CMR ENGINEERING COLLEGE
UGC AUTONOMOUS
(Approved by AICTE - New Delhi. Affiliated to JNTUH and Accredited by NAAC & NBA)
Kandlakoya (V), Medchal (M), Medchal - Malkajgiri (D)-501401

# DEPARTMENT OF CSE – DATA SCIENCE

## LIFT (Laboratory Improvement for Future Trends)

# DESIGN AND ANALYSIS OF ALGORITHMS
# II B.TECH II SEM CSE – DS
# 2024-2025

# CONTENTS

**S.No CONTENT**

**1. OBJECTIVES AND RELEVANCE**

**2. SCOPE**

**3. PREREQUISITES**

**4. SYLLABUS AS PER AUTONOMOUS**

**5. LEAD EXPERIMENTS**

**6. LAB SCHEDULE**

**7. SUGGESTED BOOKS**

**8. WEBSITES (USEFUL LINKS)**

**9. EXPERT DETAILS**

**10. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS**

**11. INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**

**12. SHADOW ENGINEERING**

**13. CALIBIRATION , TESTING AND INSPECTION**

**14. MAINTAINANCE AND TROUBLESHOOTING**

**15. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL**

**16. VIRTUAL LABS REFERENCE LINKS**

## 1. Objectives and Relevance

- To solve problems using Divide and Conquer strategy.
- To solve problems using Backtracking strategy.
- To solve problems using Greedy and Dynamic programming techniques.

## 2. Scope

The scope of this lab is to make students to acquire knowledge by solving problems using algorithm design techniques such as Divide and Conquer, Greedy, Dynamic programming, and Backtracking.

## 3. Prerequisites

Students should have foundation in Data Structures. The student should have programming experience ( in C,C++ and JAVA).

## 4. Syllabus:

**List of Experiments:**

1. Write a java program to implement Quick sort algorithm for sorting a list of integers in ascending order
2. Write a java program to implement Merge sort algorithm for sorting a list of integers in ascending order.
3. i) Write a java program to implement the DFS algorithm for a graph.
4. ii) Write a java program to implement the BFS algorithm for a graph.
5. Write a java program to implement Backtracking algorithm for the N-queens problem.
6. Write a java program to implement the backtracking algorithm for the sum of subsets problem.
7. Write a java program to implement the backtracking algorithm for the Hamiltonian Circuits problem.
8. Write a java program to implement greedy algorithm for job sequencing with deadlines.
9. Write a java program to implement Dijkstra's algorithm for the Single source shortest path problem.
10. Write a java program that implements Prim's algorithm to generate minimum cost spanning tree.
11. Write a java program that implements Kruskal's algorithm to generate minimum cost spanning tree
12. Write a java program to implement Floyd's algorithm for the all pairs shortest path problem.
13. Write a java program to implement Dynamic Programming algorithm for the 0/1 Knapsack problem.
14. Write a java program to implement Dynamic Programming algorithm for the Optimal Binary Search Tree Problem.

**Week 1**: Write a java program to implement **Quick sort** algorithm for sorting a list of integers in ascending order**(Unit -1)**

**Objective**: The main objective of this program states that list of integers are sorted in ascending order by using Quick sort algorithm.

**Prerequisites**: Requires knowledge in Data Structures, Java

**Description**: Quick sort divides the array to the values of elements .It rearranges elements if a given array A[0…..n-1] to achieve its partition, where the elements before position 's' are smaller than or equal to A[s] and all the elements after position s are greater than or equal to A[s].

**Applications:** Amazon, Flipkart can show items sorted on the basis of price, ratings, availability .

**Week 2**: Write a java program to implement **Merge sort** algorithm for sorting a list of integers in ascending order**(Unit -1)**

**Objective:** The main objective of this program states that list of integers are sorted in ascending order by using Merge sort algorithm

**Prerequisites:** Requires knowledge in Data Structures, Java

**Description**:. Merge Sort breaks a problem into sub problems that are similar to the original problem, recursively solves the sub problems, and finally combines the solutions to the sub problems to solve the original problem.

**Applications:** Real time example is maintaining Students data in Colleges.

**Week 3**: Write a java program to implement the DFS(Depth First Search) algorithm for a graph. (UNIT-II)

**Objective:** The main objective of the programs is student can able to learn DFS.

**Prerequisites:** Requires knowledge in Data Structures, Java.

**Description**: DFS is the method for exploring a tree or a graph . Depth-first search starts visiting vertices of a graph at an arbitrary vertex by marking it as having been visited. On each iteration, the algorithm proceeds to an unvisited vertex that is adjacent to the one it is currently in. This process continues until a vertex with no adjacent unvisited vertices is encountered. At a dead end, the algorithm backs up one edge to the vertex it came from and tries to continue visiting unvisited vertices from there. The algorithm eventually halts after backing up to the starting vertex, with the latter being a dead end.

**Applications:** Finding Minimum cost Spanning Trees of weighted graphs, Path finding.

**Week 4:** Write a java program to implement the BFS (Breadth First Search) algorithm for a graph **(UNIT-II)**

**Objective:** The main objective of the programs is student can able to learn BFS

**Description**: BFS explores graph moving across to all the neighbors of last visited vertex traversals i.e., it proceeds in a concentric manner by visiting all the vertices that are adjacent to a starting vertex, then all unvisited vertices two edges apart from it and so on, until all the vertices in the same connected component as the starting vertex are visited.

**Prerequisites**: Requires knowledge in data structures, java.

**Applications:** Finding Shortest Path and Minimum cost Spanning Tree.

**Week 5**: Write a java programs to implement Backtracking algorithm for the N-Queens problem. **(UNIT-II)**

**Objective:** Student can be able to learn N-Queen Problem.

**Description:** The N-Queen Problem is the problem of placing N chess queens on an N*N chessboard so that no two queens attack each other.

**Prerequisites:** Requires knowledge in Data Structures, Java.

**Applications**: N-Queens problem is used in Game playing such as chessboard.

**Week 6**: Write a java program to implement the Backtracking algorithm for the sum of subsets problem. **(UNIT-II)**

**Objective**: Student can be able to understand Sum of Subsets problem.

**Prerequisite**s: Requires knowledge in Data Structures, Java.

**Description**: Subset-Sum Problem is to find a subset of a given set S= {s1, s2… $s_n$} of n positive integers whose sum is equal to a given positive integer d. It is assumed that the set's' with n elements are sorted in increasing order. The state-space tree can then be constructed as a binary tree and applying backtracking algorithm, the solutions could be obtained.

**Applications:** sum of subsets problem is used in game playing

**Week 7:** Write a java program to implement the Backtracking algorithm for the Hamiltonian Circuits problem**.(UNIT-II)**

**Objective:** Student can be able to solve Hamiltonian Circuits problem

**Prerequisites**: Requires knowledge in Data Structures, java.

**Description**: Hamiltonian circuit of a graph is a tour that visits every vertex once, and ends at its starting vertex. Finding out if a graph has a Hamiltonian circuit is an NP-complete problem.

**Applications:** Hamiltonian Circuits problem is used in computer graphics.

**Week** 8.Write a java program to implement Greedy algorithm for job sequencing with deadlines. (UNIT-IV)

**Objective**: Student can be able to learn job sequencing with deadlines.

**Prerequisites**: Requires knowledge in Data Structures, java.

**Description**: The objective of job sequencing with deadlines is to find a sequence of jobs which is completed within the deadlines and gives maximum profit.

**Applications:** It is used in finding the maximum Profit when deadlines are given.

**Week** 9.Write a java program to implement Dijkstra's algorithm for the Single source shortest path problem.(UNIT-IV)

**Objective**: The objective of Dijkstra‟s algorithm is the best known algorithm for the single source shortest paths problem
**Prerequisites**: Require knowledge in Data structures, Java.

**Description**: This algorithm is applicable to graphs with nonnegative weights only and finds the shortest paths to a graph‟s vertices in order of their distance from a given source. It finds the shortest path from the source to a vertex nearest to it, then to a second nearest, and so on. It is applicable to both undirected and directed graphs.

**Applications:** Dijkstra's algorithm for the Single source shortest path problem .

**Week** 10: .Write a java program that implements Prim's algorithm to generate minimum cost spanning tree.(UNIT-IV)

**Objective**: Prim‟s algorithm finds the minimum spanning tree for a weighted connected graph

**Description**: Prim‟s algorithm finds the minimum spanning tree for a weighted connected graph G=(V,E) to get an acyclic sub graph with |V|-1 edges for which the sum of edge weights is the smallest.

**Prerequisites**: Requires knowledge in Data structures, java.

**Applications:** Prim 's algorithm is used for finding minimum cost spanning tree.

**Week** 11.Write a java program that implements Kruskal's algorithm to generate minimum cost spanning tree. (UNIT-IV)

**Objective**: Student can be able to understand Minimum cost spanning tree. tree

**Prerequisites**: Requires knowledge in Datastructures, java

**Description**: Kruskal's algorithm finds the minimum spanning tree for a weighted connected graph G=(V,E) to get an acyclic subgraph with |V|-1 edges for which the sum of edge weights is the smallest.. The algorithm begins by sorting the graph‟s edges in non decreasing order of their weights. Then starting with the empty subgraph, it scans the sorted list adding the next edge on the list to the current subgraph if such an inclusion does not create a cycle and simply skipping the edge otherwise.

**Applications:** Kruskal's algorithm is used for finding minimum cost spanning tree.

**Week** 12.Write a java program to implement Floyd's algorithm for the all pairs shortest path problem. (UNIT-III)

**Objective**: Student can be able to understand All Pairs shortest Path.

**Prerequisites**: Requires knowledge in Data structures, java

**Description**: Floyd‟s algorithm is applicable to both directed and undirected graphs provided that they do not contain a cycle. It is convenient to record the lengths of shortest path in an n- by- n matrix D called the distance matrix. The element dij in the ith row and jth column of matrix indicates the shortest path from the ith vertex to jth vertex (1<=i, j<=n). The element in the ith row and jth column of the current matrix.

**Applications:** It is used for finding shortest paths in directed and undirected graphs.

**Week** 13.Write a java program to implement Dynamic Programming algorithm for the 0/1 Knapsack problem. (UNIT-III)

**Objective**: Student can be able to understand 0/1 Knapsack problem

**Prerequisites**: require knowledge in data structures, java.

**Description**: In this dynamic programming problem we have n items each with an associated weight and value (benefit or profit). The objective is to fill the knapsack with items such that we have a maximum profit without crossing the weight limit of the knapsack. Since this is a 0-1 knapsack problem hence we can either take an entire item or reject it completely.

**Applications:** The 0/1 knapsack problem in wide variety of resource constrained selection tasks

**Week** 14. Write a java program to implement Dynamic Programming algorithm for the Optimal Binary Search Tree Problem. (UNIT-III)

**Prerequisites**: Requires knowledge in Data Structures, Java.

**Objective**: The objective of optimal search tree is for which nodes are arranged on levels such that the tree cost is minimum.

**Applications:** It is used to solve Discrete optimization Problems such as packaging, inventory management.

**5. LEAD**

1. Java Program to implement Travelling Sales Person problem using Dynamic Programming.
2. Design and implement in Java to find all Hamiltonian cycles in a connected undirected graph using backtracking principle.

## 6. LAB SCHEDULE:

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 |
|---|---|---|---|---|---|---|
| B1(6765 to 6799) | Demo & Introduction | Exp.1 | Exp 2 | Exp 3,4 | Exp5 | Exp 6 |
| B2(67A0 to LE10) | Demo & Introduction | Exp.1 | Exp 2 | Exp 3,4 | Exp5 | Exp 6 |

| Batches | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 |
|---|---|---|---|---|---|---|
| B1(6765 to 6799) | Exp.7 | Exp.8,9 | Exp.10,11 | Exp.12,13 | Exp 14 | Lead 1,2 |
| B2(67A0 to LE10) | Exp.7 | Exp.8,9 | Exp.10,11 | Exp.12,13 | Exp 14 | Lead 1,2 |

**VIVA SCHEDULE:** The viva schedule should be planned prior starting to the lab experiment.

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 | week-11 | week-12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| B1 | viva | | viva | | viva | | viva | | viva | | viva | |
| B2 | | viva | | viva | | viva | | viva | | viva | | viva |

| Batches | week-1 | week-2 | week-3 | week-4 | week-5 | week-6 | week-7 | week-8 | week-9 | week-10 |
|---|---|---|---|---|---|---|---|---|---|---|
| SG1 | viva | | | | | | | | | |
| SG2 | | | viva | | | | | | | |
| SG3 | | | | | viva | | | | | |
| SG4 | | | | | | | Viva | | | |
| SG5 | | | | | | | | | viva | |

*SG: Selected Group with a maximum of 6 or 12 student

# SCHEME OF EVALUATION OF LABS

| **LAB INTERNAL:** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Day to Day Evalution-10** | | | | | **Internal Exam-30** | | | |
| **Uniform** | **Observation & Record** | **Performance Of the Experiment** | **Result** | **Viva** | **Write up** | **Execution & Results** | **Viva** | **Project** |
| **Marks-2** | **Marks-2** | **Marks-2** | **Marks-2** | **Marks-2** | **Marks-5** | **Marks-5** | **Marks-10** | **Marks - 10** |
| **Total Marks-40** | | | | | | | | |

| **LAB EXTERNAL: MARKS : 60** | | | | |
|---|---|---|---|---|
| **Write Up Of Program** | **Experiment / Program** | **Evaluation Of Results** | **Another Program Presentation** | **Viva - Voce** |
| **Marks-10** | **Marks-15** | **Marks-15** | **Marks-10** | **Marks-10** |

## 7. SUGGESTED BOOKS:

a. Data structures, Algorithms and Applications in java, 2nd Edition, S. Sahani, Universities Press.
b. Data structures and Algorithms in java, 3rd edition, A. Drozdek, Cengage Learning.
c. Data structures with Java, J. R. Hubbard, 2nd edition, Schaum's Outlines, TMH.
d. Data structures and algorithms in Java, 2nd Edition, R. Lafore, Pearson Education.
Data Structures using Java, D. S. Malik and P.S. Nair, Cengage Learning

## 8. WEBSITES

a. www.cse.hcmut.edu.vn/~hungnq/courses/nap/alp.**pdf**
b. https://sites.google.com/site/cdlabjntu09/home
c. http://www.nptel.com/computerscience/designandanalysisofal gorithms
d. http://nptel.iitm.ac.in/courses.php?disciplineId=106

## 9. EXPERT DETAILS

The expert details which have been mentioned below are only a few of the eminent ones known.

### INTERNATIONAL

1. S. Sahani, Universities Press.
2. A. Drozdek, Cengage Learning
3. D. S. Malik and P.S. Nair, Cengage Learning
4. R. Lafore, Pearson Education
5 . D. S. Malik and P.S. Nair, Cengage Learning

### NATIONAL

1. G.Vijaya Kumari , Associate professor, School of Information Technology , JNTU Hyderabad
2. Dr. M. Srinivasa Rao , Professor, Dean(MSIT), JNTU Hyderabad

### REGIONAL

1. Dr.SaiBaba, Prof, IT Dept. KL University, Vijayawada.
2. Mr.S.Madhu Assoc.Prof , CSE Dept.GNEC, Hydearabad

**10. MAPPING LAB WITH PROJECT/CONSULTANCY/R&D PROPOSALS**

Proposal 1: Project Design & Execution

Proposal 2: R& D Level Project Design & Execution

Proposal 3: Consultancy Task / Project Design & Development

**11. INDUSTRIAL VISITS ( IIP– INNOVATIVE INDUSTRIAL LEARNING PROGRAM ):**

| S.No | Type of industry | Nature of industry | Date of visit | No. of students participated | Year/branch | remarks |
|---|---|---|---|---|---|---|
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |

**12. OBJECTIVES OF SHADOW ENGINEERING:**

a. The program which uplifts the knowledge of the students related to laboratories.
b. To improve the industry-college interactions.
c. To create industry like environment for all the students in order to make future Assignment.

## 13. CALIBRATION & TESTING (METHOD & TOOLS TO BE USED)

**Calibration:** There is no need to be calibrated any equipment for this laboratory.

**Installation:** Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment.

## 14. MAINTAINANCE AND TROUBLESHOOTING:

**Maintenance:**

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format

**(2)** Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically

| S.No. | Name of the Equipment | Date of Maintenance | Type of Activity | Remarks |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

**TROUBLE SHOOTING SCHEDULES:**

A proposal is to be made from each lab branch wise. The proposal should carry following details related to specific equipment in lab.

S.No., Equipment Name , Type of Problem (Abnormal Sound, Corrupt Software, Anti Virus Problem, Missing of Display, CRT not working,), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

| S.No. | Date of recording activity | Equipment Name | Type of Trouble | Remedial Activity | Remarks |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |

**15. NABL (National Accreditation Board for Testing and Calibration Laboratories)**
(Proposals, If any)

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology in each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/incharge
4. Ensure Effective environmental conditions(temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipments from dust, dirt etc.
5. Ensure Calibration of instruments/equipment(Only NABL accredited authorized laboratories provide calibration.
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team.
10. Auditors should submit the detailed report of each lab duly signed to the Principal.
11. Each lab should maintain all the bills/invoices of each instrument or equipment in a separate file.
12. All the stock registers either consumable or non consumable should be updated whenever any purchases of consumables or equipment takes place.
13. All the safety precautions are properly displayed in front of each lab.
14. All the Lead experiments should be maintained separately in a record /record in a separate folder.
15. Based on Pre Assessment report submitted by auditor, corrective actions should be carried out by each lab in charge and that must be forwarded to concerned HOD and Principal.

## 16. VIRTUAL LABS REFERENCE LINKS

http://cse01-iiith.vlabs.ac.in/
http://people.cs.pitt.edu/~kirk/cs1501/animations/Sort2.html