

## Core II

## Problem Solving with C Programming

### Course Objectives:

- To learn the C programming language to solve different scientific and business problems
- To learn how to design and write effectively codes using various programming constructs available in the C programming language

### Learning Outcomes:

Upon completion of this course, students will be able to:

1. Gain knowledge about different data types and operators in C language
2. Learn the use of various control structures and array
3. Learn the use of pointers, functions, and storage classes
4. Write programs using structures, union, and files

### Unit-I:

Introduction: Introduction to Programming Language, Introduction to C Programming, Keywords & Identifiers, Constants, Variables, Input and Output Operations, Compilation and pre-processing, Data types: Different data types, Data types qualifier, modifiers, Memory representation, size and range, Operators: Operators (Arithmetic, Relational, Logical, Bitwise, Assignment & compound assignment, Increment & Decrement, Conditional), Operator types (unary, binary, ternary). Expressions, Order of expression (Precedence and associativity)

### Unit-II:

Decision Control structures & Loops: Decision Making and Branching statements (Simple IF, IF...ELSE, Nested IF... ELSE, ELSE ... IF ladder), Selection control structure (Switch Statement). Looping statements (FOR, WHILE, DO...WHILE), break, continue and GOTO statements

Array: Concept of Array, Array Declaration, types of array (one and multiple dimension), Character Arrays and Strings, limitation of array.

### Unit-III:

Pointers: Concept of Pointer (NULL pointer, wild pointer, dangling pointer, generic pointer), Pointer Expressions, Accessing the Address of a Variable, Declaring Pointer Variables, Initializations of Pointer Variable, Accessing a Variable through its Pointer, Pointer arithmetic, Pointer representation of array, Array of Pointers, Accessing Sting using Pointer.

Function: Types of Function, Function Declaration, Function Definition, Function Call, Recursive Function, Dynamic Memory Management functions, String handling function (strlen, strcmp, strcpy, strncpy, strcat, strstr).

Storage class: Types (auto, register, static, extern), scope rules, declaration and definition.

#### **Unit-IV:**

Structure and Union: Defining, Declaring, Accessing, Initialization Structure, nested structure, self-referential structure, bit-field, Arrays of Structures, Structures and Functions, structures and pointers, Unions, difference between structure and union, structure within union. File: File Management in C, Defining and Opening a File, File opening modes (read, write, append), Closing a File, File operations, Error handling during I/O Operations, sequential and random access files. Command line arguments.

#### **Text Book:**

*Programming in ANSI C by E. Balagurusamy, TMH*

#### **Reference Books:**

- ✓ *The C Programming Language by B. Kernighan & Dennis Ritchie, PHI.*
- ✓ *C: How to Program by Paul Deitel, Harvey Deitel, Prentice Hall.*
- ✓ *Programming using C by P.C. Sethi & P.K. Behera, Kalyani Publisher.*

### **Core II- Lab: Problem Solving with C Programming**

1. Write a Program to find greatest among three numbers.
2. Write a Program to all arithmetic operation using switch case.
3. Write a Program to print the sum and product of digits of an integer.
4. Write a Program to reverse a number.
5. Write a Program to compute the sum of the first n terms of the following series  
$$S = 1 + 1/2 + 1/3 + 1/4 + \dots$$
6. Write a Program to compute the sum of the first n terms of the following series  
$$S = 1 - 2 + 3 - 4 + 5 - \dots$$
7. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
8. Write a function to find whether a given number is prime or not. Use the same to generate the prime numbers less than 100.
9. Write a Program to compute the factors of a given number.
10. Write a program to swap two numbers.

11. Write a Program to print a triangle of stars as follows (take number of lines from user):

```
*  
***  
*****  
*****
```

12. Write a Program to perform following actions on an array entered by the user:

- a) Print the even-valued elements
- b) Print the odd-valued elements
- c) Calculate and print the sum and average of the elements of array
- d) Print the maximum and minimum element of array
- e) Remove the duplicates from the array
- f) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

13. Write a Program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

14. Write a program that swaps two numbers using pointers.

15. Write a program in which a function is passed address of two variables and then alter its contents.

16. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main( ) function.

17. Write a program to find sum and average of n elements entered by the user. To write this program, allocate memory dynamically using malloc( ) / calloc( ) functions.

18. Write a menu driven program to perform following operations on strings:

- a) Show address of each character in string
- b) Concatenate two strings without using strcat function.
- c) Concatenate two strings using strcat function.
- d) Compare two strings
- e) Calculate length of the string (use pointers)
- f) Convert all lowercase characters to uppercase
- g) Convert all uppercase characters to lowercase
- h) Calculate number of vowels
- i) Reverse the string

19. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

20. Write a program to copy the content of one file to other.