National University
of Computer & Emerging Sciences

CL1002-Programming Fundamentals– FALL 2023

LAB 08

Nested Loops

Learning Outcomes

In this lab you are expected to learn the following:

- Nested For loops
- Nested While loops

Loops:

A loop is used for executing a block of statements repeatedly until a particular condition is satisfied. For example, when you are displaying number from 1 to 100 you may want set the value of a variable to 1 and display it 100 times, increasing its value by 1 on each loop iteration instead of writing the print statement 100 times.

There are 3 types of loops in C++.

1. for loop

2. while loop

3. do...while loop

While loop:

In while loop, condition is evaluated first and if it returns true then the statements inside the while loop execute, this happens repeatedly until the condition returns false. When the condition returns false, the control comes out of loop and jumps to the next statement in the program after while loop.

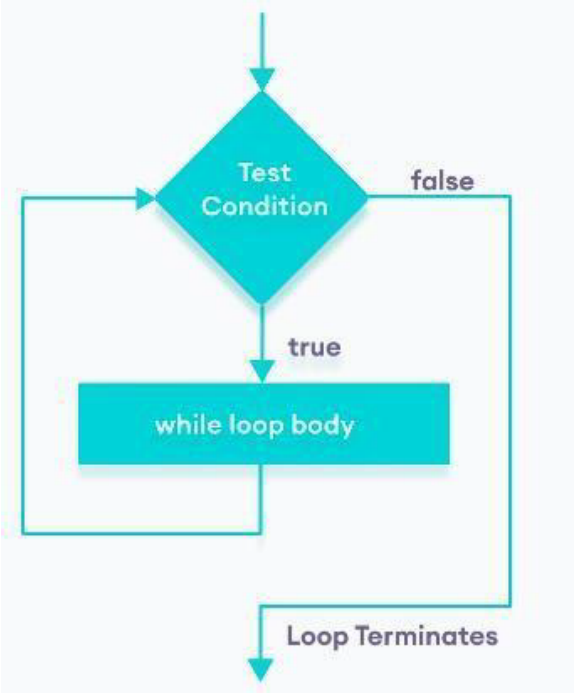
The syntax of the `while` loop is:

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

Here,

- A `while` loop evaluates the `condition`
- If the `condition` evaluates to `true`, the code inside the `while` loop is executed.
- The `condition` is evaluated again.
- This process continues until the `condition` is `false`.
- When the `condition` evaluates to `false`, the loop terminates.

Flowchart of while loop:

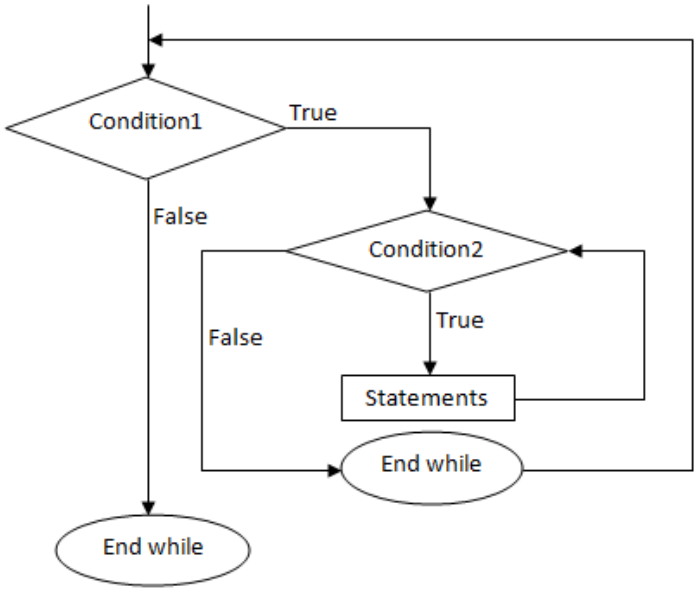


Example 1.1:

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;
}
```

Flow chart for nested while loop



For Loop:

A for loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

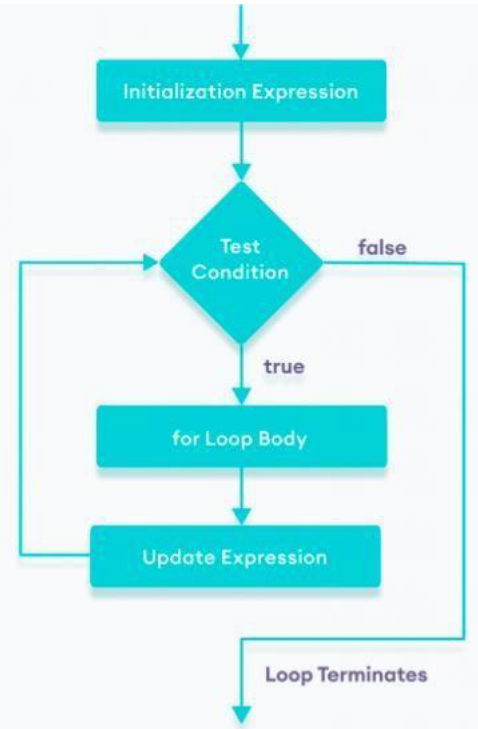
The syntax of for-loop is:

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

Here,

- `initialization` - initializes variables and is executed only once
- `condition` - if `true`, the body of `for` loop is executed
  if `false`, the for loop is terminated
- `update` - updates the value of initialized variables and again checks the condition

Flowchart of For loop:

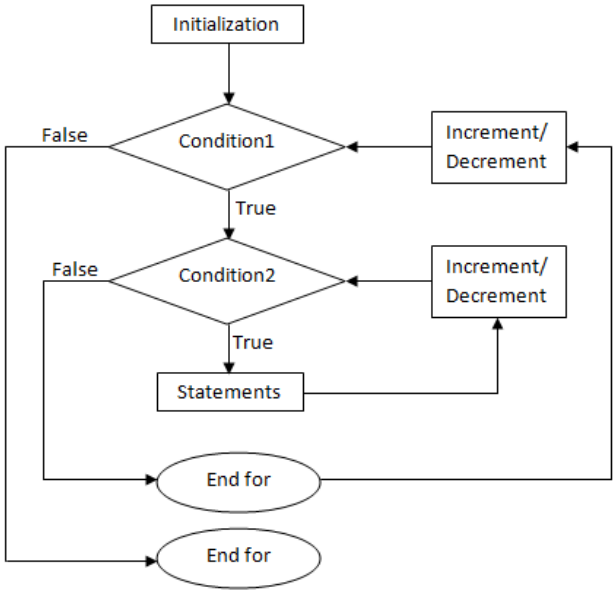


Example 2.1:

Display numbers from 1 to 5

```
#include <iostream>

using namespace std;

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

Flowchart of nested for loop



Example 2.1:

Display numbers from 1 to 50 using nested for loop

```
#include <iostream>

int main() {
    int count = 0;

    for (int i = 1; i <= 5; i++) {
        for (int j = 1; j <= 10; j++) {
            // Calculate the current number
            int number = (i - 1) * 10 + j;

            // Display the number
            std::cout << number << " ";

            }
            std::cout<<"\n";
        }

    return 0;
}
```

```
1  2  3  4  5  6  7  8  9  10
11  12  13  14  15  16  17  18  19  20
21  22  23  24  25  26  27  28  29  30
31  32  33  34  35  36  37  38  39  40
41  42  43  44  45  46  47  48  49  50
```

Submission Instructions:

1. Save the **cpp** file with the roll no and task number
   e.g. i230001_Q1.cpp
2. Now create a new folder with *ROLLNO_LAB01_SEC* **e.g. i23XXXX_LAB08_A**
3. You need to display your roll no and name before the output of each question.
4. Now you have to submit this zipped file on Google Classroom.
5. If you don't follow the above-mentioned submission instructions, you will be marked **zero.**
6. Plagiarism in the Lab Task will result in **zero** marks in the whole category.

Lab Tasks

Solve Tasks 1 to 3 Using Nested For Loop:

Task 01

Write a program that will print the following two patterns on screen using nested loops. Program will ask the user to enter the number of rows for the pattern to be printed

Output:

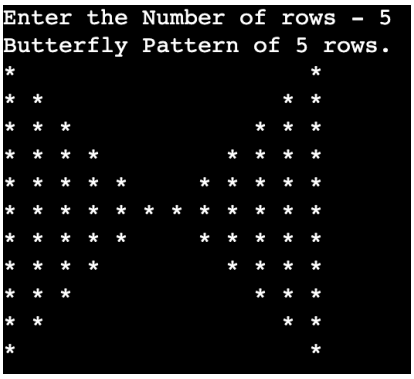
```
Enter the rows of the upside-down triangle: 5
*********
 *******
  *****
   ***
    *
```

```
Enter the number of rows (odd number): 11
     *
    ***
   *****
  *******
 *********
***********
 *********
  *******
   *****
    ***
     *
```

Task 2

Write a program that prints the following butterfly pattern using nested for loops. The program should input the number of rows for the pattern and print accordingly.

Output:



Task 03

Write a program that will calculate the sum of series given an integer n. Ask user for a n and calculate the sum of series till a negative number is entered.

If n=5

Sum= 1/5+2/4+3/3+4/2+5/1

Output:

```
Enter a positive integer (or a negative number to exit): 5
Series: 1 / 5 + 2 / 4 + 3 / 3 + 4 / 2 + 5 / 1
Sum = 8.7
Enter a positive integer (or a negative number to exit): 7
Series: 1 / 7 + 2 / 6 + 3 / 5 + 4 / 4 + 5 / 3 + 6 / 2 + 7 / 1
Sum = 13.7429
Enter a positive integer (or a negative number to exit): -1
Program ended.
```

Task 04

A Palindromic prime is a prime number that is also a palindromic number. Write a program that displays all the palindromic prime numbers between 100 and 999.

Output:

```
Palindromic prime numbers between 100 and 999 are:
101 131 151 181 191 313 353 373 383 727 757 787 797 919 929
```

Task 05

Write a program that that will take a number from the user and calculate the answer using the following method

If number= 453

Answer = 4 * 4 * 4 * 4 + 5 * 5 * 5 * 5 * 5 + 3 * 3 * 3

Output:

```
Enter a number: 453
Answer = 3408
```