TASK 2

📄 Project Report

Sorting and Searching Algorithms in C++

---

1. Introduction

This project demonstrates simple implementations of sorting and searching algorithms using the C++ programming language. It allows users to input a list of numbers, sort them using **Bubble Sort**, and search for a specific number using **Linear Search**. The project helps in understanding basic algorithmic thinking and array manipulation.

---

2. Algorithms Used

- **Bubble Sort**:
  A simple comparison-based sorting algorithm. It repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order.
  - **Time Complexity**: $O(n^2)$
- **Linear Search**:
  A straightforward search algorithm that checks each element of the array one by one until the desired element is found or the end is reached.
  - **Time Complexity**: $O(n)$

---

3. How the Program Works

1. The user is asked to enter the number of elements and then input the elements themselves.
2. The program sorts the entered elements in ascending order using Bubble Sort.
3. It displays the sorted array.
4. The user is then asked to enter a number to search for.
5. The program uses Linear Search to find the entered number and shows its index if found, or a message saying "Element not found."

---

4. Time Complexity Summary

Algorithm	Best Case	Average Case	Worst Case
Bubble Sort	O(n)	$O(n^2)$	$O(n^2)$
Linear Search	O(1)	O(n)	O(n)

---

5. Conclusion

Through this project, I practiced implementing basic algorithms and using important C++ concepts like arrays, loops, conditionals, and functions. This project helped me understand how algorithm efficiency impacts performance and gave me experience writing structured, clean C++ code.
It also strengthened my skills in getting user input and displaying results in a clear format.