

Faculty of Computing

CS110: Fundamentals of Computer Programming

Class: BESE-16B

Lab 09: One-Dimensional Arrays, Searching and Sorting

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

Date: 11th November 2025

Time: 2:00pm-5:00pm

Instructor: Dr. Momina Moetesum

Lab Engineer: Mr. Nadeem Nawaz



Introduction:

The objective of this lab is to facilitate students in understanding one-dimensional arrays. In addition to inserting and retrieving items in and from the array, student will also practice sorting and searching array items.

Learning Objectives:

By the end of this lab, students will be able to:

- Organize data into arrays.
- Sort elements of an array.
- Use arrays to solve problems like finding a specific value from a collection.
- Passing an array to a function.

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]:

Write a C++ program that declares an integer array of size 10. Initialize the array elements with random numbers in the range 1-20 and graph the information as a histogram as shown:

Element	Value	Histogram
0	3	***
1	5	*****
2	8	*****
3	11	*****

Task 2 [CLO 2]:

Write a C++ program that declares an array of characters of size 10. Initialize the array elements with random characters including a mix of lower case, upper case, digit, and special characters. The program should traverse the array elements one by one and count how many characters are lower case, upper case, digit, and special character as shown in sample output below.

*Note: Explore <cctype.h> for functions like isdigit(), islower(), etc.

a	w	R	3	%	@	Q	B	%	2
---	---	---	---	---	---	---	---	---	---

Lower case = 2

Upper case = 3

Digit = 2

Special characters = 3

Task 3 [CLO 2]:

Write a C++ function that finds the smallest element in an array of double values using the following header:

double min(double array[], int size)

Check the working of the function by writing a main() driver.



Task 4 [CLO 1]:

Given below is the algorithm for bubble sort that can be used to sort an array. Write a C++ function that implements the algorithm and then test it using a main() driver.

Algorithm 1: Bubble sort

```
Data: Input array A[]
Result: Sorted A[]
int i, j, k;
N = length(A);
for j = 1 to N do
    for i = 0 to N-1 do
        if A[i] > A[i+1] then
            temp = A[i];
            A[i] = A[i+1];
            A[i+1] = temp;
        end
    end
end
```

Task 5 [CLO 1]:

Binary search is used in some cases to improve search from an array. Given below is the algorithm for iterative binary search. Write a C++ function that implements the algorithm and then test it using a main() driver.

*Note: A pre-requisite of binary search is that array should be sorted. Therefore, use a sorted array.

Binary Search Algorithm:

```
BinarySearch(list[], min, max, key)
while min ≤ max do
    mid = (max+min) / 2
    if list[mid] > key then
        max = mid-1
    else if list[mid] < key then
        min = mid+1
    else
        return mid
    end if
end while
return false
```

Deliverables:

Compile a single Word document with codes for each question and screenshots of the outputs and submit this Word file on LMS.



Lab Rubrics:

Your Lab 9 will be graded out of 5 for each rubric according to the following rubrics.

Lab Rubrics for Lab 9					
Sr. No.	Assessment	Unacceptable (0 Marks)	Does Not Meet Expectations (1/2 Marks)	Meets Expectations (3/4 Marks)	Exceeds Expectations (5 Marks)
1	Illustrating the basic understanding of semantics and syntax (CLO1, PLO1)	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	Application of Programming Concepts (CLO2, PLO3)		<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>No Error handling has been performed.</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>