National University of Computer and Emerging Sciences



Laboratory Manual 1

for

Computer Programming

(CL103)

|  |  |
| --- | --- |
| Course Instructor | Mr. Sarim Baig |
| Lab Instructor(s) | Ahsan Rehman  Zara Asif |
| Section | E |
| Semester | Spring 2017 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

Table of Contents

[1.1 Objectives 3](#_Toc411538471)

[1.2 In Lab: Problems / Exercises 3](#_Toc411538472)

[1.2.1 Exercise 1: 3](#_Toc411538473)

[1.2.2 Exercise 2: 4](#_Toc411538474)

[1.2.3 Exercise 3: 4](#_Toc411538475)

[1.2.4 Exercise 4: 5](#_Toc411538476)

[1.3 Post Lab: Problems / Exercises 5](#_Toc411538478)

[1.3.1 Question 1: 5](#_Toc411538479)

[1.3.2 Question 2: 5](#_Toc411538480)

[1.3.3 Question 3: 5](#_Toc411538481)

# 

## Objectives

Topics will cover in this lab

* Revise the C++ concepts of ITC lab.
* Functions , arrays and Sorting

## In Lab: Problems / Exercises

|  |  |
| --- | --- |
| Exercise 1: | **Estimated completion time (mins):**30 |

A bubble sort starts at the top of the list. Each element is compared to the next. If it is greater than the next element, then swap the two. Pass through the list as many times as necessary to sort it. Usually the number of passes required is equal to (**number of elements – 1**). The smallest value bubbles up to the top of the list while the largest value sinks to the bottom.

**8 6 11 3 15 5**

**swap**

**6 8 11 3 15 5**

**okay**

**6 8 11 3 15 5**

**swap**

**6 8 3 11 15 5**

**okay**

**6 8 3 11 15 5**

**swap**

**6 8 3 11 5 15**

**Pass 1:**

**Pass 2:**

**6 8 3 11 5 15**

**okay**

**6 8 3 11 5 15**

**swap**

**6 3 8 11 5 15**

**okay**

**6 3 8 11 5 15**

**swap**

**6 3 8 5 11 15**

**Pass4:**

**3 6 5 8 11 15**

**okay**

**3 6 5 8 11 15**

**swap**

**3 5 6 8 11 15**

**Pass3:**

**swap**

**6 3 8 5 11 15**

**okay**

**3 6 8 5 11 15**

**swap**

**3 6 8 5 11 15**

**3 6 5 8 11 15**

**Scan 5 will do no swapping,**

**so the algorithm terminates**

**after this pass.**

**Pass5:**

**What to Do:**

* Write a C++ function that takes an array and its size .In this function apply bubble sort on the given array.
* Write another C++ function that displays the elements of array.
* Write a main program to test your functions. All functions should be generic and can run on any size of array.

|  |  |
| --- | --- |
| Exercise 2: | **Estimated completion time (mins):**30 |

Take input an array of fixed size n and swap only the odd number values from start and from end.

For example, if we have array of size 6 then:  
1 4 6 2 5 8

1 and 5 are swapped only. As 1 is the first odd value from start and 5 is the first odd value from the end.

|  |  |
| --- | --- |
| Exercise 3: | **Estimated completion time (mins):**30 |

Write a function "print\_pyramid(...)" which takes a single integer argument "height" and displays a "pyramid" of this height made up of of "\*" characters on the screen.

This program prints a 'pyramid' shape of a specified height on the screen.

Enter the height of the pyramid?: **6**

\*\*

\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*

|  |  |
| --- | --- |
| Exercise 4: | **Estimated completion time (mins):**30 |

## Write a C++ program that takes as input 10 numbers and prints the value and the address of the largest and second largest of the numbers from the array.

## Post Lab: Problems / Exercises

### Question 1:

Write a program such that it takes length and breadth of a rectangle and prints the specified rectangle using “\*”. E.g. For Example if length = 5 and width = 4 then output would be

\*\*\*\*\*

\* \*

\* \*

\*\*\*\*\*

### Question 2:

Write your own implementation of the function:

***Double power (double x, int y);***

That returns the value xy for integer y. Make sure that your function works correctly for both positive and negative values of y.

### Question 3:

Follow the Steps below:

Step 1: Input an array of characters *(for example ‘apple’)*

Step 2: Input a character *(for example ‘a’)*

Write a function that finds out the number of occurrences of the character in the input string

Step 3: Input another character *(for example ‘b’)*   
Write a function that replaces all the occurrences of the first character with the second one.