COMPUTER PROGRAMMING LABORATORY

Subject Code	**: 10CPL16 / 10CPL26**	**IA Marks**	**: 25**
Hrs/Week	**: 03**	**Exam Hours**	**: 03**
Total Hrs.	**: 42**	**Exam Marks**	**: 50**

PART – A

1. Design, develop and execute a program in C to find and output all the roots of a given quadratic equation, for non-zero coefficients.

2. Design, develop and execute a program in C to implement Euclid's algorithm to find the GCD and LCM of two integers and to output the results along with the given integers.

3. Design, develop and execute a program in C to reverse a given four digit integer number and check whether it is a palindrome or not. Output the given number with suitable message.

4. Design, develop and execute a program in C to evaluate the given polynomial f(x) = a4x4 + a3x3 + a2x2 + a1x + a0 for given value of x and the coefficients using Horner's method.

5. Design, develop and execute a program in C to copy its input to its output, replacing each string of one or more blanks by a single blank.

6. Design, develop and execute a program in C to input N integer numbers in ascending order into a single dimensional array and perform a binary search for a given key integer number and report success or failure in the form of a suitable message.

7. Design, develop and execute a program in C to input N integer numbers into a single dimensional array, sort them in ascending order using bubble sort technique and print both the given array and the sorted array with suitable headings.

8. Design, develop and execute a program in C to compute and print the word length on the host machine.

PART – B

9. Design, develop and execute a program in C to calculate the approximate value of exp(0.5) using the Taylor Series expansion for the exponential function. Use the terms in the expansion until the last term is less than the machine epsilon defined FLT_EPSILON in the header file <float.h>. Also print the value returned by the Mathematical function exp( ).

10. Design, develop and execute a program in C to read two matrices A (M x N) and B (P x Q) and compute the product of A and B if the matrices are compatible for multiplication. The program must print the input matrices and the resultant matrix with suitable headings and format if the matrices are compatible for multiplication, otherwise the program must print a suitable message. (For the purpose of demonstration, the array sizes M, N, P, and Q can all be less than or equal to 3)

11. Design, develop and execute a parallel program in C to add, element- wise, two one-dimensional arrays A and B of N integer elements and store the result in another one-dimensional array C of N integer elements.

12. Design and develop a function rightrot (x, n) in C that returns the value of the integer x rotated to the right by n bit positions as an unsigned integer. Invoke the function from the main with different values for x and n and print the results with suitable headings.

13. Design and develop a function isprime (x) that accepts an integer argument and returns 1 if the argument is prime and 0 otherwise. The function must use plain division checking approach to determine if a given number is prime. Invoke this function from the main with different values obtained from the user and print appropriate messages.

14. Design, develop and execute a parallel program in C to determine and print the prime numbers which are less than 100 making use of algorithm of the Sieve of Eratosthenes.

15. Design and develop a function reverses (s) in C to reverse the string s in place. Invoke this function from the main for different strings and print the original and reversed strings.

16. Design and develop a function match any (s1,s2) which returns the first location in the string s1 where any character from the string s2 occurs, or – 1 if s1 contains no character from s2. Do not use the standard library function which does a similar job! Invoke the function match any (s1. s2) from the main for different strings and print both the strings and the return value from the function match any (s1,s2).

Note: In the practical examination, the student has to answer two questions. One question from Part A and one question from Part B will be selected by the student by lots. All the questions listed in the syllabus have to be included in the lots. The change of question (Part A only / Part B only / Both Part A& Part B) has to be considered, provided the request is made for the same, within half an hour from the start of the examination. The allotment of marks is as detailed below:

Sl. No.	Activity		Max. Marks
1.	Procedure Writing program & procedure for the assigned problems along with algorithms / flowchart	Part A	5*
		Part B	5*
2.	Conduction Execution of the program and showing the results in proper format	Part A	10
		Part B	20
3.	Viva-voce**		10
Total Max. Marks			50
Minimum passing Marks (40% of Max. Marks)			20