

Course Code	Course name	L	T	P	C
<b>CSEG2021</b>	<b>Design and Analysis of Algorithms</b>	3	0	0	3
<b>Total Units to be Covered: 6</b>		<b>Total Contact Hours: 45</b>			
<b>Prerequisite(s):</b>	Data structures and algorithms	<b>Syllabus version: 1.0</b>			

### Course Objectives:

The objectives of this course are as follows:

1. Define the fundamental concepts, definitions and terminologies related to algorithms, data structures, and algorithm analysis.
2. Understand the principles behind various algorithm design techniques.
3. Apply algorithms analysis techniques to evaluate the efficiency and asymptotic performance of algorithms in terms of time and space complexity.
4. Analyse and compare different algorithmic solutions for the same problem, considering their efficiency, correctness, and suitability for specific scenarios.

### Course Outcomes

The outcomes of this course are as follows:

- CO1.** Demonstrate a solid understanding of fundamental concepts, terminologies, and principles related to algorithms, data structures, and algorithm analysis.
- CO2.** Apply algorithmic design techniques to solve real-world problems.
- CO3.** Compare and contrast multiple algorithmic approaches for the same problem, considering their efficiency, correctness, and practicality.
- CO4.** Select problem-solving strategies and algorithmic thinking to tackle new and challenging problem domains and classify the algorithms in different classes.

## CO-PO Mapping

Program Outcome S Course Outcome s	PO 1	PO2	PO3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO 1	3	3	1	2									3	1	
CO 2	3	3	2	2									3	1	
CO 3	2	2	2	1									3	1	
CO 4	2	3	2	1									3	1	
Average	2.5	2.7 5	1.7 5	1.5									3	1	

**1 – Weakly Mapped (Low)**

**2 – Moderately Mapped (Medium)**

**3 – Strongly Mapped (High)**

**“\_” means there is no correlation**

## Syllabus

### Unit I: Introduction to Algorithms

**9 Lecture Hours**

Algorithm, Characteristics of algorithm, Distinct area of study of algorithm, Different algorithm design techniques, Algorithm analysis, Growth of an algorithm, Asymptotic notations, Recurrence relation, Solving recurrence relation by iteration, substitution, recursion tree, master theorem method.

### Unit II: Algorithm design paradigm: Divide and Conquer

**7 Lecture Hours**

The divide and conquer paradigm, Analysis of Binary search, Merge sort, Quick sort, Strassen Method of Matrix Multiplication, Maximum subarray problem, Powering number, Celebrity problem.

### Unit III: Algorithm design paradigm: Greedy Method

**7 Lecture Hours**

Greedy approach design paradigm, Knapsack problem, Activity selection problem, Huffman encoding, Interval partitioning problem, Dijkstra algorithm for single source shortest path problem, Prim's and Kruskal algorithm for finding minimum cost spanning tree.

#### **Unit IV: Algorithm design paradigm: Dynamic Programming    7 Lecture Hours**

Dynamic programming design paradigm, 0/1 Knapsack problem, Matrix chain multiplication problem, longest common subsequence problem, Optimal binary search problem, Bellman ford algorithm for single source shortest path problem, Travelling salesman problem, Difference between divide and conquer, greedy and dynamic programming algorithm design approach, Floyd warshall algorithm for all pair shortest path problem

#### **Unit V: Algorithm design paradigm: Backtracking and Branch & Bound**

##### **7 Lecture Hours**

Introduction to backtracking and branch & bound approach, backtracking based problems: N Queen problem, Sum of Subset problem, 0/1 Knapsack problem, Branch & Bound based problems: FIFO, LIFO, & LC branch & bound, 0/1 Knapsack problem, Travelling salesperson problem.

#### **Unit VI: Maximum Flow and String-Matching Problems                      8 Lecture Hours**

Flow networks: Ford- Fulkerson method, Maximum bipartite matching, Modulo Representation of integers/polynomials: Chinese Remainder Theorem, String Matching: The naive string-matching algorithm, The Rabin-Karp algorithm, String matching with finite automata, The Knuth-Morris-Pratt algorithm, Different classes of problems: P, NP, NP Complete, NP Hard, reducibility property

**Total lecture Hours 45**

### **Textbooks**

1. T. H. Cormen, C. F. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms", 4th Edition, MIT Press, 2022.
2. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publication, 2010.

### **Reference Books**

1. Jon Kleinberg, and Eva Tardos, "Algorithm Design", Addison Wesley, 2005.
2. A. V. Aho, J. Hopcroft, and J. D. Ullman, "The Design and Analysis of Algorithms", Addison-Wesley, 2002.

### **Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination**

#### **Examination Scheme**

<b>Components</b>	<b>IA</b>	<b>MID SEM</b>	<b>End Sem</b>	<b>Total</b>
Weightage (%)	50	20	30	100

#### **Detailed breakup of Internal Assessment**

<b>Internal Assessment Component</b>	<b>Weightage in calculation of Internal Assessment (100 marks)</b>
Quiz 1	15%
Quiz 2	15%
Class Test 1	15%
Class Test 2	15%
Assignment 1/Project	20%
Assignment 2/Project	20%



Course Code	Course name	L	T	P	C
CSEG2121	Design and Analysis of Algorithms Lab	0	0	2	1
Total Units to be Covered:		Total Contact Hours:			
Prerequisite(s):	Data structures and algorithms Lab	Syllabus version: 1.0			

## Course Objectives

The objectives of this course are as follows:

1. Apply various algorithmic strategies, such as greedy algorithms, divide and conquer, and dynamic programming, to solve problems.
2. Analyse the time complexity, space complexity, and performance characteristics of algorithms.

## Course Outcomes

The outcomes of this course are as follows:

**CO1.** Select ideal design approach based on the problem.

**CO2.** Critically analyse and evaluate algorithms based on their time complexity, space complexity, and performance characteristics.

**CO3.** Compare and contrast multiple algorithmic approaches for the same problem, considering their efficiency, correctness, and practicality.

## CO-PO Mapping

Program Outcomes Course Outcome s	PO1	PO 2	PO3	PO4	PO 5	PO6	PO 7	PO8	PO9	PO1 0	PO11	PO12	PSO1	PSO2	PSO3
CO 1	2	2	2	2									3		
CO 2	2	2	2	2									3		
CO 3	2	2	2	2									3		
Average	2	2	2	2									3		

**1 – Weakly Mapped (Low)**

**2 – Moderately Mapped (Medium)**

### **3 – Strongly Mapped (High)**

“\_” means there is no correlation

## **Syllabus**

### **List of Experiments**

#### **Experiment 1-4: Divide and Conquer Approach**

- Implement the iterative and recursive Binary search tree and compare their performance.
- Implement divide and conquer based merge sort and quick sort algorithms and compare their performance for the same set of elements.
- Compare the performance of Strassen method of matrix multiplication with traditional way of matrix multiplication.

#### **Experiment 5-9: Greedy & Dynamic Programming Approach**

- Implement the activity selection problem to get a clear understanding of greedy approach.
- Get a detailed insight of dynamic programming approach by the implementation of Matrix Chain Multiplication problem and see the impact of parenthesis positioning on time requirements for matrix multiplication.
- Compare the performance of Dijkstra and Bellman ford algorithm for the single source shortest path problem.
- Through 0/1 Knapsack problem, analyze the greedy and dynamic programming approach for the same dataset.

#### **Experiment 10-13: Backtracking and Branch & Bound Approach**

- Implement the sum of subset and N Queen problem.
- Compare the Backtracking and Branch & Bound Approach by the implementation of 0/1 Knapsack problem. Also compare the performance with dynamic programming approach.

### Experiment 14-15: String Matching Problems

- Compare the performance of Rabin-Karp, Knuth-Morris-Pratt and naive string-matching algorithms.

**Total Lab hours 15**

#### Textbooks

1. T. H. Cormen, C. F. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms", 4th Edition, MIT Press, 2022.
2. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publication, 2010.

#### Reference Books

1. Jon Kleinberg, and Eva Tardos, "Algorithm Design", Addison Wesley, 2005.
2. A. V. Aho, J. Hopcroft, and J. D. Ullman, "The Design and Analysis of Algorithms", Addison-Wesley, 2002.

**Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination**

**Examination Scheme:** Continuous Assessment

Components	Quiz & Viva	Performance & Lab Report
Weightage (%)	50	50