**Lab Manual for**

**Data Structures**

Instructor: Ehsaan Ali

Lab # 03

Semester: Spring 2024

A blue circle with white text and a green letter

Description automatically generated

Department of AI & DS

National University of Computer and Emerging Sciences

Islamabad Campus

Carefully read the following instructions

1. Save all .cpp files with your roll no and task number e.g. i21XXXX\_Task03.cpp
2. Save all screenshots of terminal with your roll no and task number.
3. Now create a new folder with NAME\_ROLLNO\_LABNO e.g. XYZ\_i22XXXX\_LAB03
4. Move all your .cpp files to this newly created directory and compress it into .zip file.
5. At the end, when you are done with your lab tasks, make your submission on Google Classroom.

**Tasks**

**Task 1:**

Populate an array of your size choice and write a program to find the largest and smallest element in that array.

**Task 2:**

Define an array of floating points (size must be greater than 5). Get inputs from the user. Calculate and display the product of all floating points made half (e.g., if a value in the array is 3.74 – it will become 1.87) present in the array. To access the elements of array, use the pointer notation.

Input: 5.36, 31.7854, 1.43, 2.2, 62.21

Output: 2.68, 15.8927, 0.715, 1.1, 31.105

Product is: 1041.98487

**Task 3:**

Write a program to compare two arrays of different lengths to find similar values in the

arrays.

e.g

array1 = [2,4,5,6,1]

array2 = [3,6,9,0,23,55,2]

The return should be [2,6]

**Task 4:**

You are familiar with **ArrayList** and have used it in different capacities. The most prominent feature of ArrayList is that it ***expands automatically as elements are added to it***. It can be traversed using a for loop,

In this lab you are going to implement your own ArrayList and its functionalities.

* Create a class **Generic ArrayList**..
* Implement a **constructor** for your ArrayList class.
* Implement **add()** method for adding elements to ArrayList. Remember, ArrayList adds elements dynamically. It doesn't depend on user defined size.
* Implement **remove()** method for removing an element from a specific index. Remember, ArrayList stores elements in a consecutive sequence. If an element is removed from middle you have to re-adjust the elements in the correct sequence.