Anil Neerukonda Institute of Technology & Sciences
Department of Computer Science & Engineering (AI & ML, DS)

CSE117 Problem Solving with C

Handout - Lab Session - 8

Functions

Objective:

- To be able to define and call functions in C.
- To be able to use math functions in C.
- To be able to write reusable functions.
- To be able to write recursive functions.

Pre-Lab: Go through the concepts of functions. Write the algorithm and flowcharts for this handout's exercise problems.

During Lab: Solve all the exercise problems. You should work on the additional set of programs only after completing this week's tasks.

Post Lab: Take the quiz.

Read 4.1 to 4.4 & 6.9

Lab Exercises

Design algorithm, flow chart, and program using the data requirements given for the exercise problems and try all the test cases.

Exercise 1: **Math Functions in C**

a. Write a program to find the maximum of two numbers using the max() function in C.
b. Write a program to round a floating-point number to the nearest integer using the round() function in C.
c. Write a program to find the absolute value of a number using fabs() function in C.
d. Write a program to convert an angle from degrees to radians using the radians() function in C.
e. Write a program to calculate the logarithm of a number using log() function in C.
f. Write a program to find the ceiling value of a floating point number using the ceil() function in C.
g. Write a program to find the floor value of a floating point number using the floor() function in C.
h. Write a program to truncate a floating point number using the trunc() function in C.
i. Write a program to find the square root of a number using the sqrt() function in C.
j. Write a program to find the value of x raised to the power y using the pow() function in C.

Exercise 2: rand functions in C

a. Write a program to generate ten random numbers using the rand() function in C. (Temporal random numbers)
b. Write a program to generate the same number series in each run. (you can either omit the srand or provide a constant seed random, preferably a prime number, such as 997.) (Pseudorandom numbers)

Exercise 3: Function Sum

Write a function 'void sum()' that reads two integers from the user and prints their sum. *(a function that doesn't return any value and doesn't take any parameters.)*

Sample Test Cases	**Input**	**Output**
Test Case 1	Enter the numbers : 3 4	Sum = 7
Test Case 2	Enter the numbers: -3 -4	Sum = -7

Exercise 4: Function Factorial

Write a function 'void factorial(int a)' that takes an integer as an argument and print its factorial. *(a function that doesn't return any value but takes parameters.)*

Sample Test Cases	**Input**	**Output**
Test Case 1	Enter the number: 5	Factorial: 120
Test Case 2	Enter the number : 3	Factorial: 6

***Exercise 5:* Function Max**

Write a function 'int max()' that takes three integers from the user and returns the largest of the three. *(a function that returns a value but doesn't take any parameters.)*

Sample Test Cases	**Input**	**Output**
Test Case 1	Enter the numbers : 14 5 9	Largest Number : 14
Test Case 2	Enter the numbers : -14 -5 -9	Largest Number : -5

Exercise 6: **Area of a Right Triangle**

Write a function float area_triangle(int a, int b)' that takes two integers (length of two sides) as arguments and returns the area. *(a function that returns a value and takes parameters.)*

Sample Test Cases	Input	Output
Test Case 1	Enter the lengths of two sides: 2 3	area: 3.00
Test Case 2	Enter the lengths of two sides: -2 3	Invalid, Sides can't be negative.

Exercise 7: Angry Professor (Hackerrank)

Link to the problem on Hackerrank.

https://www.hackerrank.com/challenges/angry-professor/problem?h_r=internal-search

Exercise 8: Factorial using Recursion

Write a C program to calculate the factorial of a number using recursion.

Recursive method for calculating factorial.

$$\text{factorial}(n) = \begin{cases} 1 & \text{if } n = 0 \\ n * \text{factorial}(n-1) & \text{if } n > 0 \end{cases}$$

Sample Test Cases	Input	Output
Test Case 1	Enter the number: 5	Factorial: 120
Test Case 2	Enter the number : 3	Factorial: 6

Exercise 8: GCD using Recursion

Write a C program to find the GCD of two numbers using recursion.

Sample Test Cases	Input	Output
Test Case 1	Enter two numbers: 75 10	GCD: 5
Test Case 2	Enter two numbers: 75 -10	GCD: 5

***Textbook :** B. A. Forouzan and R. F. Gilberg —Cengage Learning , Computer Science: A Structured Programming Approach Using C‖ Third Edition.