


National University of Computer and Emerging Sciences



Programming Fundamentals CS118 Laboratory Manual

Course Instructor	Waqas Manzoor
Lab Instructor#01	Raja Muzammil
Lab Instructor#02	Muhammad Waqas
Semester	FALL 2021

National University of Computer and Emerging Sciences

	Lab No 7		
	Course Name	Programming Fundamentals	Course Code CS188
	Program	BS(DS)	Semester Fall 2021
	Duration	2.5 hours	Total Points
	Lab Date	05-Nov-2021	Weight 3%
	Section	BDS-1A	Page(s) 9

Topics Covered: Functions

Use Visual Studio Community 2019 or simple IDE available at <https://www.onlinegdb.com/> to write the programs

Submission Instructions:

1. Save all .cpp files according to the following naming convention {ROLLNO}_{TASKNO}.cpp FOR EXAMPLE. 21L-XXXX _P01.cpp, 21LXXXX_P02.cpp, 21L-XXXX_P03.cpp
2. Now create a new folder according to the following naming convention {Section}_{ROLLNO}_{LABNO} e.g. C1_21L-XXXX_L06, C2_21L-XXXX_L06
3. Move all of your .cpp files to this newly created directory and compress it into a single .zip file.
4. Submit this compressed file on Google Classroom.

Functions in C++

C++ Function Declaration

The syntax to declare a function is:

```
returnType functionName (parameter1, parameter2,...) {  
    // function body  
}
```

Here's an example of a function declaration.

```
// function declaration  
void greet() {  
    cout << "Hello World";  
}
```

Here,

- the name of the function is `greet()`
- the return type of the function is `void`
- the empty parentheses mean it doesn't have any parameters
- the function body is written inside `{}`

Note: We will learn about `returnType` and `parameters` later in this tutorial.

Calling a Function

In the above program, we have declared a function named `greet()` . To use the `greet ()` function, we need to call it.

Here's how we can call the above `greet()` function.



(Don't submit code for this problem)

Practice Problem

Write a function named ***power*** that takes two integer parameters ***a*** and ***b***. And returns the power as ***a^b***.

Input Format:

- First line contains input ***a***
- Second line contains input ***b***

Note:

- $a^b = 1$, *if* $b = 0$

Sample Input#1:

12

2

Sample Output#1:

144

Sample Input#2:

3

4

Sample Output#2:

81

Sample Input#3:

15

0

Sample Output#3:

1

Sample Input#4:

-5

3

Sample Output#4:

-125

(Submit codes for questions starting from here)

Problem#01

Write a function named *check palindrome* that takes an integer parameter *number and* displays the corresponding output.

Palindrome: A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam or racecar or the number 10201

For example:

11211 is palindrome

1222 is not a palindrome.

4444 is a palindrome.

Problem#02

Write a function named *rectangle* which takes two integers *height* and *width* as parameters of the function. That function is responsible to draw the rectangular pattern shown below in the Sample Cases.

Note:

- If user enters a non-positive number, display “**Rectangle printing is not possible.**”
- Use of Nested Loops is Mandatory

Sample Input#1:

3

4

Sample Output#1:

OOOO

OOOO

OOOO

Sample Input#3:

2

-7

Sample Output#3:

Rectangle printing is not possible.

Sample Input#2:

9 Sample Input#4:

1 0

12

Sample Output#2:

Sample Output#4:

Rectangle printing is not possible.

O

O

O

O

O

O

O

Problem#03

Write a function named *displayseries* that takes two integer parameter *number* and *rows* and then displays the corresponding **series pattern**.

For user input Starting Number = 5 and Rows = 6,

output should be:

5 * 6 * 7 * 8 * 9 * 0 *

1 * 2 * 3 * 4 * 5 *

6 * 7 * 8 * 9 *

0 * 1 * 2 *

3 * 4 *

5 *

Note: Code should be generic i.e. it should run for any starting number ranging from 0 to 9 and any row number.

Problem#04

Write a program using the C++ programming language to print all the prime numbers between two given numbers by creating a function.

Problem #5

Write a program that inputs a sequence of non-negative numbers terminated by a negative value and show the sum, average, maximum and minimum of the non-negative numbers

Sample Input	Sample Output
10 20 30 40 50 -10	Sum = 150 Average = 30 Maximum = 50 Minimum = 10
13 2 15 5 30 -10	Sum = 65 Average = 13 Maximum = 30 Minimum = 2

