

# **Computer Programming Laboratory**

**B.Tech. I Semester**



**Department: Computer Science and  
Engineering**

**Faculty of Engineering & Technology**

**Ramaiah University of Applied Sciences**

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

# Ramaiah University of Applied Sciences

Private University Established in Karnataka State by Act No. 15 of 2013



Faculty	Engineering & Technology
Programme	B. Tech. Civil Engineering B. Tech Computer Science and Engineering B. Tech Electrical and Electronics Engineering
Year/Semester	1 <sup>st</sup> Semester
Name of the Laboratory	Computer Programming Laboratory
Laboratory Code	ESC109A

## List of Experiments

1. Introduction to C programming
2. Variables, types, operators and expressions
3. Decision making and control flow
4. Arrays, multi-dimensional arrays
5. Character and string operations
6. Pointers and User defined functions
7. Structures and dynamic memory allocation
8. Algorithms for sorting and searching
9. Basic data structures

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

**Index Sheet**

<b>No.</b>	<b>Lab Experiment</b>	<b>Viva (6)</b>	<b>Results (7)</b>	<b>Documentation (7)</b>	<b>Total Marks (20)</b>
1	Introduction to C programming				
2	Variables, types, operators and expressions				
3	Decision making and control flow				
4	Arrays and Multi-dimensional arrays				
5	Character and String operations				
6	Pointers and User defined functions				
7	Structures and dynamic memory allocation				
8	Algorithms for sorting and searching				
9	Basic data structures				
10	Lab Internal Test conducted along the lines of SEE valued for 50 Marks and reduced for 20 Marks				
	<b>Total Marks</b>				

**Component 1 (Lab Internal Marks) =****Signature of the Staff In-charge**

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

## Laboratory 1

Title of the Laboratory Exercise: Introduction to C programming

### 1. Introduction and Purpose of Experiment

NetBeans IDE is an open-source integrated development environment. In this laboratory exercise, students get familiar with the NetBeans IDE to edit, compile and run C programs using a set of simple exercises.

### 2. Aim and Objectives

Aim

- To use NetBeans and develop C programs

Objectives

At the end of this lab, the student will be able to

- Explain the features and use of NetBeans IDE to develop C programs
- Edit, compile and execute C programs successfully using NetBeans IDE

### 3. Experimental Procedure

Students are given a set of programs. Programs should be edited, compiled and executed using NetBeans IDE.

### 4. Calculations/Computations/Algorithms

### 5. Presentation of Results

### 6. Analysis and Discussions

### 7. Conclusions

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

## 8. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 2

Title of the Laboratory Exercise: Variables, types, operators and expressions

### 1. Introduction and Purpose of Experiment

Variables are the basic data objects that are manipulated in a program. Operators specify what is to be done to them. Expressions combine variables and constants to produce new values. These building blocks are the topics of this Lab. By solving the given programming problems, the students understand and will be able to apply the concepts of variables, data types, operators and expressions.

### 2. Aim and Objectives

Aim

- To develop programs using variables of basic data types and compute simple expressions involving arithmetic operators

Objectives

At the end of this lab, the student will be able to

- Use variables of the basic data types with proper declarations
- Apply various arithmetic operators in expressions
- Create C programs to solve simple numeric problems

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

4. Questions

- 1) Write a C program to print the size of basic data types int, char, float and double using sizeof () operator.
- 2) Write a C program to swap two numbers using a temporary variable.
- 3) Write a C program to obtain solution of second order quadratic equation.

5. Calculations/Computations/Algorithms

6. Presentation of Results

7. Analysis and Discussions

8. Conclusions

9. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 3

Title of the Laboratory Exercise: Decision making and control flow

### 1. Introduction and Purpose of Experiment

C language provides number of control flow instructions/statements to control the flow of program execution conditionally. These are classified as

- I. Decision making control structures (if-else, switch-case)
- II. Iterative control structures (do-while, while, for)
- III. Jump statements (break, continue)

By solving the problems, students will be able to apply both decision and iterative control statements to control the program execution.

### 2. Aim and Objectives

Aim

- To develop programs involving loops and branching using appropriate C language control statements

Objectives

At the end of this lab, the student will be able to

- Apply control statements such as if-else, nested if-else to express decisions
- Use the switch statement to create multiple branching based on expression matching
- Create C programs using loops such as for, while, do-while to repeat a block of code

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program



Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

vi. Document the Results

vii. Analyse and discuss the outcomes of your experiment

4. Questions

- 1) Write a C program to check the given number is zero, positive or negative.
- 2) Write a C program to count the number of digits in a number.
- 3) Write a C program to create a simple calculator for addition, subtraction, multiplication and division using switch-case statement.

5. Calculations/Computations/Algorithms

6. Presentation of Results

7. Analysis and Discussions

8. Conclusions

9. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 4

Title of the Laboratory Exercise: Arrays and multi-dimensional arrays

### 1. Introduction and Purpose of Experiment

Array is a collection of data items, all of the same type, accessed using a common name. Using arrays we can store linear collections of data. Sometimes we need to store more complex structures such as matrices and tables. This is done using multidimensional arrays. By solving these problems, students will be able use arrays to obtain sum, averages and manipulate matrix to obtain sum, product.

### 2. Aim and Objectives

Aim

- To develop programs using arrays and multi-dimensional arrays

Objectives

At the end of this lab, the student will be able to

- Use arrays of different data types with proper declarations
- Use multi-dimensional arrays to carry out matrix manipulations such as matrix multiplication, matrix transpose

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

#### 4. Questions

- 1) Write a C program to read an array of  $N$  elements. Find the sum and average of elements of an array. Use `#define` to define the size of the array.
- 2) Write a C program that reads an  $M * N$  matrix and display the elements in matrix format. Display the sum of all elements of each row of the matrix.

#### 5. Calculations/Computations/Algorithms

#### 6. Presentation of Results

#### 7. Analysis and Discussions

#### 8. Conclusions

#### 9. Comments

##### 1. Limitations of Experiments

##### 2. Limitations of Results

##### 3. Learning happened

##### 4. Recommendations

## Laboratory 5

Title of the Laboratory Exercise: Character and String operations

### 1. Introduction and Purpose of Experiment

Character types are used to store character value. In C language, a String is an array of characters. The length of a string is determined by a terminating null character: '\0'. Strings can be manipulated using a number of string handling functions. By solving this, students will be able to manipulate character and string data types.

### 2. Aim and Objectives

Aim

- To develop programs using characters and manipulating characters and strings.

Objectives

At the end of this lab, the student will be able to

- Read characters and strings from terminal and print them back
- Create C programs to manipulate characters
- Apply characters in logic of programs

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

4. Questions

- 1) Write a C program to reverse the given string without using string built in function. Compare the obtained string with the original string using strcmp function and justify whether it is palindrome or not.
- 2) Write a C program to find the length of the given string. Apply the strlen function on the same string and verify the result.

5. Calculations/Computations/Algorithms

6. Presentation of Results

7. Analysis and Discussions

8. Conclusions

9. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 6

### Title of the Laboratory Exercise: Pointers and User defined functions

#### 1. Introduction and Purpose of Experiment

A pointer is a variable that contains the address of a variable. By solving these problems students will be able to develop programs using pointers. C allows programmers to define their own function according to their requirement. C also allows declaring variables private or local to functions. This gives scope to variables. By solving these problems, students will be able to create user defined function and change the scope of variables.

#### Aim and Objectives

##### Aim

- To develop programs using pointers and user defined functions

##### Objectives

At the end of this lab, the student will be able to

- Use pointers with proper declarations
- Create C programs using pointers for basic integer utilities
- Apply user defined functions with proper definition and declarations

#### 2. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

3. Questions

- 1) Write a C program to swap two numbers using pointers (use call by reference method).
- 2) Write a C program to find factorial of a number using
  - a. User defined function (pass the value of the number as argument and return the result)
  - b. Recursive function

4. Calculations/Computations/Algorithms

5. Presentation of Results

6. Analysis and Discussions

7. Conclusions

8. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 7

Title of the Laboratory Exercise: Structures and dynamic memory allocation

### 1. Introduction and Purpose of Experiment

Structure is a user-defined data type in C which allows you to combine different data types to store a particular type of record. Students will be able to create records using Structure data type. Dynamic memory allocation allows a program to obtain more memory space, while running or to release space when no space is required. Students will be able to allocate and de allocate memory using malloc, realloc and free functions.

### 2. Aim and Objectives

Aim

- To develop programs using Structures and dynamic memory allocation

Objectives

At the end of this lab, the student will be able to

- Use structures with proper declarations
- Create C programs using structures to store records
- Use **malloc**, **realloc** and **free** functions to dynamically allocate and deallocate memory involving pointers
- Apply appropriate dynamic memory allocation methods in programs

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment



Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

#### 4. Questions

- 1) Write a C program using an array of structures to store records of  $N$  students in a subject. Each record should contain Name of type char, Roll No of type char, and marks of type int. Calculate the average marks of  $N$  students.
- 2) Write a C program to dynamically allocate memory for pointer arrays using malloc() and free(). Assign  $n$  values to the pointer array and find the smallest element in the array.

#### 5. Calculations/Computations/Algorithms

#### 6. Presentation of Results

#### 7. Analysis and Discussions

#### 8. Conclusions

#### 9. Comments

##### 1. Limitations of Experiments

##### 2. Limitations of Results

##### 3. Learning happened

##### 4. Recommendations

## Laboratory 8

Title of the Laboratory Exercise: Algorithms for sorting and searching

### 1. Introduction and Purpose of Experiment

Sorting and merging provide us with means of organising information to facilitate the retrieval of specific data. Searching is the process by which one searches the group of elements for the desired element. Searching methods are designed to take advantage of the organisation of information. By solving these problems, students will be able to use sorting and searching algorithms to sort a randomly ordered set of numbers, and search for key element.

### 2. Aim and Objectives

Aim

- To develop programs for searching and sorting algorithms

Objectives

At the end of this lab, the student will be able to

- Create C programs using searching algorithms
- Create C programs using sorting algorithms

### 3. Experimental Procedure

- Analyse the problem statement
- Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- Implement the algorithm in C language
- Compile the C program
- Test the implemented program
- Document the Results
- Analyse and discuss the outcomes of your experiment

### 4. Questions

- 1) Write a C program to implement Bubble Sort technique.
- 2) Given an element  $x$  and a set of integer elements, find whether  $x$  is present or not in the set using Binary search method

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

5. Calculations/Computations/Algorithms

6. Presentation of Results

7. Analysis and Discussions

8. Conclusions

9. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations

## Laboratory 9

Title of the Laboratory Exercise: Basic data structures

### 1. Introduction and Purpose of Experiment

Data structure is a way of collecting and organising data in such a way that we can perform operations on these data in an effective way. By solving these problems, students will become familiar with the implementations of Stacks and Queues.

### 2. Aim and Objectives

Aim

- To develop programs using array based data structures

Objectives

At the end of this lab, the student will be able to

- Use of appropriate data structure to store data
- Create C programs of basic data structures such as stacks and queues

### 3. Experimental Procedure

- i. Analyse the problem statement
- ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
- iii. Implement the algorithm in C language
- iv. Compile the C program
- v. Test the implemented program
- vi. Document the Results
- vii. Analyse and discuss the outcomes of your experiment

### 4. Questions

- 1) Write a C program to implement Queue operations using arrays.
- 2) Write a C program to implement Stack operations using arrays.

Name: \_\_\_\_\_

Roll Number: \_\_\_\_\_

5. Calculations/Computations/Algorithms

6. Presentation of Results

7. Analysis and Discussions

8. Conclusions

9. Comments

1. Limitations of Experiments

2. Limitations of Results

3. Learning happened

4. Recommendations