20XC15 PROBLEM SOLVING AND C PROGRAMMING 4 0 0 4

PROBLEM SOLVING: Introduction to Problem Solving - Program development - Analyzing and Defining the Problem - Algorithm - Flow Chart - Programming Languages - Types of programming languages - Program Development Environment. (5)

C LANGUAGE: Introduction to C Language - C Character Set - Identifiers and Keywords - Data Types – Literal Constants - Variables – l-value - r-value - Qualifiers – Modifiers - Operators and Expressions – Type conversions - Library Functions - Data Input and Output Functions – escape sequence characters – Formatted input and output. (6)

CONTROL STATEMENTS: Making Decisions: If Statement – If/else Statement - If/else if Statement – Nested if Statements – dangling else - Switch Multiple Selection Statement– Repetition: Repetition Essentials - While Loop – do-While Loop – For Loop – Nested Loops – Breaking out of a Loop Continue statement – goto Statement. (6)

FUNCTIONS: Modular Programming – Function Prototypes - Defining and Calling Functions – Function Call Stack and Activation Records - Passing Arguments to Functions – Returning a value from a function - Recursion – Recursion vs. Iteration – Scope and lifetime of variables – Memory layout of a C program - Storage Classes - Auto - Static - Extern and Register Variables. (8)

ARRAYS: Defining Array – Array Initialization - Accessing array elements - Processing arrays - Arrays as function arguments - Multidimensional arrays – Memory address calculation of an array – Row major and column major order - String Handling. (8)

POINTERS: Pointer Variable Definitions and Initializations – Passing Arguments to Functions by address – Pointer Expressions and Pointer Arithmetic - Relationship between Pointers and Arrays - Pointers and multidimensional arrays –Constant Pointer – Pointer to Constant – NULL pointer- dangling pointers - Pointers to functions - passing functions to other functions – Introduction to Stack and Heap Memory - Dynamic Memory Allocation. (10)

STRUCTURES AND UNIONS: Structure Definitions – Initializing Structures – Accessing Structure Members - Processing a structure - typedef- Structures and pointers - Passing structures to functions – Self-Referential Structures - Bit fields - Unions – Enumeration Constants. (8)

FILES: Files and Streams - Operations on Files – Types of Files, Various Read and Write Functions for Sequential - Access and Random-Access Files - Command Line Arguments. (5)

PREPROCESSOR DIRECTIVES: #include Preprocessor Directive - #define Preprocessor Directive: Symbolic Constants - #define Preprocessor Directive: Macros - Conditional Compilation (4)

Total L: 60

TEXT BOOKS:

1. Brian W. Kernighan and Dennis Ritchie, “The C Programming Language”, Pearson Education India, 2015
2. R G Dromey, “How to solve it by Computer”, Pearson, 2008.

REFERENCES:

1. Herbert Schildt, “C The Complete Reference", McGraw Hill, 2017.
2. Gottfried B, “Programming with C”, McGraw Hill, 2011.
3. Peter Prinz and Tony Crawford, “C in a Nutshell”, O'Reilly, 2016.

20XC17 C PROGRAMMING LAB **0 0 4 2**

1. Simple programs to understand the concepts of data types.
2. Familiarizing conditional, control and repetition statements.
3. Usage of single and double dimensional arrays including storage operations.
4. Implementation of functions, recursive functions.
5. Defining and handling structures, array of structures and union.
6. Implementation of pointers, operation on pointers dynamic storage allocation.
7. Creating and processing data files.

Total P: 60