Name	
USN	
Date and Batch	

C PROGRAMMING LAB EXAMINATION

Department of Computer Science & Engineering
Course: Programming in C Laboratory
Duration: 1 Hour 30 Minutes   Total Marks: 10

General Instructions to Students

1. Each student will be assigned one question set (Set No. 1 to 25).

2. Every set contains two programs — both must be completed within the given time.

3. Total Marks: 10

    o Write-up: 4 Marks

    o Execution & Output: 4 Marks

    o Viva voce: 2 Marks

4. You must write the algorithm and program (write-up) neatly in your sheet before using the system.

    o Once the write-up is completed, get it signed by the faculty.

    o Only after faculty verification and signature are you allowed to switch on the system and start typing the code.

5. Any change of program after starting execution will lead to a mark's deduction:

    o 2 marks deduction for changing one program.

    o 5 marks deduction for changing both programs.

6. Use proper comments, indentation, and meaningful variable names.

7. Ensure your program produces the correct output with sample test cases.

8. After execution, show the output to the faculty and get the verification signed.

9. Maintain discipline inside the lab; no communication or sharing of code is allowed.

10. Ensure your roll number and set number are written clearly on the top of your paper.

Mark Split-Up

Component	Description	Marks
Write-up	Algorithm, flow, and proper syntax	/4
Execution	Successful compilation and correct output	/4
Viva voce	Understanding of logic and code explanation	/2
Total		/10 Marks

Student Signature   **Faculty Signature**

Set 1

1. Write a C program to swap two numbers using a temporary variable.

2. Write a C program to calculate simple interest for given principal, rate, and time.

Set 2

1. Write a C program to check whether a number is even or odd.

2. Write a C program to find the factorial of a number using a for loop.

Set 3

1. Write a C program to find the largest of two numbers.

2. Write a C program to find the roots of a quadratic equation.

Set 4

1. Write a C program to find the sum of the first n natural numbers.

2. Write a C program to check whether a number is a palindrome.

Set 5

1. Write a C program to reverse a given integer number.

2. Write a C program to calculate the power of a number (x^n) using a loop.

Set 6

1. Write a C program to check whether a given number is positive, negative, or zero.

2. Write a C program to print the Fibonacci series up to n terms.

Set 7

1. Write a C program to check whether a year is a leap year or not.

2. Write a C program to find the GCD (Greatest Common Divisor) of two numbers using loops.

Set 8

1. Write a C program to find the area and perimeter of a rectangle.

2. Write a C program to perform arithmetic operations (+, −, *, /) based on user input.

Set 9

1. Write a C program to display the multiplication table of a given number.

2. Write a C program that continues performing arithmetic operations until the user enters 'q' to quit.

Set 10

1. Write a C program to find the largest of three numbers.

2. Write a C program to check whether a number is an Armstrong number.

Set 11

1. Write a C program to find the sum of digits of a number.

2. Write a C program to print the Fibonacci series and find the nth Fibonacci term.

Set 12

1. Write a C program to calculate the area and circumference of a circle.

2. Write a C program to find both the GCD and LCM of two numbers.

Set 13

1. Write a C program to print the ASCII value of any character entered by the user.

2. Write a C program to compute factorial using a function.

Set 14

1. Write a C program to find the largest digit in a given number.

2. Write a C program to find the smallest and largest digit in a given number.

Set 15

1. Write a C program to check whether a given character is a vowel or a consonant.

2. Write a C program to check whether a given number is prime or not.

Set 16

1. Write a C program to print the first 10 natural numbers using a loop.

2. Write a C program to print all prime numbers up to n.

Set 17

1. Write a C program to convert temperature from Celsius to Fahrenheit.

2. Write a C program to print a simple pattern:

*

* *

* * *

* * * *

Set 18

1. Write a C program to find the average of three numbers.

2. Write a C program to reverse a number and check if it is a palindrome.

Set 19

1. Write a C program to check whether a number is divisible by 5 and 11 or not.

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

Set 20

1. Write a C program to find the square and cube of a number.

2. Write a C program to calculate the value of x^n using a user-defined function.

Set 21

1. Write a C program to read two numbers and print their sum and difference.

2. Write a C program to implement a calculator that continues until the user enters 'q'.

Set 22

1. Write a C program to check whether a number is multiple of 3 or 7.

2. Write a C program to print Fibonacci series using a function.

Set 23

1. Write a C program to count the number of digits in an integer.

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

Set 24

1. Write a C program to print all even numbers between 1 and 50.

2. Write a C program to find the sum of all odd numbers up to n.

Set 25

1. Write a C program to display the first 10 Fibonacci numbers.

2. Write a C program to find the roots of a quadratic equation and display whether they are real or imaginary.