I Year I Semester	**L**	**T**	**P**	**C**
Code: 20ES1107	**0**	**0**	**3**	**1.5**

C-PROGRAMMING LAB

Course Outcomes:
The objectives of C Programming lab are:

1. Apply the principles of C language in problem solving.
2. To design flowcharts, algorithms and knowing how to debug programs.
3. To design & develop of C programs using arrays, strings, pointers & functions.
4. To review the file operations, preprocessor commands.

Course Outcomes:
By the end of the lab, the student

- Gains Knowledge on various concepts of a C language.
- Able to draw flowcharts and write algorithms.
- Able design and development of C problem solving skills.
- Able to develop modular programming skills and to trace and debug a program.

Exercise 1:

1. Write a C program to compute the perimeter and area of a rectangle with a height of 7 inches and width of 5 inches.
2. Write a C program to read and display different data type variables.

Exercise 2:

1. Write a C program to calculate the distance between the two points.
2. Write a C program that accepts 4 integers p, q, r, s from the user where r and s are positive and pi seven .If q is greater than rands is greater than pandif the sum of rands is greater than the sum of p and q print "Correct values", otherwise print "Wrong values".

Exercise 3:

1. Write a C program to calculate roots of a quadratic equation.
2. Write a program in C which is a Menu-Driven Program to compute the area of the various geometrical shape.
3. Write a C program to calculate the factorial of a given number.

Exercise 4:

1. Write a program in C to display the first n even natural numbers and their sum.
2. Write a program in C to display the n terms of harmonic series and their sum. 1 +1/2 + 1/3 +1/4 +1/5... 1/n terms.
3. Write a C program to check whether a given number is an Armstrong number or not.

Exercise 5:

1. Write a program in C to print all unique elements in an array.
2. Write a program in C to separate odd and even integers in separate arrays.
3. Write a program in C to sort elements of array in ascending order.

Exercise 6:

1. Write a program in C for multiplication of two square Matrices.
2. Write a program in C to find transpose of a given matrix.

Exercise 7:

1. Write a program in C to print maximum element in each row and each column of a given matrix.
2. Write a program in C to print individual characters of string in reverse order.

Exercise 8:

1. Write a program in C to compare two strings without using string library functions.
2. Write a program in C to copy one string to another string.

Exercise9:

1. Write a program in C to check whether a number is a prime number or not using the function.
2. Write a program in C to get the largest element of an array using the function.
3. Write a program in C to convert decimal number to binary number using the function.

Exercise 10:

1. Write a program in C to demonstrate the use of &(address of) and *(value at address) operator.
2. Write a program in C to add two numbers using pointers.
3. Write a program in C to show how a function returning pointer.

Exercise 11:

1. Write a program in C to add numbers using call by reference.
2. Write a program in C to swap elements using call by reference.

Exercise 12:

1. Write a program in C to find the largest element using Dynamic Memory Allocation.
2. Write a program in C to count the number of vowels and consonants in a string using a pointer.

Exercise 13:

1. Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc() function.
2. Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc( ) function. Understand the difference between the above two programs

Exercise 14:

1. Write a C program to implement the Addition of 2 complex numbers using structure.
2. Write a C program to implement the Subtraction of 2 complex numbers using structure.

Exercise 15:

1. Write a C program to demonstrate Command Line Arguments.

2. Write a program in C to copy a file in another name.

Exercise 16:

1. Write a program in C to append multiple lines at the end of a text file.
2. Write a program in C to remove a file from the disk.