

## **Faculty of Computing**

### **CS110: Fundamentals of Computer Programming**

**Class: BESE-16B**

#### **Lab 05: Nested Loops and Control Flow using ‘break’ and ‘continue’**

<b>CLO 1</b>	Understand the syntax and semantics of different programming constructs
<b>CLO 2</b>	Solve given real-world problem by applying appropriate programming concepts and techniques

**Date: 7<sup>th</sup> October, 2025**

**Time: 2:00pm-5:00pm**

**Instructor: Dr. Momina Moetesum**

**Lab Engineer: Mr. Nadeem Nawaz**



### Introduction:

This lab focuses on *nested loops* in C++. Students will learn how to use nested loops by solving pattern printing problems. They will also learn flow control using 'break' and 'continue' within a loop based on specific conditions.

### Learning Objectives:

After completing this section, you will be able to:

- **Understand** the syntax and semantics of nested loops in C++.
- **Apply** nested loops to solve logic-building problems.

### Tools/Software Requirement:

- Microsoft Visual Studio (any version)
- C++ Compiler (integrated within Visual Studio)
- Word Processor (MS Word or equivalent for compiling deliverables)

**Task 1 [CLO 1]:** Print all possible coordinates from (0,0) up to (5,5) using nested loops. Your output must appear in rows and columns as follows. Use escape sequences to format your output as given below:

```
(0,0) (0,1) (0,2) (0,3) (0,4) (0,5)
(1,0) (1,1) (1,2) (1,3) (1,4) (1,5)
(2,0) (2,1) (2,2) (2,3) (2,4) (2,5)
(3,0) (3,1) (3,2) (3,3) (3,4) (3,5)
(4,0) (4,1) (4,2) (4,3) (4,4) (4,5)
(5,0) (5,1) (5,2) (5,3) (5,4) (5,5)
```

**Hint:** The first/outer loop will be till the number of rows and the second/inner loop will be till the number of columns.

**Task 2 [CLO 1]:** Print the following patterns using nested for loops:

Pattern A	Pattern B	Pattern C	Pattern D
123456	1	3	1
1 6	1 2 3	3 3	1 2
1 6	1 2 3 4 5	3 3 3	1 2 3
1 6	1 2 3 4 5 6 7	3 3	1 2 3 4
1 6	1 2 3 4 5 6 8 9	3	1 2 3 4 5
123456			1 2 3 4 5 6

**Task 3 [CLO 2]:** Use loops and break statement to write a C++ program that prompts the user to enter N numbers and prints the **first prime number** it finds.

- The user first enters how many numbers they will input.



**National University of Sciences and Technology (NUST)**  
**School of Electrical Engineering and Computer Science**

- The program should read each number, check if it's prime, and stop immediately (**using break**) when it encounters the first prime number.
- If no prime number is found, print **"No prime numbers in the list."**

**Task 4 [CLO 2]:** Use loops and continue statement to write a program to calculate the **average marks** of a class of students.

- The program should repeatedly ask the user to enter marks (integer between **0 and 100**).
- If the user enters an invalid mark (less than 0 or greater than 100), **skip it** (don't include it in the average) and display a message **"Invalid mark skipped!"**.
- The user can enter **-1 to stop entering marks**.
- Finally, print the average of all valid marks entered.

**Deliverables:**

Compile a single Word document as displayed in solution/answer part and submit this Word file on LMS.



**Lab Rubrics:**

Your Lab 5 will be graded out of 5 for each rubric according to the following rubrics.

Lab Rubrics for Lab 5					
Sr. No.	Assessment	Unacceptable (0 Marks)	Does Not Meet Expectations (1/2 Marks)	Meets Expectations (3/4 Marks)	Exceeds Expectations (5 Marks)
1	<b>Application of Programming Concepts (CLO1, PLO2)</b>	The student did not submit any work. OR The student plagiarized the solution and/or used unfair means.	<p>The student is unable to demonstrate the understanding of syntax of C++ language and is unable to write an executable code.</p> <p>The student is not able to understand the structure of a program at all.</p>	<p>The student demonstrates some understanding of syntax of C++ language and is able to write a code with few errors.</p> <p>The student is able to understand the structure but still learning the syntax.</p>	<p>The student demonstrates good understanding of syntax of C++ language and is able to write executable code without help</p> <p>The student is able to understand the structure and is able to identify problems in the code when introduced.</p>
2	<b>Application of Programming Concepts (CLO2, PLO3)</b>		<p>The student is unable to apply the appropriate programming concepts to solve the given problem thus resulting in an incomplete or ineffective solution.</p> <p>The program flow is messy and incomprehensible.</p> <p>Codes are non-modular and cannot be reused.</p>	<p>The student requires some guidance to apply the appropriate programming concepts to solve the given problem.</p> <p>The program flow requires minor improvements.</p> <p>Codes are semi-modular and semi-reusable.</p>	<p>The student demonstrates a clear ability to apply the appropriate programming concepts to solve the given problem.</p> <p>The program flow is adequate.</p> <p>Codes are modular, reusable, and easily readable.</p>