PROBLEM SOLVING LAB USING C

Course Code: 19CT1102

L	T	P	C
0	0	3	1.5

Course Outcomes:

At the end of the Course the student shall be able to:

CO1: Apply the concepts of variables, data types, operators and expressions.
CO2: Demonstrate the usage of Conditional and Unconditional statements.
CO3: Demonstrate the usage of functions and related functions with respect to arrays and strings.
CO4: Implement the concept of pointers and structures.
CO5: Demonstrate the usage of files and Command Line Arguments.

List of Programs:

Week 1 (Basic Programs)

1. C program to display hello world message.
2. C program to scan all data type variables as input and print it as output.
3. C program to perform arithmetic operations like +,-,*,/,% on two input variables.
4. C program to perform temperature conversions from Centigrade to Fahrenheit and vice versa.

Week 2 (Programs on Operators)

1. C program to scan an input and perform pre and post increment operation on it and display the result.
2. C program to perform all bit wise operations.
3. C program to extract the last two digits of a given integer n, where the number of digits should be greater than 2.
4. C program to display the greatest of three numbers using conditional operator.
5. C program to swap two numbers without using third variable.

Week 3 (Programs on Conditional Statements)

1. C program to check whether a given input integer is in between two values x and y.
2. C program to check whether a given character is a vowel or a consonant or a digit or a special symbol.
3. C program to display the nature and roots of a quadratic equation.
4. C program to perform arithmetic operations using switch statement.
5. C program to convert upper case character to lower case and vice versa.

Week 4 (Programs on Loop Statements)

1. C program to print odd numbers between specified ranges.
2. C program to display the factors of a given number and check whether it is a prime or not.
3. C program to display the sum of individual digits of a given integer raised to the power of n. Also check whether the given integer is Armstrong or not.
4. C Program to demonstrate the usage of unconditional control statements.
5. C program to display the following pattern.

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

Week 5 (Programs on Functions)

1. C program to demonstrate the various categories of functions with respect to return type and number of arguments.
2. C program to find the LCM of two numbers using functions.
3. Create a header file which contains the following prototype:
   i. int factorial (int) ; // non-recursive function
   ii. intfactorial_rec(int); //Recursive function
   iii. int prime (int) ;

   Use the above functions in a C program by including the above header file.
4. C program to display Pascal's triangle using functions.

Week 6 (Programs on Arrays)

1. C program to read n integer values into an array and display them
2. C program to count and display the number of positive, negative, even and odd numbers in a given array of integers and also display their sum.
3. C program to find the smallest and largest numbers in an array of integers.
4. C program to perform addition, multiplication, transpose of given matrices using functions.
5. C program to check whether a given integer exists in a list of numbers and print its index value if it is present, otherwise print "No".

Week 7 (Programs on Strings)

1. C program to convert upper case character to lower case and vice versa in a given string.
2. C program to delete all vowels in a given string and display the remaining string.
3. C program to check whether a given string is palindrome or not.
4. C program that reads two integers as strings and display their sum.

Week 8 (Programs on Strings)

1. C program to demonstrate the usage of at least 10 predefined string handling functions.
2. C program that implements the following user defined string handling functions
   i. To find the length of the given string
   ii. To copy the contents of one string to another
   iii. To reverse the contents of a string
   iv. To compare two strings
   v. To concatenate two strings

Week 9 (Programs on Pointers and Dynamic Memory Allocation)

1. C program to demonstrate the usage of pointers.
2. C program that uses dynamic memory allocation functions to add n elements and display their average.
3. C program that performs pointer arithmetic.
4. C program that implements call by reference.

Week 10 (Programs on Pointers)

1. C program to demonstrate the following
   i. Pointers to Pointers
   ii. Array of Pointers
   iii. Pointer to Array
   iv. Pointers to Functions

Week 11 (Programs on Structures)

1. C program to access and display the members of the structure.
2. C program that demonstrates different ways to access the structure elements using pointers.

Week 12 (Programs on Files)

1. C program to read the contents of a file and display on to output screen.
2. C program to copy the contents of one file to another.
3. C program to count and display the number of characters, words and lines in a file.
4. C program to print last n characters of a file by reading file name and n value from command line.

Programs to be covered beyond syllabus:

1. C program to find the factorial of a given number using recursive and non recursive functions.
2. C program to display the first n terms of the Fibonacci sequence.
   Example: If n= 5 it has to print 0 1 1 2 3

3. Write a general-purpose function to convert any given year into its roman equivalent. The following table shows the roman equivalents of decimal numbers:

Decimal	Roman
1	I
5	V
10	X
50	L
100	C
500	D
1000	M

Example:
Roman equivalent of 1988 is MDCCCCLXXXVIII
Roman equivalent of 1525 is MDXXV

4. C program to display upper and lower triangles of a given matrix.
5. C program to add the sum of row wise elements, column wise elements and diagonal elements of a given square matrix and display the result.
6. C program to check whether the matrix is symmetric or not.
7. Given a positive integer (<= 1000000), find the minimum number of bits required to represent it as a binary number.
8. C program to perform left rotation of the array.
9. C program to implement binary search
10. C program to sort a given list of values using bubble sort.
11. C program to find the LCM of array of integers
12. C program to find the two's complement of a given binary input.
13. C program to replace all the vowels in a given string with a given character
14. C program to perform arithmetic operations using command line arguments
15. C program that writes the contents to a file and reads the contents from a file using structures.

Text Books:

1. Ashok N Kamthane, Amit Ashok Kamthane, *Programming in C*, 3rd Edition, Pearson Publication 2015.
2. HarshaPriya, R. Ranjeet, *Programming and Problem Solving Through "C" Language*, New Edition, Fire Wall Media 2015.
3. Herbert Schildt, *The Complete Reference*, C 4th Edition, Tata McGraw-Hill 2000.

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Languag*e, 2nd Edition , Prentice-Hall, 2006.
2. Rajaraman V, *The Fundamentals of Computer*, 4th Edition , Prentice-Hall of India 2006.
3. Steve Oualline, *Practical C Programming*, 3rd Edition, O'Reilly Press 2006.
4. Jeri R. Hanly, Elliot B. Koffman, *Problem Solving and Program Design in C*, 5th Edition , Pearson Education 2007.
5. Balagurusamy E, *Programming in ANSI C*, 4th Edition, Tata Mcgraw Hill. 82, 2008.
6. Gottfried, *Programming with C*, 3rd Edition , Tata Mcgraw Hill, 2010.
7. R G Dromey, *How to Solve it by Computer*, 1st Edition , Pearson Education 2006.