

# 16CS102 COMPUTER PROGRAMMING

Hours Per Week :

L	T	P	C
3	1	2	5

Total Hours :

L	T	P	WA/RA	SSH/HSB	CS	SA	S	BS
48	15	30	5	40	5	8	5	5

## Course Description and Objectives:

This course is aimed at offering fundamental concepts of programming language to the students. It starts with the basics of C-programming and deals with the structure and various attributes required for writing a 'C' program. It also introduces various operators and control statements used in programming. Then it switches to functions and arrays. It goes on with strings, pointers, files & the user defined data types. As a first-level course in computer science, it forms the basis to understand usage of various attributes in writing a program.

## Course Outcomes:

The student will be able to :

- understand the basic terminology used in computer programming to write, compile & debug programs in 'C' language.
- use different data types to design programs involving decisions, loops and functions.
- understand the allocation and Usage of dynamic memory.
- understand the usage of files & structures.

## SKILLS:

- ✓ *Identify suitable data types for an application.*
- ✓ *Apply control statements for decision making problems.*
- ✓ *Use multidimension array for matrix application.*
- ✓ *Design a program to calculate average of a class.*
- ✓ *Analyze the difference between static & dynamic memory allocation.*

**UNIT - 1****L- 10,T-3**

**INTRODUCTION TO C PROGRAMMING:** Structure of C program - Comments, Processor statement, Function header statement, Variable declaration statement and Executable statement; C character set, Constants, Identifiers, Operators, Punctuations, Keywords, Modifiers, Identifiers, Variables, C scopes, Basic data types, Type qualifiers, Storage classes, Reading and writing characters, Formatted I/O.

**UNIT - 2****L- 9,T-3**

**OPERATORS AND CONTROL STATEMENTS:** Operators - Assignment, Arithmetic, Relational, Logical, Bitwise, Ternary, Address, Indirection, Sizeof, Dot, Arrow, Parentheses operators; Expressions - Operator precedence, Associative rules; Control statements - Category of statements, Selection, Iteration, Jump, Label, Expression and Block.

**UNIT - 3****L- 9,T-3**

**FUNCTIONS AND ARRAYS:** Function - Declaration, Prototype, Definition, Calling by value and call by address, Standard library functions and Recursive functions; Array - Declaration, Initialization, Reading, Writing, Accessing and Passing as a parameter to functions, 2D-arrays, Multidimensional arrays.

**UNIT - 4****L- 9,T-3**

**STRINGS AND POINTERS:** Strings - Declaration, String library functions, Array of strings, Command line arguments; Pointers - Declaration, Initializing pointers, Multiple indirection, Relationship between arrays and pointers; Scaling up - Array of arrays, Array of pointers, Pointer to a pointer, Pointer to an array; Pointer to functions, Dynamic memory allocation functions.

**UNIT - 5****L- 8,T-3**

**STRUCTURES AND FILES:** Structures - Declaration, Initialization and accessing, Array of structures and passing structures to functions, Structure pointers, Arrays and structures within structures, Unions, Bit-fields, Types and enumerations; Files - I/O and processing operations on text and binary files; Pre-processor directives.

**LABORATORY EXPERIMENTS****Course Outcomes:**

Upon successful completion of this course, the student will be able to:

- write, compile and debug programs in C language.
- formulate problems and implement algorithms in C.
- develop programming components that efficiently solve computing problems in real-world.

**LIST OF EXPERIMENTS**

Total hours-30

1. Compute the factors of a number.
2. Compute the average of 'n' numbers.
3. Find whether a number is palindrome or not.
4. Find whether a number is a power of 2 or not.
5. Compute the factorial of a number.
6. Implement any kind of operation (+,-,\*,/,%) using a switch case.
7. Swap two values using call by value and call by reference.
8. Using structure of arrays.

**ACTIVITIES:**

- *Implement matrix operations.*
- *Implement malloc and calloc functions.*
- *Copy the content of one file into the other.*
- *Implement string manipulations functions.*

9. Find the reversal of a number.
10. Find the frequency of each number in the array.
11. Which takes 0's & 1's as input and the array should consist of all 0's first and then 1's.
12. Copy the first 10 words of a file into the other file.
13. Count the number of words in a file.
14. Create a structure which stores the student's information in a class.
15. Reverse the contents of the array.
16. Implement pointer of pointers.
17. Give  $n^{\text{th}}$  term of the Fibonacci number.
18. Find the factorial of a number using recursion.
19. Find the number of vowels in a file.
20. Access the structure and union members.

**TEXT BOOK:**

1. Ajay Mittal, "Programming in C - A practical Approach", 1<sup>st</sup> edition, Pearson Education, India, 2015.

**REFERENCE BOOKS:**

1. Reema Thareja, "Introduction to C Programming", 2<sup>nd</sup> edition, Oxford University Press India, 2015.
2. Herbert Schildt, C, "The Complete Reference", 4<sup>th</sup> edition, Tata McGraw-Hill, 2000.
3. E. Balagurusamy, "Programming in ANSI C", 4<sup>th</sup> edition, Tata McGraw- Hill, 2008.