

While Loop Structure and Do While Loop Structure

While loop structure

“While” loop structure looks simpler than the “for” loop. Although it uses the same element as “for” loop structure but these elements are distributed throughout the program. It has the general form as:

while (Test expression) statements;

The value of the testing variable is checked at the top but all the variation like increment, decrement, comparison etc are taking place within the statements in body of the loop.

Elements of ‘While’ loop

Certainly the expression in the parentheses is simpler than the three part expression of the “for” loop. It dispenses with the initialization and increment expression retaining only the test expression. The initialization step is included in the variable declaration usually while the increment or decrement is present in the body of the loop.

Usage of appropriate loop

One should use the most appropriate loop in the program. The “While” loop shine in the condition where a loop may be terminated unexpectedly by condition developing within the loop. “While” loop are more appropriate than “For” loops when the condition that terminates the loop occur unexpectedly; for example, condition when the testing variable (say a character variable) is input from the keyboard.

Example-1

Example of while loop

```
int count = 0;
while (count < 5)
{
    cout << count << "\n";
    count++;
}
```

Output:

```
0
1
2
3
4
```

Do while loop

The last of the three loop structure in C++ language is the “do while” loop. This loop is very similar to the “while” loop, the difference is that in the “do” loop the test condition is evaluated after the loop is executed rather than before.

General form

The general form of the “do while” loop structure is as:

```
do
{
    Statements;
}
while(expression);
```

Example:

```
#include<conio.h>
#include<iostream>
using namespace std;

void main (void)
{
    int number, sum = 0;
    do {
        cout<<"Enter a number: ";
        cin>>number;
        sum += number;
    }
    while(number != 0);
    cout<<"Total sum = "<<sum;
    getch();
}
```

Output:

```
Enter a number: 3
Enter a number: 3
Enter a number: 2
Enter a number: 3
Enter a number: 0
Total sum = 11
```

Do while operation

The “do” loop execute the statement or statements while the expression is true. It stops when the expression becomes false. The “do” loop is unique because it will always execute the code at least once, since the expression controlling the loop is tested at the bottom off the loop.

The operation of the loop is sort of an upside down version of the “while” loop. The body of the loop is first executed then the test condition is checked therefore the body of the loop is always executed at least once.

Use of do while loop

“Do while” loop is always utilized in conditions where the body of the loop is executed at least once. These are the situations when your program is waiting for some event to occur.

