

B.Tech CSE (all Specializations) Semester III/BCA,B.SC III Semester

Department		School of Engineering Technology	
Course Name: Competitive Coding-I	Course Code	L-T-P	Credits
		2-0-0	NIL
Type of Course:	Audit Course		
Prerequisite(s), if any: Fundamentals of programming			
Course Objective: To enhance students' problem-solving abilities in competitive coding by providing in-depth knowledge of core data structures, algorithms, and efficient coding techniques. This course aims to prepare students for technical assessments and coding interviews, building a strong foundation for tackling real-world coding challenges.			
Course Outcomes: <ul style="list-style-type: none">• Apply fundamental and advanced coding techniques to solve problems involving arrays, strings, recursion, matrices, and linked lists.• Analyze and implement efficient data structure operations, including stacks, queues, and their real-world applications in competitive programming.• Evaluate and optimize problem-solving approaches through comprehensive understanding and revision of key concepts from previous sessions.			
SESSION WISE DETAILS			
Session: 1	Introduction to competitive programming		No. of hours: 2
Content Summary: Introduction to LeetCode and Codechef coding platforms, Overview of competitive programming, setting up environment, approach to problem solving			
Session: 2	Array I		No. of hours: 2
Reversing the array , finding maximum and minimum elements , Running sum of 1d Array , count elements with maximum frequency , left/right rotate an array by k positions .			
Session: 3	Array II		No. of hours: 2
Content Summary: find element in an array , Remove duplicate elements from an sorted array , find repeating element an array , find equilibrium element in an array .			
Session: 4	Array's Sorting and Time and space complexity Analysis		No. of hours: 2
Content Summary: Bubble sort , selection sort , Insertion Sort and complexity Analysis			

Session: 5	Array III	No. of hours: 2
Content Summary: union and intersection of sorted arrays , maximum subarray sum (Kadane's Algorithm) , maximum product subarray(based on Kandane's) , majority Element (moore's voting algorithm)		
Session: 6	Strings I	No. of hours: 2
Content Summary: check given string is palindrome or not , count number of vowel and consonant , remove character except alphabet .		
Session: 7	String II	No. of hours: 2
Content Summary: Calculate frequency of a character , print maximum occurring character in a string , Remove duplicate character from a string , count number of word in a string		
Session: 8	Recursion I	No. of hours: 2
Content Summary: find factorial , find power of a number , (printing increasing, decreasing and Decreasing Increasing) , count digit , sum of array using recursion		
Session: 9	Recursion II	No. of hours: 2
Content Summary: find pivot index , remove duplicates , fibonacci number , tower of hanoi with recursion tree presentation ,		
Session: 11	Matrix Problems I	No. of hours: 2
Content Summary: Spiral traversal , searching elements in a matrix , Printing elements in sorted order .		
Session: 12	Matrix Problems II	No. of hours: 2
Content Summary: Finding median in row-wise sorted matrix , identifying rows with maximum 1s , rotating matrices by 90 degrees .		
Session: 13	LinkedList Introduction.	No. of hours: 2
Content Summary: add Node on any position , delete Node from given position , search Node in a linked List , Count Node in linked List		
Session: 14	LinkedList I	No. of hours: 2
Content Summary: reverse LinkedList , find mid of the linkedList , Merge Two sorted LinkedList .		
Session: 15	LinkedList II	No. of hours: 2
Content Summary: add two number , rotate list , remove duplicates from sorted list		
Session: 16	Stack Implementation	No. of hours: 2
Content Summary: Stack Implementation using Array , Next Greater Element		
Session: 17	Stack I	No. of hours: 2

Content Summary: Smaller element on left , valid parentheses , Evaluate postfix expression		
Session: 18	Stack II	No. of hours: 2
Content Summary: min stack , asteroid collision , stock span problem		
Session : 19	Queue Introduction.	No. of hours: 2
Content Summary: Queue implementation using array , Implement circular queue , queue using stack		
Session :20	Summary	
Content Summary: Revising the completed topics and company specific problems on given topics.		

Evaluation Criteria

Criteria	Internal (50 marks)
After 2 weeks Coding Test	No. of tests: 6 Marks per test: 5 Total marks: 30
Mid Term Coding Test	20 Marks

External (50 marks)
End Term Paper

B.Tech CSE (all Specializations) Semester IV/BCA,B.SC IV Semester

Department : SOET		School of Engineering technology	
Course Name: Course Name: Competitive Coding-II	Cour se Code	L-T-P	Credits
		2-0-0	NIL
Type of Course:		Audit Course	
Prerequisite(s), if any:Competitive Coding-I, Fundamentals of programming & data structure			
Course Outcomes: <ul style="list-style-type: none">■ Apply advanced string algorithms to solve complex problems.■ Analyze and implement efficient linked list operations and complex problem solutions.■ Evaluate and apply various tree traversal techniques to solve traversal and view-related problems.			
SESSION WISE DETAILS			
Session:1	Advance Array-I		No. of hours: 2
Content summary: Two sum , Best time to buy and sell stocks , Sort 0, 1 and 2(Dutch flag algorithm) ,			
Session: 2	Advance Array-II		No. of hours: 2
Content Summary: container with most water , merge sorted array , trapping rain water			
Session: 3	Binary Search-I		No. of hours: 2
Content Summary: lower bound , upper bound , koko eating bananas , first bad version			
Session: 4	Binary Search-II		
Content Summary: Search in rotated sorted array , Search in rotated sorted array II , aggressive cows			
Session: 5	Binary Tree Introduction		No. of hours: 2
Content Summary: Introduction of Tree, type of tree, implementation of tree .			

Session: 6	Binary Tree Traversal	No. of hours: 2
Content Summary: Tree Traversal, preorder traversal , inorder traversal , postorder traversal , level order traversal (Morris traversal).		
Session: 7	Binary Tree-III.	No. of hours: 2
Content Summary: Height of the tree , same tree , symmetric tree ,		
Session: 8	Binary Tree-IV.	No. of hours: 2
Content Summary: diameter of tree , path sum , print left/right view of Binary tree .		
Session : 9	Binary Search Tree.	No. of hours: 2
Content Summary: Implementation of BST , check valid BST		
Session : 10	Binary Search-II	No. of hours: 2
Content Summary: convert sorted array to BST , Delete node in BST , lowest common ancestor		
Session : 11	HashMap Introduction.	No. of hours: 2
Content Summary: HashMap Implementation (operations put, get, containsKey, KeySet)		
Session: 12	HashMap-II.	No. of hours: 2
Content Summary: Two Sum , highest frequency character , missing number		
Session:13	HashMap-III.	
Content Summary: intersection of two arrays , set matrix zeros , valid anagram		
Session: 14	hashmap/Sliding window-technique Algorithm	No. of hours:2
Content Summary: longest consecutive sequence , longest substring without repeating character , bulls and cows		
Session: 15	hashmap/Sliding window-technique Algorithm	No. of hours: 2
Content Summary: largest subarray with 0 sum , count of zero sum subarray , length of largest subarray with contiguous element		
Session: 16	Priority Queue	No. of hours: 2
Content Summary: Implementation of Priority queue , min and max Heap		
Session: 17	priority Queue-II	No. of hours: 2

Content Summary: Inplace heap sort , kth largest element , kth smallest element		
Session: 18	priority Queue-III	No. of hours: 2
Content Summary: check max heap , top k frequent element , sliding window maximum		
Session: 19	Sum up Binary tree and Binary search Tree	No. of hours: 2
Content Summary: sum of leaves , top view , bottom view ,		
Session: 20	Sum up Hashmap / Sliding window technique.	No. of hours: 2
Content Summary: find all anagram in string , isomorphic string		
Reference Books:		
<ul style="list-style-type: none"> ■ "Introduction to Algorithms" by Cormen, Leiserson, Rivest, and Stein ■ "Cracking the Coding Interview" by Gayle Laakmann McDowell ■ "Elements of Programming Interviews" by Adnan Aziz, Tsung-Hsien Lee, and Amit Prakash 		

Evaluation Criteria

Criteria	Internal (50 marks)
After 2 weeks Coding Test	No. of tests: 6 Marks per test: 5 Total marks: 30
Mid Term Coding Test	20 Marks

External (50 marks)
End Term Paper

B.Tech CSE (all Specializations) Semester V

Department	School of Engineering technology		
Course Name: Competitive Coding-III	Course Code	L-T-P	Credits
		2-0-0	NIL
Type of Course:	Audit Course		
Prerequisite(s), if any: Competitive Coding-II, Fundamentals of Data Structures and Algorithms			
Course Outcome:			
<ul style="list-style-type: none">■ Apply bit manipulation, number theory, and string algorithms to solve computational problems.■ Analyze and implement advanced backtracking and recursion techniques for combinatorial problems.■ Evaluate sliding window techniques and two-pointer algorithms for efficient solutions.■ Solve graph problems using foundational and advanced concepts in competitive programming.			
SESSION WISE DETAILS			
Session 1	Bit Manipulation Introduction.	No. of hours: 2	
Content Summary: Introduction to AND, OR, XOR operations, Count Set/unset Bits , Toggle a given kth bit , check if nth bit is set or unset , Check Power of Two/Four .			
Session: 2	Bit Manipulation-II.	No. of hours: 2	
Content Summary: Counting bits , Single Number 1 , Single number 2 , Subsets using Bits (power set problem) , Find Missing number , Duplicate Numbers .			
Session: 3	Number theory basics.	No. of hours: 2	
Content Summary: Sieve of Eratosthenes , Modular Arithmetic , Modular Exponentiation , Chinese Remainder Theorem			
Session: 4	Mathematical Algorithms.	No. of hours: 2	
Content Summary: Euler’s Totient Function , Permutations and Combinations , Inclusion-Exclusion Principle , Catalan Numbers .			

Session 5	Advance Recursion.	No. of hours: 2
Content Summary: print all subset , permutation of a string , find all unique subset		
Session: 6	Backtracking I	No. of hours: 2
Content Summary: rat in maze , rat in a maze all path , N Queens		
Session: 7	Backtracking-2	No. of hours: 2
Content Summary: combination , combination sum , combination sum-2		
Session: 8	Backtracking-3	No. of hours: 2
Content Summary: generate parentheses , subset-2 , sudoku solver		
Session : 9	Greedy I	No. of hours: 2
Content Summary: assign cookies , array partition , can place flower , lemonade change		
Session: 10	Greedy-II.	No. of hours: 2
Content Summary: Activity selection , minimum platform , coin change		
Session : 11	Greedy-III.	No. of hours: 2
Content Summary: max chunk to make sorted , max chunk to make sorted-2 , 0/1 knapsack .		
Session: 12	Graph Introduction and representation.	No. of hours: 2
Content Summary: Introduction,Representation using adjacency matrix and adjacency list		
Session: 13	Graph-Traversal Algorithm.	No. of hours: 2
Content Summary: Graph Traversal BFS(Breadth first search) and DFS(Depth first search)		
Session: 14	Graph-III	No. of hours: 2
Content Summary : Connected Components , Detecting Cycles in Graphs		
Session: 15	Graph Problems-IV.	No. of hours: 2
Content summary: find if path exist(has path) , print all path from source to destination , Number of Island		
Session: 16	Advanced Graph.	No. of hours: 2
Content summary: Number of Provinces , Flood Fill , Number of closed islands .		
Session: 17	Minimum Spanning Tree algorithms.	No. of hours: 2
Content summary: Prim's Algorithm , Kruskal's algorithm .		
Session: 18	Shortest Path Algorithm.	No. of hours: 2
Content summary: Dijkstra algorithm , Bellman ford algorithm .		

Session: 19	Summarizing the Semester 5.	No. of hours: 2
Content summary: Company specific problems on Graphs, sliding window and recursion.		
Session: 20	Summarizing the Semester 5.	No. of hours: 2
Content summary: Company specific problems on Graphs, sliding window and recursion.		
Reference Books: <ul style="list-style-type: none"> ■ "Competitive Programming" by Steven Halim, Felix Halim ■ "Cracking the Coding Interview" by Gayle Laakmann McDowell ■ "Elements of Programming Interviews" by Adnan Aziz, Tsung-Hsien Lee, and Amit Prakash 		

Evaluation Criteria

Criteria	Internal (50 marks)
After 2 weeks Coding Test	No. of tests: 6 Marks per test: 5 Total marks: 30
Mid Term Coding Test	20 Marks

External (50 marks)
End Term Paper

B.Tech CSE (all Specializations) Semester VI/BCA,B.SC VI Semester

Department	School of Engineering technology		
Course Name: Course Name: Competitive Programming-IV.	Course Code	L-T-P	Credits
		2-0-0	NIL
Type of Course:	Audit Course		
Prerequisite(s), if any: Competitive programming III, Fundamentals of programming & data structure			
Course Outcomes: <ul style="list-style-type: none">■ Apply advanced string algorithms and data structures, such as Trie and Huffman Coding, to solve complex problems.■ Analyze and implement efficient solutions to dynamic programming problems using memoization and tabulation approaches.■ Evaluate and apply tree and segment tree operations to solve traversal, range queries, and interval-based problems.			
SESSION WISE DETAILS			
Session:1	Trie	No. of hours: 2	
Content summary: what is trie DS, use of trie, hashmap vs trie, implementation(representation, insert node, search node)			
Session: 2	Trie-II	No. of hours: 2	
Content Summary: delete node , application of trie , count word in trie , word break ,			
Session: 3	Huffman coding	No. of hours: 2	
Content Summary: huffman coding algorithm , decompression in huffman coding			
Session: 4	Dynamic programming	No. of hours: 2	
Content Summary: introduction of dynamic programing , covering of prerequisites for dynamic programming, discussion of memoization and tabulation using fibonacci number / any problem			

Session: 5	Dynamic programming-II (both approach memoization and tabulation)	No. of hours: 2
Content Summary: staircase , min cost climbing stairs , counting bits , perfect square		
Session: 6	Dynamic programming-III(both approach memoization and tabulation)	No. of hours:2
Content Summary: house robber , house robber 2 , goldmine , path with maximum gold		
Session: 7	Dynamic programming-IV	No. of hours: 2
Content Summary: unique path-1 , unique path-2 , minimum path sum , target sum subset (DP)		
Session: 8	Dynamic programming-V.	No. of hours: 2
Content Summary: coin change , coin change-2 , jump game , jump game 2		
Session : 9	Dynamic programming-VI.	No. of hours: 2
Content Summary: longest common subsequences(recursive, memoization tabulation) , longest palindrome substring ,		
Session : 10	Dynamic programming-VII.	No. of hours: 2
Content Summary: longest palindrome subsequences , palindromic substring		
Session : 11	Dynamic programming-VIII.	No. of hours: 2
Content Summary: wildcard matching problem , egg dropping problem		
Session: 12	Revised session on DP	No. of hours: 2
Content Summary: regular expression matching problem		
Session:13	Segment tree	
Content Summary: what is segment tree , what kind of problem we solve using segment tree , max in interval ,		
Session: 14	Segment tree-II/ range query	No. of hours:2
Content Summary: sum of interval , sum of ranges(lazy propagation)		
Session: 15	Company specific problem	No. of hours: 2
Content Summary: Problems on topics covered.		
Session: 16	Company specific problem	No. of hours: 2

Content Summary: Problems on topics covered.		
Session: 17	Company specific problem	No. of hours: 2
Content Summary: Problems on topics covered.		
Session: 18	Constructive Algorithmic Problems	No. of hours: 2
Content Summary: company specific problems		
Session: 19	Scenario based problems.	No. of hours: 2
Content Summary: problems on topics covered.		
Session: 20	Scenario based problems.	No. of hours: 2
Content Summary: Problems on topics covered.		
Reference Books:		
<ul style="list-style-type: none"> ■ "Introduction to Algorithms" by Cormen, Leiserson, Rivest, and Stein ■ "Cracking the Coding Interview" by Gayle Laakmann McDowell ■ "Elements of Programming Interviews" by Adnan Aziz, Tsung-Hsien Lee, and Amit Prakash 		

Evaluation Criteria

Criteria	Internal (50 marks)
After 2 weeks Coding Test	No. of tests: 6 Marks per test: 5 Total marks: 30
Mid Term Coding Test	20 Marks

External (50 marks)
End Term Paper