## 22DS2E2: DESIGN & ANALYSIS OF ALGORITHMS

| Course Name | Design & Analysis of Algorithms | | | L | T | P | C | CIA | SEE | TM |
|---|---|---|---|---|---|---|---|---|---|---|
| Course Code | 22DS2E2 | | | 4 | 0 | 0 | 4 | 30 | 70 | 100 |
| Year of Introduction: 2005 | Year of Offering: 2021 | Year of Revision: 2022 | | Percentage of Revision: NIL | | | | | | |
| **L**-Lecture, T-Tutorial, **P**-Practical, **C**-Credits, **CIA**-Internal Marks, **SEE**-External Marks, **TM**-Total Marks | | | | | | | | | | |

**Course Description and Purpose:** Design & Analysis of Algorithms (22DS2E2) is a course that illustrates *Algorithms*, *Analysis*, *Elementary Data Structures, Divide - and -Conquer Technique* and *The Greedy Method, Dynamic Programming* and *Basic Traversal and Search Techniques, Backtracking* and *Branch and Bound Techniques, NP Hard* and *NP Complete Problem.*

**Course Objectives:**
This course will help enable the students t o understand and learn various *Algorithms*, *Analysis*, *Elementary Data Structures, Divide -and -Conquer Technique* and *The Greedy Method, Dynamic Programming* and *Basic Traversal and Search Techniques, Backtracking* and *Branch and Bound Techniques, NP Hard* and *NP Complete Problem.*

**Course Objectives:**
- To understand *Algorithms*, *Analysis*, *Elementary Data Structures*.
- To gain familiarity in *Divide -and -Conquer Technique* and *The Greedy Method*.
- To apply the concepts of *Dynamic Programming* and *Basic Traversal and Search Techniques*.
- To understand the concepts of *Backtracking* and *Branch and Bound Techniques*.
- To acquire knowledge in *NP Hard* and *NP Complete Problem*.

**Course Outcomes:**
Upon successful completion of the course, the student will be able to:
CO1: Understand *Algorithms*, *Analysis*, *Elementary Data Structures*.
CO2: Gains familiarity in *Divide-and-Conquer Technique* and *The Greedy Method*.
CO3: Apply the concepts of *Dynamic Programming* and *Basic Traversal and Search Techniques*.
CO4: Understand the concepts of *Backtracking* and Branch and Bound techniques.
CO5: Acquire knowledge in *NP Hard* and *NP Complete Problem*.

### UNIT I (12 Hours)

**Introduction:** What IS Algorithm - Algorithm Specification - Pseudocode Conventions - Recursive Algorithms - Performance Analysis: Space Complexity Time Complexity - Asymptotic Notation - Performance Measurement - Randomized Algorithms (Basics of Probability Theory, Randomized Algorithms Identifying the Repeated Element, Primality Testing: Advantages and Disadvantages).
**Elementary Data Structures:** Binary Trees - Dictionaries (Binary Search Trees, Priority Queues, Heaps, Heap sort) - Sets and Disjoint Set Union (Introduction, Union and Find Operations).

### UNIT II (12 Hours)

**Divide - and - Conquer:** General Method - Defective Chess Board - Binary Search - Finding Maximum and Minimum - Merge Sort - Quick Sort - Selection Problem - Strassen's Matrix Multiplication - Convex Hull: (Some Geometric Primitives, The Quick Hull Algorithm, Graham's Scan ,An O(nlogn) Divide and Conquer Algorithm).

**The Greedy Method:** The General Method - Container Loading - Knapsack Problem - Tree Vertex Splitting - Job Sequencing with Deadlines - Minimum Cost Spanning Trees: Prim's Algorithm - Kruskal's Algorithm - Optimal Storage on Tapes - Optimal Merge Patterns - Single Source Shortest Paths.

**UNIT III (12 Hours)**

**Dynamic Programming:** The General Method - Multi Stage Graphs - All Pairs Shortest Paths - Single Source Shortest Paths - Optimal Binary Search Trees - String Editing 0/1 Knapsack - Reliability Design - The Traveling Sales Person Problem - Flow Shop Scheduling.

**Basic Traversal and Search Techniques:** Techniques for Binary Trees - Techniques for Graphs: Breadth First Search and Traversal Depth First Search - Connected Components and Spanning Trees -Bi Connected Components and DFS.

**UNIT IV (12 Hours)**

**Backtracking:** The General Method - The 8 Queens Problem - Sum of Subsets - Graph Coloring - Hamiltonian Cycles - Knapsack Problem.

**Branch and Bound:** The Method: (Least Cost Search, The 15 Puzzle Control Abstractions for LC Search, Bounding, FIFO Branch and Bound - LC Branch and Bound ) - 0/1 Knapsack Problem (LC Branch and Bound Solution - FIFO Branch and Bound Solution) - Traveling Sales Person.

**UNIT V (12 Hours)**

**NP Hard and NP Complete Problems:** Basic Concepts: Non Deterministic Algorithms - The Classes NP Hard and NP Complex - Cook's Theorem - NP Hard Graph Problems (Clique Decision Problem, Node Cover Decision Problem, Chromatic Number Decision Problem, Directed Hamiltonian Cycle, Traveling Sales Person Decision Problem, AND/OR Graph Decision Problem) - NP Hard Scheduling Problems ( Scheduling Identical Processors, Flow Shop Scheduling, Job Scheduling) - NP Hard Code Generation Problems (Code Generation With Common Sub Expressions, Implementing Parallel Assignment Instructions) - Some Simplified NP-Hard Problems.

**Reference Text Books:**

1.Sartaj Sahni and Sanguthevar Rajasekaran Ellis Horowitz, Fundamentals of Computer Algorithms, Fourth Edition, Universities Press, 2018

2. Sartaj Sahni, Fundamentals of Computer Algorithms, Second Edition, Universities Press, 2008

3. Coremen TH Leiserson CE, Rivest R L and Stein, Clifford, Introduction to Algorithms, PHI , Third Edition, 2010, 35th Chapter

4. Anany Levitin, Introduction to the Design & Analysis of Algorithms, Second Edition, Pearson Education (2007)

5. I.Chandra Mohan, Design and Analysis of Algorithms, PHI

6. Prabhakar Gupta, Vineet Agrawal, Design and Analysis of Algorithms, PHI

7. Parag Himanshu, Dave, Design and Analysis of Algorithms, Pearson Education (2008)

**22DS2E2**

**P.B.SIDDHARTHA COLLEGE OF ARTS & SCIENCE (AUTONOMOUS), VIJAYAWADA-520010**

(An Autonomous College in the Jurisdiction of Krishna University, A.P., India.)

**M.Sc.(COMPUTATIONAL DATA SCIENCE) DEGREE EXAMINATIONS**

**SECOND SEMESTER**

**DESIGN & ANALYSIS OF ALGORITHMS**

**SYLLABUS W.E.F 2022-2023**

**Time 3 Hours** **Max.Marks: 70**

**SECTION-A**

**Answer ALL questions** **(5×4 = 20 Marks)**

1. (a) Define *Algorithm*. Explain the algorithm specification briefly.(CO1, L1)

(or)

(b) What are the operations in a *priority queue*? (CO1, L1)

2. (a) Explain the Divide and Conquer Algorithms to solve *Convex Hull Problem.* (CO2, L1)

(or)

(b) What is *tree vertex splitting?* (CO2, L1)

3. (a) What is *String Editing ?* (CO3, L1)

(or)

(b) Differentiate *DFS and BFS.* (CO3, L1)

4. (a) What is *Graph colouring?* (CO4, L1)

(or)

(b) What is *Branch and Bound* technique*?*(CO4, L1)

5. (a) Compare *NP hard and NP complete classes.* (CO5, L1)

(or)

(b) Explain *flow shop scheduling* in *NP Hard Scheduling problems.* (CO5, L1)

**SECTION - B**

**Answer all questions. All question carry equal marks.** **5 × 10 = 50 Marks**

6. (a) Define Algorithm. Discuss *Performance Analysis of Algorithms* briefly. ( CO1, L2) 10 Marks

(or)

(b) Explain Disjoint Sets, Disjoint Set Union & Find Operations with Algorithms. (CO1, L2) 10 Marks

7. (a) Discuss the method for *Divide and Conquer* approach and write algorithm for Quick Sort with an example. (CO2, L6) 10 Marks

(or)

(b) Discuss the general method for *Greedy Method*. Apply it on *Single Source Shortest Path* by writing an algorithm with suitable example. (CO2,L6) 10 Marks

8. (a) Examine algorithm and procedure of finding *Optimal Binary Search Tree* using Dynamic Programming with example. (CO3,L4) 10 Marks

(or)

(b) Examine *Traversal Techniques for Graphs* with an example. (CO3,L4) 10 Marks

9. (a) Explain *Control Abstraction for LC Search*. Solve *0/1-Knapsack Problem* using *Branch and Bound Technique*. (CO4,L5) 10 Marks

(or)

(b) Explain the *Sum of Subsets Problem* using *Back Tracking Technique.* (CO4,L5) 10 Marks

10.(a) Make use of different formulae prove *COOKs Theorem.*. (CO5,L3) 10 Marks

(or)

(b) Choose *NP-Hard Graph problems* and explain. (CO5,L3) 10 Marks