# CSE231:COMPETITIVE PROGRAMMING

L:0 T:0 P:2 Credits:1

**Course Outcomes:** Through this course students should be able to

CO1 :: relate the theoretical as well as practical knowledge to form an amalgamation of working code

CO2 :: employ a working combination of solution to ubiquitous problems which are time and space efficient

CO3 :: identify  and comprehend the inner workings behind the design of an optimal solution

CO4 :: illustrate the usage of algorithms and data structures in the design of an optimal solution towards a problem

CO5 :: recall the knowledge obtained from various algorithmic paradigms to formulate optimal solutions to real world problems

CO6 :: examine and utilise knowledge to build and design reliable code which is capable of passing various test cases

## List of Practicals / Experiments:

### Strings and Linear Data Structures

- Longest substring problem
- Reverse the individual words of the string
- Decimal number to Roman Numeral
- Longest Palindromic substring
- Implement a stack using one queue
- Find kth largest element
- Detect duplicate parenthesis
- Reverse a stack problem
- Print minimum element of the array using recursion
- Last non zero digit of the factorial
- Find symmetric pairs problem
- Pascal triangle problem
- Two people meet each other
- Next Largest Number problem
- Minimum swaps to get the Sorted Array
- Print smallest k elements in same order in an array

### Binary Trees and Binary Search Trees

- Left view of a binary tree
- Flatten a binary tree
- Lowest common ancestor in a BST
- Spiral order traversal in a binary tree
- Merge K sorted lists using heaps
- Maximum sum pairs problem using heaps

**Hashing and Hashtables**

- Four elements such that a+b=c+d using sets and hashtables
- Print K most frequent numbers using hashing
- Shortest substring of all distinct characters

**Graphs and Shortest Distance between every pair of vertices**

- Check whether the graph is Bipartite
- Shortest distance between every pair problem

**Two dimensional arrays and Linked Lists**

- Implement own arraylist using arrays
- Print the matrix in spiral form
- Last nth node of a linked List
- Remove a loop in singly linked list
- Subtract two numbers
- Flatten the given linked list

**References:**

1. DATA STRUCTURES AND ALGORITHMS : CONCEPTS, TECHNIQUES AND APPLICATIONS by G. A. V. PAI, Mc Graw Hill Education
2. CRACKING THE CODING INTERVIEW by GAYLE LAAKMANN MCDOWELL, CAREERCUP