Do-While Statements loops

Do / while Statements

The do while loop is a variant of the while loop that executes the code block once before checking the condition. Then it will repeat the loop as long as the condition is true. Syntax
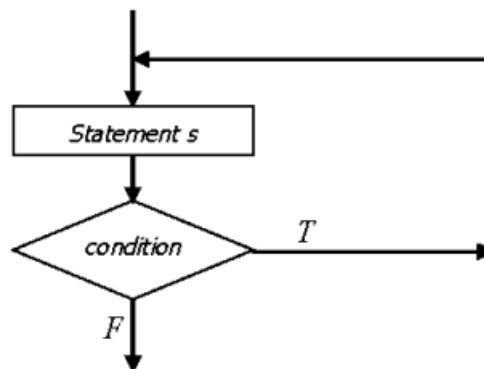
Here's the basic syntax for a do while loop:

do {

   // body of the loop

}

while (condition);

General Form of Do / While statement:

```
do
    statement1 ;
while (condition );
```

```
do
{
    statement1 ;
    statement2 ;
         :
    statement-n ;
}
while (condition );
```

The body of the loop is executed at first. Then the condition is evaluated.

If the condition evaluates to true, the body of the loop inside the do statement is executed again.

The condition is evaluated once again.

If the condition evaluates to true, the body of the loop inside the do statement is executed again.

This process continues until the condition evaluates to false. Then the loop stops.

Example 1:
```
i = 0;
do
    {
        cout << i;
        i ++;
    }
while ( i < 10 )
```
Output:
```
0 1 2 3 4 5 6 7 8 9
       i=10
```

Example 2:
```
i = 0;
do
    {
        cout << i;
        i += 2;
    }
while ( i < 10 )
```
Output: even numbers only
```
0 2 4 6 8
   i=10
```

Example 1: This program will try to print "Hello World" depending on a few conditions.

// C++ program to illustrate do-while loop

#include <iostream>

using namespace std;

int main()

{

        // Initialization expression

        int i = 2;

        do {

                // Loop body

```
        cout << "Hello World\n";
        // Update expression
        i++;
    }
    // Test expression
    while (i < 1);
    return 0; }
```

Output:
Hello World

Example 2:

```cpp
// C++ program to illustrate do-while loop
#include <iostream>
using namespace std;
int main()
{
    // Initialization expression
    int i = 1;
    do {
        // Loop body
        cout << i << endl;
        // Update expression
        i++;
    }
    // Test expression
    while (i <= 5);
    return 0; }
```

Output:
1
2
3
4
5

Here is how the program works.

Iteration	Variable	i <= 5	Action
	i = 1	not checked	1 is printed and i is increased to 2
1st	i = 2	true	2 is printed and i is increased to 3
2nd	i = 3	true	3 is printed and i is increased to 4
3rd	i = 4	true	4 is printed and i is increased to 5
4th	i = 5	true	5 is printed and i is increased to **6**
5th	i = 6	false	The loop is terminated

Example 3: Sum of Positive Numbers Only

```cpp
// program to find the sum of positive numbers
// If the user enters a negative number, the loop ends
// the negative number entered is not added to the sum
#include <iostream>
using namespace std;
int main() {
    int number = 0;
    int sum = 0;
    do {
        sum += number;
        // take input from the user
        cout << "Enter a number: ";
        cin >> number;
    }
    while (number >= 0);
    // display the sum
```

```
    cout << "\nThe sum is " << sum << endl;

    return 0;

}
```

Output 1

Enter a number: 6

Enter a number: 12

Enter a number: 7

Enter a number: 0

Enter a number: -2

The sum is 25

Here, the do...while loop continues until the user enters a negative number. When the number is negative, the loop terminates; the negative number is not added to the sum variable.

Output 2

Enter a number: -6

The sum is 0.

The body of the do...while loop runs only once if the user enters a negative number.

💻 Write C++ program to find the summation of even numbers

```
#include<iostream.h>
void main( )
{
        int max,sum,digit;
        digit=2;
        cout << "enter a number: ";
        cin >> max;
        sum=0;
        do
        {
            Sum=sum+digit;
            Digit+=2;
        }
        while ( digit<=max );
        cout << "2+4+...="<<max<<"sum="<<sum<<endl; }
```

💻 Write C++ program to find the factorial of n:

$$n! = n * n-1 * n-2 * n-3 * ... * 2 * 1$$

```
#include<iostream.h>
void main( )
{
        int n, f = 1;
        cout << "enter positive number: ";
        cin >> n;
        do
        {
            f = f * n;
            n --;
        }
        while ( n > 1 );
        cout << "factorial is: " << f;
}
```

💻 Write C++ program to find the summation of student's marks, and it's average, assume the student have 8 marks:

```cpp
#include<iostream.h>
void main( )
{
        int mark, i, sum = 0;
        float av = 0;
        i = 1;
        do
        {
            cout << "enter mark: ";
            cin >> mark;
            sum = sum + mark;
            i++;
        }
        while ( i <= 8 )
        cout << "sum is: " << sum << endl;
        av = sum / 8;
        cout << "average is: " << av;
}
```

💻 Write C++ program to valid input checking, that accept the numbers between 50 ... 70 only:

```cpp
#include<iostream.h>
void main( )
{
        int accept = 1;
        int x, low = 50, high = 70;
        do
        {
            cout << "enter number: ";
            cin >> x;
            if ( x >= low  &&  x <= high )
                    accept =1;
            else
                    accept = 0;
        }
        while ( ! accept );            ◄────────  while (accept == 1) or
}                                                 while (accept != 0)
```