 4: printf("Wednesday"); break;
                case 5: printf("Thursday"); break;
                case 6: printf("Friday"); break;
                case 7: printf("Saturday"); break;
                default: printf("Error");
        }
}
```

C:\HuongNT\Untitled2.exe

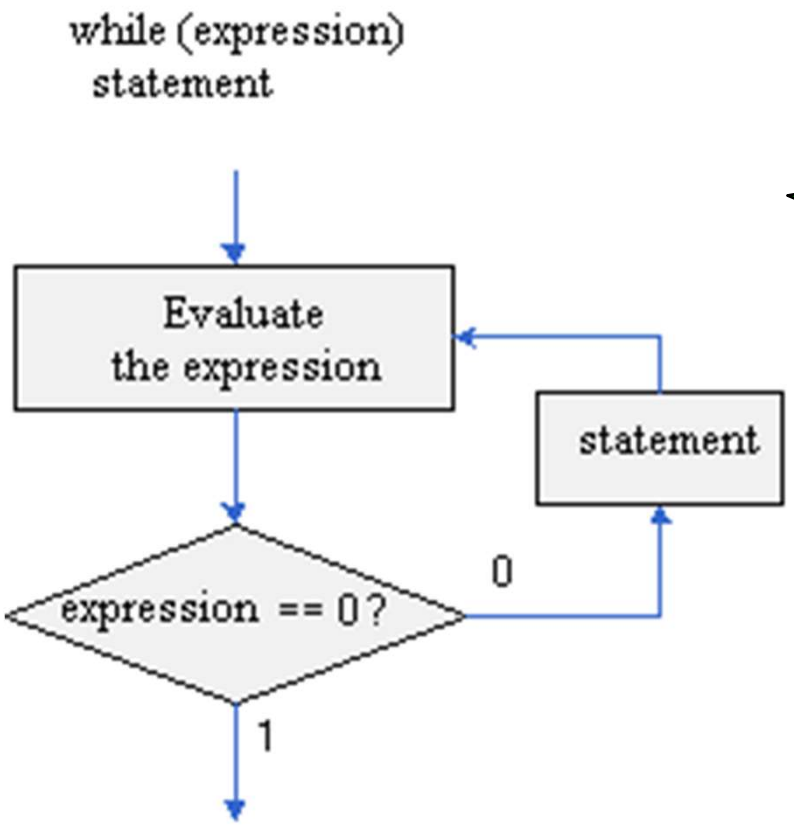
```
Enter a number value for a weekday (from 1 to 7): 3
Tuesday
--------------------------------
Process exited after 6.426 seconds with return value 0
Press any key to continue . . .
```

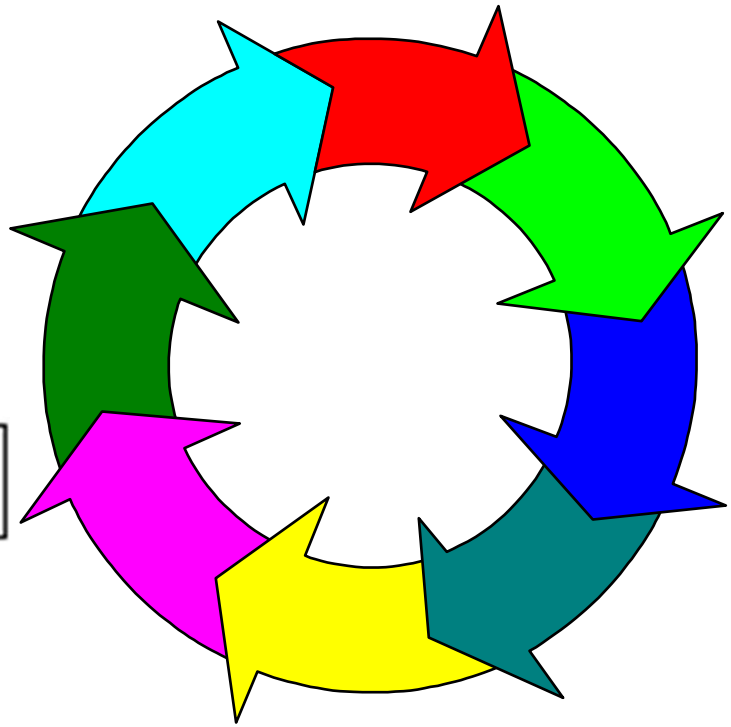
# EXAMPLE 3

- Return the number of days in the month of the Gregorian calendar.
- The program ask the user to type the number of a month and  a year

# THE WHILE STATEMENT

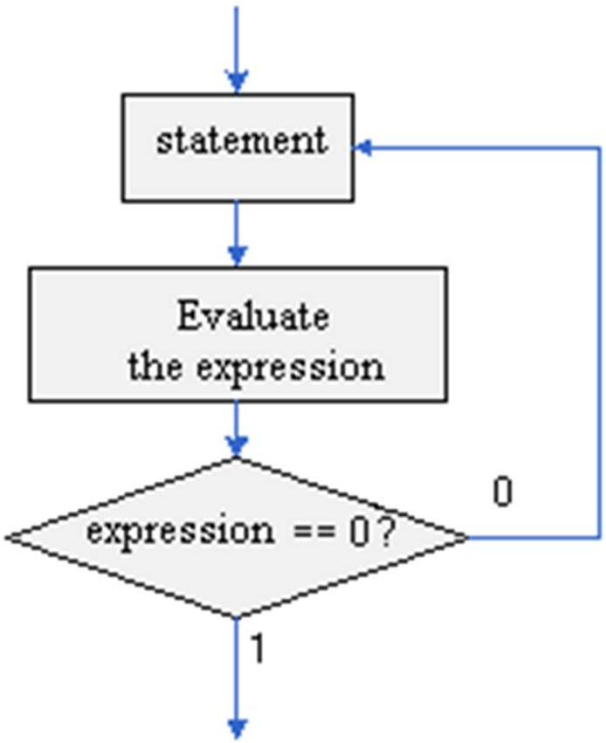
It's a pre-test loop.



# THE DO WHILE STATEMENT

```
do
{statement}
while (expression)
```



# THE FOR STATEMENT

```
for (expression1 ; expression2; expression3)
  statement
```