CSE101:COMPUTER PROGRAMMING
L:2   T:0   P:2   Credits:3

Course Outcomes: Through this course students should be able to

- Discuss the various approaches towards solving a particular problem using the C language constructs
- Write programs to solve different problems using C constructs irrespective of the compilers
- Plan the process of code reuse by forming a custom library of one's own functions
- Complete the understanding and usage of one of the building blocks of data structures namely pointers
- Categorize the theoretical knowledge and insights gained thus far to formulate working code
- Assess, validate and restate the underlying logic and formulate code which is capable of passing various test cases

Unit I
Basics and introduction to C : Program development in C, structured programming using algorithm and flow chart, The C character set, Identifiers and keywords, Data types, Constants and variables, Expressions, Arithmetic operators, Unary, Relational, Logical, Assignment and conditional operators, Bitwise operators.

Unit II
Control structures and Input/Output functions : If, If else, Switch case statements, While, For, Do-while loops, Break and continue statements, Goto, Return, Type conversion and type modifiers, Designing structured programs in C, Formatted and unformatted Input/Output functions like printf(), Scanf(), Puts(), Gets() etc.

Unit III
User defined functions, Storage classes: Function prototypes, Function definition, Function call including passing arguments by value and passing arguments by reference, Math library functions, Recursive functions, Scope rules (local and global scope), Storage classes in C namely auto, Extern, Register, Static storage classes.

Unit IV
Arrays in C : Declaring and initializing arrays in C, Defining and processing 1D and 2D arrays, Array applications, Passing arrays to functions, inserting and deleting elements of an array, Searching including linear and binary search methods, Sorting of array using bubble sort.

Unit V
Pointers, Dynamic memory allocation: Pointer declaration and initialization, Types of pointers - dangling , wild, null, generic (void), Pointer expressions and arithmetic, Pointer operators, Operations on pointers, Passing pointer to a function, Pointer and one dimensional array, Dynamic memory management functions (malloc, calloc, realloc and free),

Unit VI
Strings, User defined types including structures and unions : Defining and initializing strings, Reading and writing a string, Processing of string, Character arithmetic, String manipulation functions and library functions of string, Declaration of a structure, Definition and initialization of structures, Accessing structures, Structures and pointers, Nested structures, Declaration of a union.

List of Practicals / Experiments:

Basics and introduction to C
- Programs to explore different data types and usage.
- Programs for different type of operators and the usage.

Control structures and Input/Output functions
- Programs on decision making constructs as if, if else and switch.
- Programs on formatted and unformatted functions as printf(),scanf(),gets() and puts().

User defined functions, Storage classes
- Program to explore different prototypes.
- Program to differentiate between call by value, call by address.
- Program to demonstrate storage classes as auto, register, extern, static.

Arrays in C
- Program to demonstrate memory arrangement of 1D and 2D array.
- Program to demonstrate operations on array as insertion, deletion, searching (linear, binary).
- Program to demonstrate bubble sort.

Pointers, Dynamic memory allocation
- Program to demonstrate type of pointers.
- Program to demonstrate pointer vs array name.
- Program to demonstrate dynamic memory management functions (malloc(),calloc(),realloc() and free()).

Strings, User defined types including structures and unions
- Program to demonstrate string operations.
- Program to demonstrate structure and nested structures.
- Program to differentiate between structure and union.

Text Books:

1. PROGRAMMING IN C by ASHOK N. KAMTHANE, PEARSON

References:

1. PROGRAMMING IN ANSI C by E. BALAGURUSAMY, MCGRAW HILL EDUCATION

2. C HOW TO PROGRAM by PAUL DEITEL AND HARVEY DEITEL, PRENTICE HALL