Academic Year: 2021-22	Semester:4th
Course Name: Design and Analysis of Algorithm	Course Code:18CS42
Total Contact hours: 50	Credits:3
SEE Marks: 60; CIE:40	Total Marks: 100
Course Plan Author:	Date:

Course Prerequisites: Discrete Mathematics - sets, functions, relations; proofs, and proofs by induction; Boolean logic. Basic probability, Basic knowledge of Java.

Course Objectives:

This course (18CS42) will enable students to:

- Explain various computational problem solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

Course Outcomes:

CO Number	Course Outcome At the end of the course, student should be able to . . .	Blooms' Level
CO1	Describe and Analyze the computational complexity	L4
CO2	Estimate computational complexity and design the algorithm for Divide and Conquer	L6
CO3	Estimate computational complexity and design the algorithm for Greedy Methods	L6
CO4	Estimate computational complexity and design the algorithm for Dynamic Programming	L6
CO5	Estimate computational complexity and design the algorithm for Backtracking	L6

2. CO – PO Mapping

Course Outcomes	Program Outcomes														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	3	1	2	2	0	2	2	2	2	2	1	1
CO2	3	3	3	3	1	2	2	0	2	1	2	2	2	1	1
CO3	3	3	3	3	1	2	2	0	2	1	2	2	2	1	1
CO4	3	3	3	3	1	2	2	0	2	1	2	2	2	1	1
CO5	3	3	3	3	1	2	2	0	2	1	2	2	2	1	1

Program Outcome and Program Specific Outcome:

PO1	Engineering Knowledge;
PO2	Problem Analysis;
PO3	Design / Development of Solutions;
PO4	Conduct Investigations of Complex Problems;
PO5	Modern Tool Usage;
PO6	The Engineer and Society;
PO7	Environment and Sustainability;
PO8	Ethics;
PO9	Individual and Teamwork;
PO10	Communication;
PO11	Project Management and Finance;
PO12	Life-long Learning;
PSO1	Graduates will have the ability to adapt, contribute and innovate ideas in the field of Artificial Intelligence and Machine Learning
PSO2	To provide a concrete foundation and enrich their abilities to qualify for Employment, Higher studies and Research in various domains of Artificial Intelligence and Machine Learning such as Data Science, Computer Vision, Natural Language Processing with ethical values
PSO3	Graduates will acquire the practical proficiency with niche technologies and open source platforms and to become Entrepreneur in the domain of Artificial Intelligence and Machine Learning

Course Content (Syllabus)

DESIGN AND ANALYSIS OF ALGORITHMS (Effective from the academic year 2018 -2019) SEMESTER – IV			
Subject Code	18CS42	**CIE Marks**	40
Number of Contact Hours/Week	3:2:0	**SEE Marks**	60
Total Number of Contact Hours	50	**Exam Hours**	3 Hrs
CREDITS –4			
Course Learning Objectives: This course (18CS42) will enable students to:			
• Explain various computational problem solving techniques. • Apply appropriate method to solve a given problem. • Describe various methods of algorithm analysis.			
Module 1			**Contact Hours**

Introduction: What is an Algorithm? **(T2:1.1),** Algorithm Specification **(T2:1.2),** Analysis Framework **(T1:2.1), Performance Analysis**: Space complexity, Time complexity **(T2:1.3)**. **Asymptotic Notations:** Big-Oh notation (*O*), Omega notation (*fi*), Theta notation (*Θ*), and Little-oh notation (*o*), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples **(T1:2.2, 2.3, 2.4). Important Problem Types:** Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. **Fundamental Data Structures:** Stacks, Queues, Graphs, Trees, Sets and Dictionaries. **(T1:1.3,1.4).** **RBT: L1, L2, L3**	10
Module 2	
Divide and Conquer: General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum **(T2:3.1, 3.3, 3.4)**, Merge sort, Quick sort **(T1:4.1, 4.2)**, Strassen's matrix multiplication **(T2:3.8)**, Advantages and Disadvantages of divide and conquer. **Decrease and Conquer Approach:** Topological Sort. **(T1:5.3).** **RBT: L1, L2, L3**	10
Module 3	
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines **(T2:4.1, 4.3, 4.5). Minimum cost spanning trees:** Prim's Algorithm, Kruskal's Algorithm **(T1:9.1, 9.2)**. **Single source shortest paths:** Dijkstra's Algorithm **(T1:9.3)**. **Optimal Tree problem:** Huffman Trees and Codes **(T1:9.4)**. **Transform and Conquer Approach:** Heaps and Heap Sort **(T1:6.4)**. **RBT: L1, L2, L3**	10
Module 4	
Dynamic Programming: General method with Examples, Multistage Graphs **(T2:5.1, 5.2)**. **Transitive Closure:** Warshall's Algorithm, **All Pairs Shortest Paths:** Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem **((T1:8.2, 8.3, 8.4)**, Bellman-Ford Algorithm **(T2:5.4)**, Travelling Sales Person problem **(T2:5.9)**, Reliability design **(T2:5.8)**. **RBT: L1, L2, L3**	10
Module 5	
Backtracking: General method **(T2:7.1)**, N-Queens problem **(T1:12.1)**, Sum of subsets problem **(T1:12.1)**, Graph coloring **(T2:7.4)**, Hamiltonian cycles **(T2:7.5)**. **Branch and Bound:** Assignment Problem, Travelling Sales Person problem **(T1:12.2)**, **0/1 Knapsack problem (T2:8.2, T1:12.2):** LC Branch and Bound solution **(T2:8.2)**, FIFO Branch and Bound solution **(T2:8.2)**. **NP-Complete and NP-Hard problems:** Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes **(T2:11.1)**. **RBT: L1, L2, L3**	10
Course Outcomes: The student will be able to :	
• Describe computational solution to well known problems like searching, sorting etc. • Estimate the computational complexity of different algorithms. • Devise an algorithm using appropriate design strategies for problem solving.	
Question Paper Pattern:	
• The question paper will have ten questions. • Each full Question consisting of 20 marks • There will be 2 full questions (with a maximum of four sub questions) from each module. • Each full question will have sub questions covering all the topics under a module. • The students will have to answer 5 full questions, selecting one full question from each module.	
Textbooks:	

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
2. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education).

Schedule of Instruction

S.NO	Class No	Module	Topic planned	Reference& Text Book, (Page no.)	Course Outcome	Delivery mode
1.		**Module : 1**	Introduction to Subjects		C01	Black board
2.	1		Introduction to Module		C01	Black board
3.	2		**Introduction:** What is an Algorithm? **(T2:1.1),** Algorithm Specification	T2-1	C01	ICT
4.	3		Analysis Framework **(T1:2.1)**	T1-40	C01	ICT
5.	4		**Performance Analysis**: Space complexity, Time complexity **(T2:1.3)**.	T2-11	C01	ICT
6.	5		**Asymptotic Notations:** Big-Oh notation (*O*), Omega notation (_), Theta notation (Q), Little-oh notation (*o*).	T1-49	C01	ICT
7.	6		Mathematical analysis of Non-Recursive and recursive Algorithms with Examples **(T1:2.2, 2.3, 2.4**	T1-57	C01	ICT
8.	7		**Important Problem Types:** SortingSearching String processing Graph Problems Combinatorial Problems	T1-17	C01	ICT
9.	8		**Fundamental Data Structures:** Stacks, Queues, Graphs, Trees,	T1-26	C01	Black board
10.	9		Sets and Dictionaries. **(T1:1.3,1.4).**	T1-33		Black board

11.			**Module 2: Introduction**		C02	Black board
12.	1		**Divide and Conquer**		C02	Black board
13.	2		General method	T2-137	C02	ICT
14.	3		Binary search	T2-145	C02	ICT
15.	4		Recurrence equation for divide and conquer		C02	ICT
16.	5	**Module 2:**	Finding the maximum and minimum **(T2:3.1, 3.3, 3.4)**,	T2-154	C02	ICT
17.	6		Merge sort	T1-119	C02	ICT
18.	7		Quick sort **(T1:4.1, 4.2**	T1-123	C02	ICT
19.	8		Stassen's matrix multiplication **(T2:3.8)**	T2-192	C02	ICT
20.	9		Advantages and Disadvantages of divide and conquer		C02	ICT
21.	10		**Decrease and Conquer Approach**		C02	ICT
22.	11		Topological Sort. **(T1:5.3).**	T1-163	C02	Black board
23.	12		Problem Solving	T1-166	C02	Black board
24.	13		Problem solving	T1-166	C02	Black board
25.			**Module 3 -Introduction**		C03	Black board
26.	1		**Greedy Method:**, , ,		C03	Black board
27.	2		General method	T2-210	C03	ICT
28.	3		Coin Change Problem		C03	ICT
29.	4	**Module 3:**	Knapsack Problem	T2-218	C03	ICT
30.	5		Jobsequencing with deadlines **(T2:4.1, 4.3, 4.5).**	T2-227	C03	ICT
31.	6		**Minimum cost spanning trees:** Prim.s Algorithm,	T1-292	C03	ICT
32.	7		Kruskal.s Algorithm **(T1:9.1, 9.2)**.	T1-298	C03	ICT
33.	8		**Single source shortest paths:**		C03	ICT
34.	9		Dijkstra's Algorithm **(T1:9.3)**.	T1-305	C03	ICT
35.	10		**Optimal Tree problem:**		C03	ICT
36.	11		Huffman Trees and Codes **(T1:9.4)**.	T1-311	C03	ICT

37.	12		**Transform and Conquer Approach:** Heaps	T1-211	C03	Black board
38.	13		Heap Sort **(T1:6.4)**.	T1-215	C03	Black board
39.		**Module 4:**	**Module 4: Introduction**		C04	Black board
40.	1		**Dynamic Programming:**, General method with Examples	T2-272	C04	Black board
41.	3		Multistage Graphs **(T2:5.1, 5.2)**.	T2-276	C04	ICT
42.	4		**Transitive Closure:** Warshall.s Algorithm	T1-270	C04	ICT
43.	5		**All Pairs Shortest Paths:**	T1-284	C04	ICT
44.	6		Floyd's Algorithm	T1-273	C04	ICT
45.	7		Optimal Binary Search Trees	T1-278	C04	ICT
46.	8		Knapsack problem **((T1:8.2, 8.3, 8.4)**,	T1-284	C04	ICT
47.	9		Bellman-Ford Algorithm **(T2:5.4)**,	T2-289	C04	ICT
48.	10		Travelling Sales Person problem **(T2:5.9)**	T2-318	C04	ICT
49.	11		Reliability design **(T2:5.8)**.	T2-315	C04	Black board
50.	12		Problem solving		C04	Black board
51.	13		Problem solving		C04	Black board
52.		**Module :5**	**Module :5**		C05	Black board
53.	1		**Backtracking**		C05	Black board
54.	2		General method **(T2:7.1)**	T2-359	C05	Black board
55.	3		N-Queens problem **(T1:12.1)**	T1-395	C05	ICT
56.	4		Sum of subsets problem **(T1:12.1)**	T1-397	C05	ICT
57.	5		Graph coloring **(T2:7.4)**	T2-379	C05	ICT
58.	6		Hamiltonian cycles **(T2:7.5**	T2-383	C05	ICT
59.	7		**Branch and Bound:**	T1-402	C05	ICT
60.	8		Assignment Problem	T1-403	C05	ICT
61.	9		Travelling Sales Person problem **(T1:12.2)**	T1-408	C05	ICT
62.	10		**0/1 Knapsack**	T2-412	C05	ICT

			problem (T2:8.2, T1:12.2):			
63.	11		LC Branch and Bound solution **(T2:8.2)**,	T2-413	C05	ICT
64.	12		FIFO Branch and Bound solution **(T2:8.2)**.	T2-416	C05	ICT
65.	13		**NP-Complete and NP-Hard problems:**	T2-515	C05	ICT
66.	14		Basic concepts, nondeterministic algorithms	T2-516	C05	Black board
67.	15		P, NP, NP-Complete, and NP-Hard classes **(T2:11.1)**.	T2-524	C05	Black board

**L – Lecture, V- Videos or any other mode*

Textbooks	
T1	Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
T2	Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press
Reference books	
R1	Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
R2	Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education).

Web links and Video Lectures (e-Resources):	
1	https://sites.google.com/skit.org.in/4thsem/about-the-faculty
2	https://www.javatpoint.com/daa-tutorial
3	https://www.tutorialspoint.com/design_and_analysis_of_algorithms/index.html
4	
5	

Assessment Schedule:

Sl.No.	Assessment type	Contents	CO	Duration In Hours	Marks	Date & Time
1	**CIE Test 1**	Module 1,2	CO1,CO2	01:15	30	
2	**CIE Test 2**	Module 3,4	CO3,CO4	01:15	30	
	CIE Test 3	Module 5	CO5	01:15	30	
3	**Assignment 1**	Module 1,2	CO1,CO2	-	10	
4	**Assignment 2**	Module 3,4	CO3,CO4	-	10	
5	**Seminar *(or any planned activtiy)***	Module 5	CO5	-	10	

6	**Semester End Examination**					

Seminar: Group of 6-8 students
Module 1,2,3,4 & 5

****The sum of total marks of three tests, two assignments, and seminar will be out of 100 marks and will be scaled down to 50 marks.**

CIE + SEE = 40 + 60 = 100 marks

Faculty Incharge DAC Chairman

*** Please mention as per the scheme.*