National University of Computer and Emerging Sciences

Lab Manual # 08

Programming Fundamentals

Course Instructor	Ms. Anoosha Khan
Lab Instructor(s)	Junaid Hussain Hassan Masood
Section	
Semester	Fall 2022

Department of Computer Science FAST-NU, Lahore, Pakistan

Instructions:

- Do your tasks individually
- Anyone indulged in the act of plagiarism would be awarded zero
- Pay attention to details, do paper work before starting your code
- Understanding of the question is also part of the lab work **Objectives:**

In this lab, students will study:

- Functions
- Arrays

Searching in arrays

Searching is the process of finding the required data in the array. Searching become more important when the length of the array is large.

Task 1: (1D- Array) Write a program that declare an int array of size 6 inside the main function. Now pass this array and the size to a function *getData()*. The function *getData()* will be used to populate the array by taking input from user. Now pass this array and the size into a function *countEven()* from main function. The function *countEven()* should count and print only even values from the array on console and return the count to main function. Now pass this array, size and two integer variables min and max by reference to a function *findMinMax()*. This function should store the min and max value in the relevant variables. Now prompt the user to enter the value to search and get the input in integer variable "key" inside main function and pass this array, size and key to a function *searchData()*. This function should search the key using linear search technique and return the index. If key isn't available in the array then return -1.

Task 2: (1D-Array)

a) Write C++ program that declare and initialize the global constant integer size with 10 and declare the integer array of given size inside main function. Pass the array to a function *getData()* from main function and fill the array with user input. Since you have declared size as global constant so you don't need to pass size into a function as parameter. Pass this array into a function *sortData()* from main function. You can use any sorting technique to sort the data in ascending order.

b) Now, take that sorted array and searches the value in the array by using function *searchData( )* which must be done the following tasks:

   - First locate the middle element of array and compare with the search number
   - If they are equal, search is successful and the index of middle element is returned
   - If they are not equal, reduces the search to the half of the array

- If the search number is less than the middle element, the first half of the array search, otherwise the second half of the array is search
- The process continues until the required number is found or loop completes without successful search **Output:**

Enter values: 10 2 40 44 1 0 -1 99 34 51

Sorted Array: -1 0 1 2 10 34 40 44 51 99

Enter any number to find: 40

Output: 40 found at index 6