LAB NO. 01

LOOP, IF ELSE AND SWITCH

Lab Objectives

Following are the lab objectives:

1. Refresh loop concepts
2. If else in C++
3. Switch in C++

Instructions

- This is individual Lab work/task.
- Complete this lab work within lab timing.
- Discussion with peers is not allowed.
- Copy paste from Internet will give you **negative marks**.
- Lab work is divided into small tasks, complete all tasks sequentially.

1. Introduction

Simple program in C++

```
#include<iostream.h>/*Header File*/
int main()/*Main Function*/
{
cout<<"\n*HELLO*\n";
/*Output Statements*/
}
```

C++ DATA TYPES

primary data type	**int, float, char, void**
user defined data type	**structure, union, class, enumerate on**
derived data type	**array, function, pointer, reference**

C++ VARIABLES SCOPE

- A scope is a region of the program and broadly speaking there are three places, where variables can be declared –
- Inside a function or a block which is called local variables,
- In the definition of function parameters which is called
- Formal parameters.
- Outside of all functions which is called global variables.

Local Variable

```
#include <iostream>
using namespace std;
int main() {
       // Local variable declaration:
       int a, b;
       int c;
       // actual initialization
       a = 10;
       b = 20;
       c = a + b;
       cout << c;
       return 0;
}
```

Global Variable

```
#include <iostream>
using namespace std;
// Global variable declaration:
int g;
int main() {
       // Local variable declaration:
       int a, b;
       // actual initialization
       a = 10;
       b = 20;
       g = a + b;
       cout << g;
       return 0;
}
```

C++ for Loops

In computer programming, loops are used to repeat a block of code.

For example, let's say we want to show a message 100 times. Then instead of writing the print statement 100 times, we can use a loop.

That was just a simple example; we can achieve much more efficiency and sophistication in our programs by making effective use of loops.

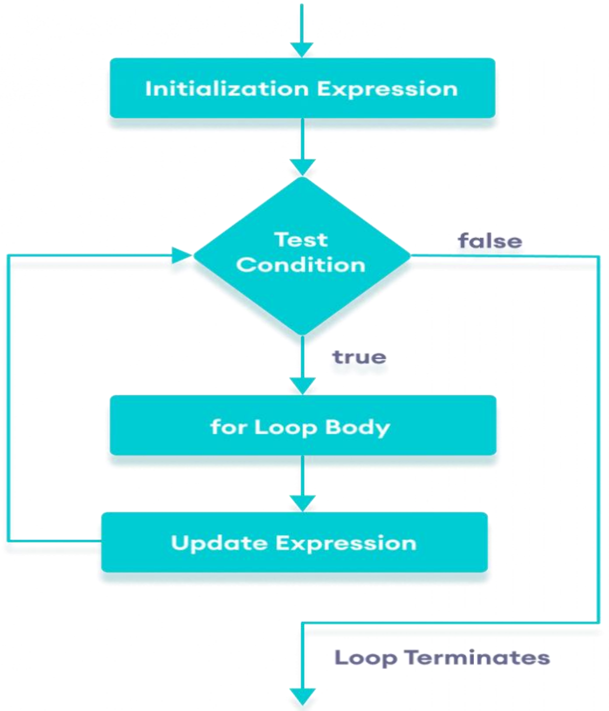
There are 3 types of loops in C++.

1. for loop
2. while loop
3. do...while loop

The syntax of for-loop is:

```
for (initialization; condition; update) {
 // body of-loop
}
```

Flowchart of for Loop in C++



Example 1: Printing Numbers From 1 to 5

```
#include <iostream>

using namespace std;

int main() {
        for (int i = 1; i <= 5; ++i) {
        cout << i << " ";
    }
    return 0;
}
```

Output

```
1 2 3 4 5
```

Example 2: Display a text 5 times

```
// C++ Program to display a text 5 times

#include <iostream>

using namespace std;

int main() {
    for (int i = 1; i <= 5; ++i) {
        cout <<  "Hello World! " << endl;
    }
    return 0;
}
```

Output

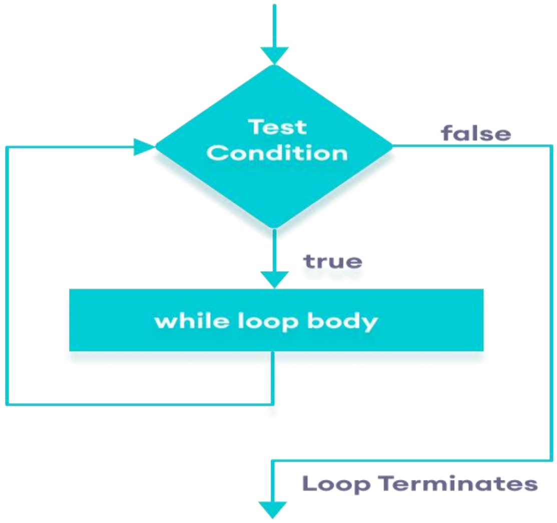
```
Hello World!
Hello World!
Hello World!
Hello World!
Hello World!
```

C++ while Loop

The syntax of the `while` loop is:

```
while (condition) {
    // body of the loop
}
```

Flowchart of while Loop



Example 1: Display Numbers from 1 to 5

```
// C++ Program to print numbers from 1 to 5

#include <iostream>

using namespace std;

int main() {
    int i = 1;

    // while loop from 1 to 5
    while (i <= 5) {
        cout << i << " ";
        ++i;
    }

    return 0;
}
```

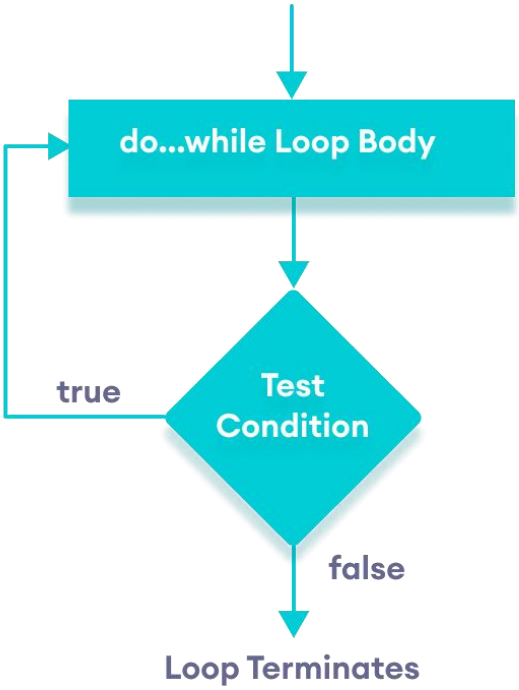
Output

```
1 2 3 4 5
```

C++ do...while Loop

```
do {
   // body of loop;
}
while (condition);
```

Flowchart of do...while Loop



Display Numbers from 1 to 5

```
// C++ Program to print numbers from 1 to 5

#include <iostream>

using namespace std;

int main() {
    int i = 1;
    // do...while loop from 1 to 5
    do {
        cout << i << " ";
        ++i;
    }
    while (i <= 5);

    return 0;
}
```

Output

```
1 2 3 4 5
```

C++ if, if...else and Nested if...else

In computer programming, we use the if statement to run a block code only when a certain condition is met.

For example, assigning grades (A, B, C) based on marks obtained by a student.

if the percentage is above 90, assign grade A

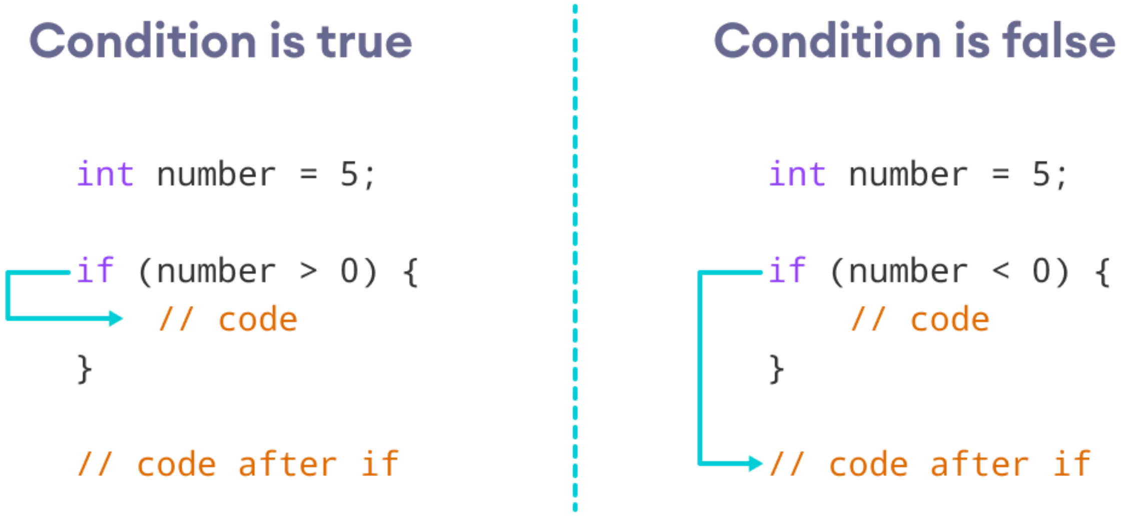
if the percentage is above 75, assign grade B

if the percentage is above 65, assign grade C

C++ if Statement

The syntax of the `if` statement is:

```
if (condition) {
   // body of if statement
}
```



Condition is true

```
int number = 5;

if (number > 0) {
    // code
}

// code after if
```

Condition is false

```
int number = 5;

if (number < 0) {
    // code
}

// code after if
```

C++ if Statement

```
// Program to print positive number entered by the user
// If the user enters a negative number, it is skipped

#include <iostream>
using namespace std;

int main() {
    int number;

    cout << "Enter an integer: ";
    cin >> number;

    // checks if the number is positive
    if (number > 0) {
        cout << "You entered a positive integer: " << number << endl;
    }
    cout << "This statement is always executed.";
    return 0;
}
```

Output

```
Enter an integer: 5
```

```
You entered a positive number: 5
This statement is always executed.
```

C++ switch..case Statement

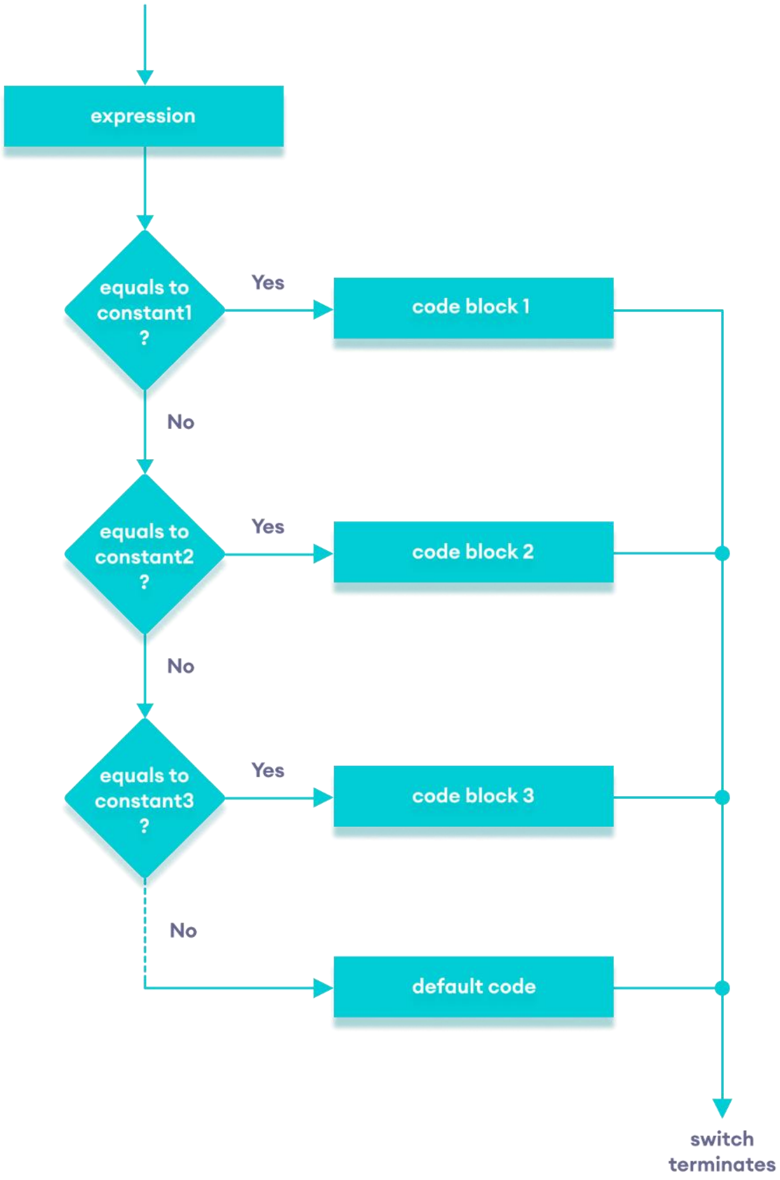
The `switch` statement allows us to execute a block of code among many alternatives.

The syntax of the `switch` statement in C++ is:

```
switch (expression)  {
    case constant1:
        // code to be executed if
        // expression is equal to constant1;
        break;

    case constant2:
        // code to be executed if
        // expression is equal to constant2;
        break;
        .
        .
        .
    default:
        // code to be executed if
        // expression doesn't match any constant
}
```

Flowchart of switch Statement



Create a Calculator using the switch Statement

```
// Program to build a simple calculator using switch Statement
#include <iostream>
using namespace std;

int main() {
    char oper;
    float num1, num2;
    cout << "Enter an operator (+, -, *, /): ";
    cin >> oper;
    cout << "Enter two numbers: " << endl;
    cin >> num1 >> num2;

    switch (oper) {
        case '+':
            cout << num1 << " + " << num2 << " = " << num1 + num2;
            break;
        case '-':
            cout << num1 << " - " << num2 << " = " << num1 - num2;
            break;
        case '*':
            cout << num1 << " * " << num2 << " = " << num1 * num2;
            break;
        case '/':
            cout << num1 << " / " << num2 << " = " << num1 / num2;
            break;
        default:
            // operator is doesn't match any case constant (+, -, *, /)
            cout << "Error! The operator is not correct";
            break;
    }

    return 0;
}
```

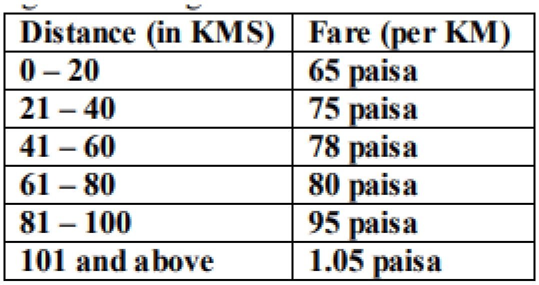
Output

```
Enter an operator (+, -, *, /): +
Enter two numbers:
2.3
4.5
2.3 + 4.5 = 6.8
```

Lab Tasks

i. Write a C++ program to check whether the given number is even or odd.

ii. Write a Program to calculate the fare for the passengers traveling in a bus. When a Passenger enters the bus, the conductor asks “What distance will you travel?” On knowing distance from passenger (as an approximate integer), the conductor mentions the fare to the passenger according to following criteria.



Distance (in KMS)	Fare (per KM)
0 – 20	65 paisa
21 – 40	75 paisa
41 – 60	78 paisa
61 – 80	80 paisa
81 – 100	95 paisa
101 and above	1.05 paisa