CONCEPTS OF COMPUTER & PROGRAMMING IN ‘C’ (ICS-101/201)

L	T	P
3	1	0

Unit – I

Introduction To Computers: Computer hardware Components, peripherals and their functions, Number Systems and conversion methods, Concept of an algorithm; termination and correctness. Algorithms to programs: specification, top-down development and stepwise refinement. Introduction to Programming Environment, Use of high level programming language for the systematic development of programs. Introduction to the design and implementation of correct, efficient and maintainable programs, Structured Programming, Trace an algorithm to depict the logic.

Unit – II

Basic operating System Concepts: Introduction of MS-DOS, WINDOWS, and LINUX Operating Systems, Functional Knowledge of these operating systems. Introduction of Basic Commands of LINUX and Editors, Managing Files and Directories in LINUX, Programming Environment in LINUX, Writing and executing programs in LINUX.

Unit – III

Programming in C: History, Introduction to C Programming Languages, Structure of C programs, compilation and execution of C programs, Debugging Techniques, Data Types and Sizes, Declaration of variables, Modifiers, Identifiers and keywords, Symbolic constants, Storage classes (automatic, external, register and static), Enumerations, command line parameters, Macros, The C Preprocessor.

Unit – IV

Operators: Unary operators, Arithmetic & logical operators, Bit wise operators, Assignment operators and expressions, Conditional expressions, Precedence and order of evaluation.

Control statements: if-else, switch, break, and continue, the comma operator, goto statement.

Loops: for, while, do-while. **Functions:** built-in and user-defined, function declaration, definition and function call, and parameter passing: call by value, call by reference, recursive functions, Multi file programs. **Arrays:** linear arrays, multidimensional arrays, passing arrays to functions, Arrays and strings.

Unit – V

Structure and Union: definition and differences, self-referential structure. **Pointers:** value at (*) and address of (&) operator, pointer to pointer, Dynamic Memory Allocation, calloc and malloc functions, array of pointers, function of pointers, structures and pointers. **File Handling**

in C: opening and closing a data file, creating a data file, read and write functions, unformatted data files.

Text and References Books:

1. Kernighan, Ritchie, "The C Programming Language", PHI
2. V. Rajaraman, "Fundamentals of Computers", PHI
3. Peter Norton's, "Introduction to Computers", TMH
4. Gottfried, "Programming in C", Schaum's Series, Tata McGraw Hill
5. Yashwant Kanitkar, "Working with C", BPB
6. E. Balagurusamy, "Programming in ANSI C", TMH

PROGRAMMING LAB (ICS-151/251)

L T P

0 0 3

1. Write C program to find largest of three integers.
2. Write C program to check whether the given string is palindrome or not.
3. Write C program to find whether the given integer is
   (i). a prime number
   (ii). an Armstrong number.
4. Write C program for Pascal triangle.
5. Write C program to find sum and average of n integer using linear array.
6. Write C program to perform addition, multiplication, transpose on matrices.
7. Write C program to find fibbonacci series of iterative method using user-defined function.
8. Write C program to find factorial of n by recursion using user-defined functions.
9. Write C program to perform following operations by using user defined functions:
   (i) Concatenation
   (ii) Reverse
   (iii) String Matching
10. Write C program to find sum of n terms of series:
    n – n*2/2! + n*3/3! - n*4/4! + ...
11. Write C program to interchange two values using
    (i). Call by value.
    (ii).Call by reference.
12. Write C program to sort the list of integers using dynamic memory allocation.
13. Write C program to display the mark sheet of a student using structure.
14. Write C program to perform following operations on data files:
    (i) read from data file.
    (ii) write to data file.
15. Write C program to copy the content of one file to another file using command line argument.