KLE Technological University | Creating Value, Leveraging Knowledge

Department of Computer Science and Engineering(AI)

Laboratory Plan

FMTH0303-3.1

Semester: III *Year: 2025-26*

Laboratory Title: Algorithms Lab	*Lab. Code*: 25ECAP206
Total Hours: 45	*Duration of Exam*: 3 hrs
Total Exam Marks: 100	*Total ISA. Marks*: 80
Lab. Plan Author: Mr. Pradeep	*Date*: 09 Sept 2025
Checked By: Dr. Vijayalakshmi M	*Date:* 13 Sept 2025

Preamble:

Algorithms help to break down a complex problem into several simple steps so that the computation is essentially understood. They are the logical map of the program. To be distinct in this competitive world, one not only needs to be well versed with algorithms but also should be able to design an efficient one with respect to time and space complexity analysis. The objective of this laboratory is to develop the capability of writing algorithms for the given task, implement it using suitable data structures and work towards improving its efficiency. An algorithms lab in engineering is necessary as they bridge theoretical knowledge and practical applications.

The lab provides students with hands-on experience in problem-solving, enabling them to implement and test algorithms in real-world scenarios. Different algorithms have a varying impact on real-world scenarios by optimizing processes, enhancing decision-making, and solving complex problems. For example, search algorithms drive web searches, making information accessible instantly. Sorting and searching along with efficient data management are used in many search and recommendation engine applications. Each algorithm's ability to process and analyze data directly affects the efficiency, accuracy, and innovation of real-world applications.

Course Outcomes-CO

At the end of the course students will be able to:

i. Theorize and reflect on the properties, operations and applications of data structures and algorithms for the given application case studies.

ii. Implement the various data structures and algorithms and perform efficiency analysis.

iii. Explain causes and analyze the role of algorithms in structuring and manipulating data for problem-solving and evaluate on their time and space efficiency.

iv. Design and generalize solutions for real-world problems; optimize existing code using algorithms and data structures.

KLE Technological University | Creating Value, Leveraging Knowledge

Department of Computer Science and Engineering(AI)

Course Articulation Matrix: Mapping of Course Outcomes (CO) Program outcomes

Laboratory (Course) Title:Algorithms Lab Laboratory (Course) code:24ECSP206 Semester:III Year:2025-26

Course outcomes-CO/Program Outcomes-PO	1	2	3	4	5	6	7	8	9	10	11	12	13
1. Theorize and reflect on the properties, operations and applications of data structures and algorithms for the given application case studies.	**H**	M	M										
2. Implement the various data structures and algorithms and perform efficiency analysis	**H**	M	M										
3. Explain causes and analyze the role of algorithms in structuring and manipulating data for problem-solving and evaluate on their time and space efficiency.	**H**	M											
4. Design and generalize solutions for real-world problems; optimize existing code using algorithms and data structures.		**H**	**H**		M	M			M			M	

Degree of compliance **L**: Low **M**: Medium **H**: High

Competency addressed in the Course and corresponding Performance Indicators

Competency	Performance Indicators	Course Specific Performance Indicators (CSPI)
1.1: Demonstrate competence in mathematics	1.1.3 Apply probability and statistics to model and analyze data for informed decision making.	1.1.3.1 Apply probability and statistics to model and analyze data structure and algorithmic problems for informed decision making.
	1.1.3 Apply concepts of mathematical structures to solve complex systems in engineering	1.1.3.1 Use summations, series and set up recurrence relations and solve it for analyzing algorithmic efficiency for iterative and recursive problems. 1.1.3.2 Apply tree and graph principles for structured and un-structured data management. 1.1.3.3 Apply mathematical structures to solve sorting, searching and graph problems
1.3: Demonstrate competence in engineering fundamentals	1.3.2 Apply the principles of computer science including algorithms, data structures and programming to develop a solution for complex engineering problems	1.3.2.1 Apply object oriented programming concepts and principles to solve data structure and algorithm design problems 1.3.2.2 Apply principles of computer science algorithms, data structures and programming to develop solution for algorithmic real-world scenarios
2.1: Demonstrate an ability to identify and define complex engineering problems by understanding real-world contexts and constraints.	2.1.2: Identify critical systems/sub systems, variables, and constraints that characterize the problem environment by analyzing practical contexts and real-world limitations.	2.1.2.1: Identify critical systems/sub systems that characterize the problem environment by analyzing data structures, algorithm design techniques and real-world limitations..

2.2 Demonstrate an ability to apply principles of mathematics, natural sciences, and engineering to formulate and analyze complex problems, incorporating literature review to support sustainable development.	2.2.1: Decompose complex problems into interconnected sub-problems.	2.2.1.1: Decompose algorithmic problems into interconnected sub-problems.
3.2: Demonstrate an ability to generate a diverse set of alternative design solutions.	3.2.3 Identify suitable criteria for the evaluation of alternate design solutions	3.2.3.1 Identify space and time efficiency in evaluating design techniques
3.3: Demonstrate an ability to select the optimal design scheme for further development	3.3.1 Apply suitable criteria to select optimal design solutions for further development	3.3.1.1 Apply design technique to select optimal design solutions for optimization
5.2: Demonstrate an ability to evaluate the suitability and limitations of the tools used for engineering activity.	5.2.2 Select tool based on suitability and limitations of tools for performing engineering activities	5.2.2.1 Select engineering tools, apply them by selecting based on the suitability and limitations in performing engineering activities.
6.1 Demonstrate an ability to analyse societal and environmental aspects of computing solutions.	6.1.2Analyze the implications of computing solutions on environmental integrity, resource consumption, and ecological sustainability.	6.1.2.1 Analyze how algorithmic solutions used in city design such as transportation planning, resource allocation, and waste management impact the environment, use of resources, and long-term sustainability
9.2: Demonstrate effective individual and team operations: communications, problem-solving, conflict resolution and leadership skills	9.2.1. Demonstrate effective communication, problem-solving, conflict resolution and leadership skills	9.2.1.1 Demonstrate effective communication,apply problem solving strategies,analyze and resolve conflicts.
12.2: Demonstrate an ability to Identify changing trends in engineering knowledge and practice	12.2.2 Recognize the need and be able to clearly explain why it is vitally important to keep current regarding new developments in your field	12.2.2.1 Recognize the importance of current field,apply learning practices, analyze the impact of new developments.

Eg: 1.2.3: Represents program outcome ‘1’, competency ‘2’ and performance indicator ‘3’.

KLE Technological University | Creating Value, Leveraging Knowledge

Department of Computer Science and Engineering(AI)

Experiment wise plan

1. ***List of experiments/jobs planned to meet the requirements of the course.***

Category: Exercises	***Total Weightage: 05***	***No. of lab sessions: 02***

Learning Outcomes :

The students should be able to:

1. Gain the ability to implement various data structures such as arrays, linked lists, trees, and graphs.
2. Design and implement algorithms that efficiently solve computational problems using appropriate data structures.
3. Perform efficiency analysis, evaluating the time and space complexity of implemented algorithms to ensure optimal performance.

Expt./Job No.	***Experiment/job Details***	***No. of Lab. Session/s per batch (estimate)***	***Marks/ Experiment***	***Correlation of Experiment with the theory***
1	Efficiency Analysis	*1*	*2.5*	*Efficiency Analysis and Basic DS and Algorithms Implementations*
2	Implementation of DFS and BFS with appropriate data structures	*1*	*2.5*	

Category: Structured Enquiry	***Total Weightage: 35***	***No. of lab sessions:08***

Learning Outcomes:

The students should be able to:

1. .Theorize the properties and operations of various data structures and algorithms, understanding their fundamental characteristics.
2. Critically reflect on the applications of data structures and algorithms, assessing their suitability for solving specific problems within the given case studies.
3. Apply their theoretical knowledge to analyze and evaluate data structures and algorithms in the context of real-world case studies, identifying the most effective solutions.
4. Explain the causes and analyze the role of algorithms in structuring and manipulating data for effective problem-solving.
5. Evaluate the time and space efficiency of algorithms,assessing their performance in different problem-solving scenarios.

.

Expt./Job No.	***Experiment/job Details***	***No. of Lab. Session/s per batch (estimate)***	***Marks/Experiment***	***Correlation of Experiment with the theory***
1	Trees and Applications	*1*	*5*	*Trees and Tree Algorithms (BST, AVL, 2-3, Red-Black)*
2.	Array Query Algorithms	*2*	*5*	*Fenwick Tree, Sparse Table, Look-up Table*
3.	Sorting and Searching Applications	*2*	*10*	*Sorting and Searching Algorithms*
4.	Graph Algorithms	*3*	*15*	*Traversals, Shortest Path and Spanning Tree Algorithms*

Category: Project ***Total Weightage: 40*** ***No. of lab sessions:04***

Learning Outcomes :

The students should be able to:

1. Design solutions for real-world problems by applying appropriate algorithms and data structures.
2. Generalize these solutions to apply to a broader range of similar problems, ensuring adaptability and scalability.
3. Optimize existing code by refining algorithms and data structures to enhance performance and efficiency.

Expt./Job No.	***Experiment/job Details***	***No. of Lab. Session/s per batch (estimate)***	***Marks/ Experiment***	***Correlation of Experiment with the theory***
1.	Design Plan	*01*	*05*	*ALL Clusters*
2.	Identify the scenarios	*01*	*05*	*ALL Clusters*
3.	Identify the data structures and algorithms for each scenarios	*01*	*10*	*ALL Clusters*
4.	Implement and optimize the scenarios	*01*	*20*	*ALL Clusters*

5. ***Materials and Resources Required*:**

Text Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, Fourth Edition, The MIT Press, 2022.
2. Anany V. Levitin, Introduction to the Design and Analysis of Algorithms. Addison-Wesley Longman Publishing Co, 2012.

References

1. Hemant Jain, Problem Solving Using Data and Algorithms Using C, Taran Technologies Private Limited, 2016.
2. HackerRank / CodeChef / SPOJ

1. Manuals:
   a. Kindly refer to the lab manual provided for further details
2. Others:
   a. LeetCode, Link: https://leetcode.com/
   b. HackerEarth, Link: https://www.hackerearth.com/

Rubrics for Lab Evaluations: (As mapped to PI)

Criteria	5 Points	4 Points	3 Points	2 Points	1 Point
Design Solutions	Design is highly innovative, thoroughly addresses the problem, and includes comprehensive, well-structured solutions with clear justifications.	Design is effective, addresses the problem well, and includes mostly structured solutions with some justifications.	Design is adequate, addresses the problem with basic solutions, but may lack depth or clear justification.	Design partially addresses the problem with incomplete solutions and minimal justification.	Design does not adequately address the problem and lacks structured solutions and justification.
Generalize Approaches	Solutions are excellently generalized to a wide range of similar problems with clear, logical explanations.	Solutions are well generalized to similar problems with good explanations and logical reasoning.	Solutions are somewhat generalized with basic explanations, but may not cover a broad range of problems.	Solutions are minimally generalized with limited explanations and coverage.	Solutions are not generalized or lack relevant explanations.
Optimize Existing Code	Code is highly optimized with significant improvements in performance and efficiency, demonstrating advanced understanding of algorithms and data structures.	Code is well optimized with noticeable improvements in performance and efficiency, showing a good grasp of algorithms and data structures.	Code is somewhat optimized with moderate improvements in performance and efficiency, showing basic understanding.	Code shows minimal optimization with slight improvements, reflecting limited understanding of algorithms and data structures.	Code is not optimized or shows no significant improvement in performance or efficiency.

Explain and Analyze Algorithms	Thoroughly explains and analyzes the role of algorithms in data structuring and manipulation with insightful and accurate assessments of efficiency.	Effectively explains and analyzes algorithms with clear understanding and accurate assessments of efficiency.	Provides a basic explanation and analysis of algorithms with some understanding of efficiency, though may lack depth.	Explanation and analysis are minimal with limited understanding of algorithms and efficiency.	Explanation and analysis are unclear or incorrect, with minimal understanding of algorithms and efficiency.
Evaluate Efficiency	Provides a comprehensive evaluation of time and space efficiency with detailed and accurate analysis of different algorithms.	Evaluates time and space efficiency well, with a good level of detail and accurate analysis.	Provides a basic evaluation of efficiency with some detail, though analysis may lack depth.	Minimal evaluation of efficiency with limited detail and understanding.	Evaluation of efficiency is unclear or incorrect, with little to no analysis.

Project Description:

Project is designed to provide hands- on experience in the design,Implementation, and analysis of algorithms. Students will learn how to apply algorithmic techniques to solve computational problems efficiently, evaluate performance, and optimize solutions for real world applications.

1. To design and implement algorithms for searching, sorting, graph processing and optimization.
2. To analyze the time and space efficiency of algorithms.
3. To apply divide and conquer,greedy,dynamic programming, and backtracking approaches to solve
4. complex problems.
5. To generalize algorithmic solutions for real world case studies.

Description for Project:

In this project, students design a simple city and apply data structures and algorithms to solve challenges such as traffic flow, resource distribution, and waste management. By doing so, they explore how computational solutions can support the creation of safe, inclusive, resilient, and sustainable cities, aligning directly with SDG 11: Sustainable Cities and Communities. Algorithms such as shortest path, graph traversal, sorting, searching, query etc. help in planning efficient transportation, reducing congestion, and optimizing resources and other aligned problems. This project highlights the role of algorithmic thinking in problem solving and in promoting sustainability within urban environments.

KLE Technological University | Creating Value, Leveraging Knowledge

Department of Computer Science and Engineering(AI)

Rubrics for Project: (As mapped to PI)

Criteria	5 Points	4 Points	3 Points	2 Points	1 Point
Design	Design is highly innovative, thoroughly addresses the problem with a well-structured, comprehensive approach.	Design is effective, addresses the problem well, and is mostly structured with good coverage.	Design is adequate, addresses the problem with basic solutions, lacking depth or clarity.	Design partially addresses the problem with incomplete or disorganized solutions.	Design does not adequately address the problem, lacking structure and clarity.
Identification of Data Structures and Algorithms	Accurately identifies and uses appropriate data structures and algorithms, demonstrating deep understanding.	Identifies and uses suitable data structures and algorithms with good understanding.	Identifies and uses data structures and algorithms with basic understanding, some errors possible.	Identifies and uses data structures and algorithms with limited accuracy or understanding.	Incorrectly identifies or uses data structures and algorithms, demonstrating minimal understanding.
Using Principles and Generalizing	Applies principles effectively, generalizes solutions to a wide range of problems with clear logic and relevance.	Applies principles well and generalizes solutions to similar problems with good logic and explanation.	Applies principles with basic generalization, but may not cover all relevant scenarios thoroughly.	Applies principles minimally with limited generalization and logical coverage.	Fails to apply principles effectively or generalize solutions, with unclear or irrelevant logic.
Implementation	Implements solutions flawlessly with clean, efficient code and thorough testing, reflecting a high level of skill.	Implements solutions effectively with mostly clean code and good testing, demonstrating strong skills.	Implements solutions with basic functionality, but code may lack efficiency or completeness.	Implements solutions with some functionality, but code is inefficient or incomplete.	Implements solutions poorly with faulty or inefficient code, demonstrating minimal skill.
Efficiency Analysis	Provides a comprehensive and accurate analysis of time and space efficiency, demonstrating deep insight.	Provides a clear and accurate analysis of time and space efficiency with good understanding.	Provides a basic analysis of efficiency with some understanding, but may lack depth.	Provides minimal analysis of efficiency with limited detail and understanding.	Provides unclear or incorrect analysis of efficiency, with little understanding.
Transfer of Knowledge	Demonstrates exceptional ability to transfer and apply knowledge to new problems or contexts, showing insight and adaptability.	Effectively transfers and applies knowledge to new problems with good understanding and adaptability.	Transfers and applies knowledge with basic effectiveness, showing some adaptability.	Transfers and applies knowledge with limited effectiveness and minimal adaptability.	Fails to transfer or apply knowledge effectively to new problems, showing minimal adaptability.

6. ***Evaluation***:

Students Assessment through ISA (80%) + ESA (20%)

In Semester Assessment (80%)	**Assessment**	**Weightage in Marks**
	Exercise	05
	Structured Enquiry	35
	Project Work	40
End Semester Assessment (20%)	Implementation	**20**
	Set 1 - Bubble sort, Insertion sort, Selection sort, Brute Force String Search, Floyd, Warshall algorithm.	**04**
	Set 2 - Quicksort, Mergesort, Bi nary Search Tree(Insert,Delete, One traversal),Heap, DFS, BFS.	**08**
	Set 3 - Prims, Kruskal, Dijkstra a lgorithm.	**08**
	Total	**100**

Date:13 Sept 2025 **HOD CSE**