Programming Fundamentals

BS (CS) _Fall_2025

# Lab_10 Tasks

## Learning Objectives:

1. 1-D Array

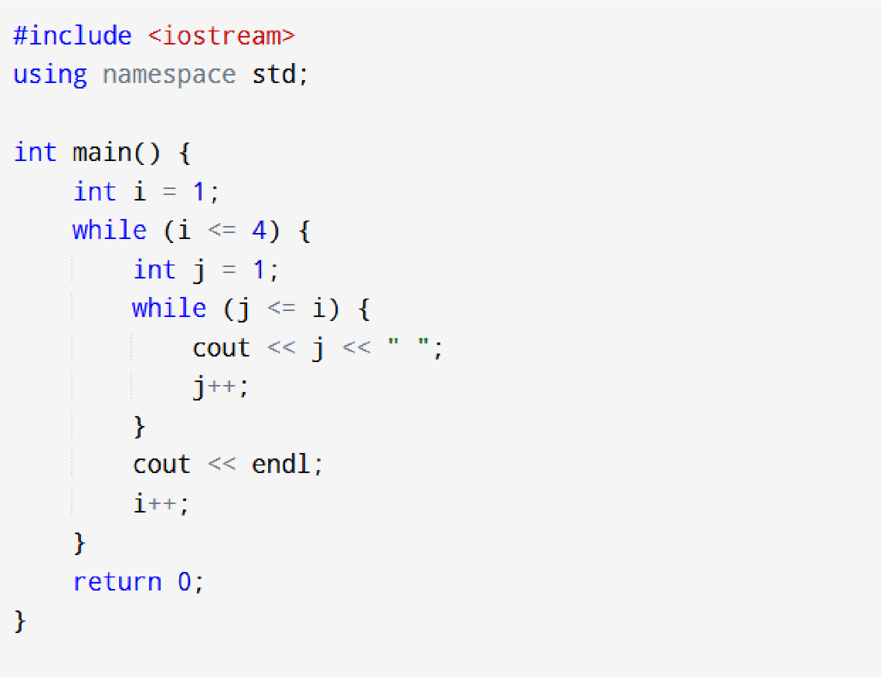
# Lab Tasks

## Submission Instructions

1. Name each Task question as **i25XXXX_Task<NO>** **e.g.** i250000_Task1.cpp
2. Compress all **.cpp** files into a **.zip** file, and name it as *ROLLNO_SEC_LAB10* **e.g. i25XXXX_A_LAB10.**
3. Now you have to submit this zipped file on Google Classroom.
4. If you don’t follow the above-mentioned submission instruction, you will be marked **zero.**
5. Plagiarism in the Lab Task will result in **zero** marks in the whole category.

# Zero Tasks

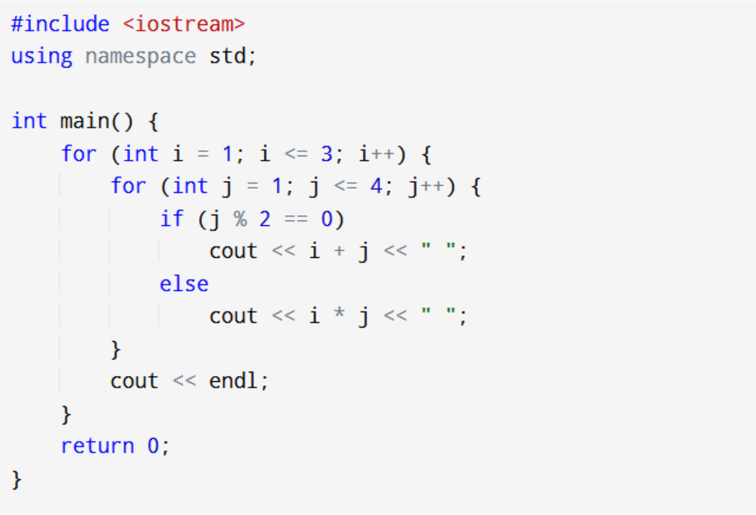
**Q1.** Dry Run the code



```
#include <iostream>
using namespace std;

int main() {
    int i = 1;
    while (i <= 4) {
        int j = 1;
        while (j <= i) {
            cout << j << " ";
            j++;
        }
        cout << endl;
        i++;
    }
    return 0;
}
```

**Q2.** Dry Run the code



```
#include <iostream>
using namespace std;

int main() {
    for (int i = 1; i <= 3; i++) {
        for (int j = 1; j <= 4; j++) {
            if (j % 2 == 0)
                cout << i + j << " ";
            else
                cout << i * j << " ";
        }
        cout << endl;
    }
    return 0;
}
```

# Lab Tasks

**Q3.** Take 5 integers (between 1-9) inputs from user and store them in an array. Again, ask user to give a number.

Now, tell how many times this number appears in array.

**Q4.** Where are N warehouses. The warehouses are located in a straight line and are indexed from 1 to N. Each warehouse contains some number of sacks. A thief decides to rob these warehouses. Thief figured out that he can escape the police if and only if he follows both the following 2 constraints:
He will rob only one continuous segment of warehouses.
He will rob same number of sacks from each warehouse.
Thief wants to calculate the maximum number of sacks he can steal without getting caught by the police. Also tell which segment and which value x give the maximum total sacks stolen? Output the maximum possible total.

**Q5.** Initialize an array with your name. Check how many a's are there in your named

**Q6.** A traveler wants to travel from point A to point B and he has two path options (path1, path2). Find a new path with minimum cost from the beginning of any path to the end of any of the two paths. We can switch from one path to another path only at the common elements.
The path costs are given: □□□*h*1 = [2, 3, 7, 10, 12]□□*h*2 = [1, 5, 7, 8]
Output: 20
Explanation: The path will be 2+3+7+8= 20