

Course: CS2700 (Programming and Data Structures Theory)
Instructor: Dr. Aritra Hazra (Assistant Professor, Dept. of CSE, IIT Madras)

August	
01	Course Details and Class Announcements An Introduction to Problem Solving
02	Problem Solving Approaches using – (a) Algorithm Design and (b) Data Structuring
03	Arguments on Correctness of Programs (Implementation): Pre/Post Conditions, Loop Invariants
04	Complexity (Efficiency) of Computation / Algorithms: Time and Space
08	Asymptotic Analysis of Programs using Asymptotic Growth Functions
09	Problem Decomposition using Recursion: (Examples: Factorial, Fibonacci, Sorting)
10	Problem Decomposition using Recursion: (Example: Tower of Hanoi)
11 (T)	<i>Tutorial-01: Analyzing Program Complexity and Correctness</i>
16	ADT Array: Searching Element in Array and its Complexity Analysis (a) Linear Search (b) Binary Search
17	ADT Array: Comparison based Sorting and its Complexity Analysis (a) Selection Sort (b) Bubble Sort
18	ADT Array: Comparison based Sorting and its Complexity Analysis (c) Insertion Sort (d) Merge Sort (Merging Technique)
22	ADT Array: Comparison based Sorting and its Complexity Analysis (e) Quick Sort A note on Lower Bound Complexity of Comparison based Sorting
23	ADT Array: Linear-Time Sorting Procedures (a) Counting Sort (b) Radix Sort
24	ADT Array: Linear-Time Sorting Procedures (c) Bucket Sort A note on Stable and In-place Sorting
25 (T)	<i>Tutorial-02: Array ADT</i>
29	ADT Linked List: Basic Operations (a) Dynamic Structure based Implementation of Single Linked List (b) List Manipulation: Insertion, Deletion, Find-element
30	ADT Linked List: Basic Operations (c) Reversing List: Recursive and Non-recursive way (d) Cursor (Array-based) Implementation of Linked List
31	ADT Linked List: Variations (a) Double Linked List (b) Circular Linked List (c) Sparse Matric Implementation using Linked List
September	
01 (T)	<i>Tutorial-03: Linked List ADT</i>

06	ADT Stack: Basic Operations (Empty, Full, Push and Pop) (a) Array based Implementation (b) Linked List based Implementation
07	ADT Queue: Basic Operations (Empty, Full, Enqueue and Dequeue) (a) Array based Implementation (b) Linked List based Implementation
08 (Q)	Quiz-1: 20 Marks, 45 Minutes (Time: 08:00AM – 08:50AM, Rooms: TBD); Syllabus: 01/08/2016 – 01/09/2016
13	ADT Stack and ADT Queue: Various Usage (a) Equality of Two Stacks / Queues (b) Parenthesis Matching (c) Expression Evaluation (d) Transformation of Infix to Postfix Expression
14	ADT Binary Trees: (a) Mathematical Properties of Tree / Binary Tree (b) Representation using Dynamic Data Structures (c) Tree traversal: Pre-order, In-order, Post-order
15 (T)	<i>Tutorial-04: Stack ADT</i>
16	ADT Binary Trees: Various Operations (a) Recursive Traversals: Determining Height / Depth, Number of Nodes (b) Binary Tree Construction from Traversal Data: Inorder + Preorder and Inorder + Postorder
19	ADT Binary Trees: Applications (a) Huffman Coding (b) Expression trees and Evaluation
20	ADT Binary Search Tree (BST): (a) Basic Operations: Find a Key and Min-Max, Insert and Delete Element (b) BST Traversal to Sort Data
21	Tree Height Balancing (AVL Tree): (a) Representation / Implementation (b) Rotation Techniques for Balancing Tree Height (Single and Double Rotation)
22 (T)	<i>Tutorial-05: Queue ADT</i>
26	ADT B-Tree: Few Variations (2-3 Tree, B+ Tree)
27	Hashing: (a) Hash Function: Properties and Examples (b) Collision and Prevention: Separate Chaining, Open Addressing (Linear Probing, Quadratic Probing, Double Hashing), Rehashing
28	ADT Heap (Priority Queue): Basic Operations and Complexity (a) Array based Implementation: Build-Heap (b) Find Min/Max, Insert and Delete Operations
29 (T)	<i>Tutorial-06: Binary Tree ADT</i>
October	
03	Heap-Sort: Algorithm and Analysis
04	ADT Disjoint Set: Tree based Implementation, Find and Union Operation
05	ADT Disjoint Set: Path Compression and Application (Equivalence Class)
06 (T)	<i>Tutorial-07: Binary Search Tree ADT</i>
10 (T)	<i>Tutorial-08: Hashing</i>
13 (T)	<i>Tutorial-09: Heap (Priority Queue) ADT</i>

17	Graph Representation: Matrix and Adjacency List Graph Traversals: Depth-First and Breadth-First (using Stacks and Queues)
18	Graph Traversal Applications: (a) Topological Sort, (b) Connected Components, (c) Strongly Connected Components
19	Shortest Path in Graph: (a) Single Source All Destinations, (a) All Pair Shortest Path
20 (T)	<i>Tutorial-10: Disjoint Set ADT</i>
24	Minimum Cost Spanning Tree: Prim's and Kruskals Approach
25 (Q)	Quiz-2: 20 Marks, 45 Minutes (Time: 08:00AM – 08:50AM, Rooms: TBD); Syllabus: 06/09/2010 – 13/10/2016
26	Advanced Data Structures and Analysis: (a) Red-Black Tree (b) Amortized Analysis
27 (T)	<i>Tutorial-11: Graphs and Traversals</i>
31	Advanced Data Structures and Analysis: (c) Binomial Heap (d) Skew Heap
November	
01	Advanced Data Structures and Analysis: (e) Fibonacci Heap (f) Splay Tree
02	Algorithm Design Techniques: (a) Greedy Approach (b) Divide and Conquer Approach
03 (T)	<i>Tutorial-12: Graph Algorithms</i>
07	Algorithm Design Techniques: (c) Dynamic Programming (d) Branch and Bound / Backtracking
08	Computational Intractability: P and NP, NP-Completeness
09	Algorithm Design Techniques: (e) Approximate Algorithm (f) Randomized Algorithm
10 (T)	<i>Tutorial-13: Advanced Data Structures and Algorithm Design Techniques</i>
21 (E)	<i>End-Semester: 50 Marks, 2 Hours (Time: 09:00AM – 11:00AM, Rooms: TBD); Syllabus: Full</i>

Course: CS2710 (Programming and Data Structures Laboratory)
Instructor: Dr. Aritra Hazra (Assistant Professor, Dept. of CSE, IIT Madras)

August		
01	Course Details and Introduction to Programming Environment	
08	Assignment-01: Programming Basics	(5 Marks)
18	Assignment-02: Problem Solving using Recursions	(5 Marks)
22	Assignment-03: Manipulating Array – (a) Searching and (b) Sorting	(5+5 = 10 Marks)
29	Assignment-04: Linked list	(5 Marks)
September		
16	Assignment-05: Stack and Queue	(5 Marks)
19 (L)	LabTest-1: 20 Marks, 2 Hours 30 Minutes (Time: 02:00PM – 04:40PM, Room: DCF)	
26	Assignment-06: Binary Tree	(5 Marks)
October		
03	Assignment-07: Binary Search Tree	(5 Marks)
10	Assignment-08: Heap and Priority Queue	(5 Marks)
17	Assignment-09: Set ADT and Hashing	(5 Marks)
24	Assignment-10: Graph	(5 Marks)
31 (L)	LabTest-2: 20 Marks, 2 Hours 30 Minutes (Time: 02:00PM – 04:40PM, Room: DCF)	
November		
07	Assignment-11: Advanced Data Structures and Algorithms	(5 Marks)