FR. Conceicao Rodrigues College Of Engineering
Department of Computer Engineering
S.E. (Computer) (semester IV)
(2022-2023)

Branch: Computer Engineering
Semester:III semester **Year: 2022-2023**

Course Title: Analysis of Algorithms (CSC402)	**SEE: 3 Hours-Theory & Oral Examination**
Total contact Hours: 36 Hours	**Duration of SEE: 3 Hrs**
SEE Marks: 80 (Theory) + 20 (IA)	
Lesson Plan Author: Prajakta Dhamanskar (Div. A)	**Date:**
Checked By:	**Date:**

Course Outcomes and Assessment Plan

Prerequisite: Data structure concepts, Discrete structures	
Course Objectives:	
1	To provide mathematical approaches for Analysis of Algorithms
2	To understand and solve problems using various algorithmic approaches
3	To analyze algorithms using various methods
Course Outcomes: At the end of the course learner will be able to	
1	Analyze the running time and space complexity of algorithms.
2	Describe, apply and analyze the complexity of divide and conquer strategy.
3	Describe, apply and analyze the complexity of greedy strategy.
4	Describe, apply and analyze the complexity of dynamic programming strategy.
5	Explain and apply backtracking, branch and bound.
6	Explain and apply string matching techniques.

Syllabus:

Module		Detailed Contents	Hours
1		**Introduction**	8
	1.1	Performance analysis, space, and time complexity Growth of function, Big-Oh, Omega Theta notation Mathematical background for algorithm analysis. Complexity class: Definition of P, NP, NP-Hard, NP-Complete Analysis of selection sort, insertion sort.	
	1.2	Recurrences: The substitution method, Recursion tree method, Master method	
2		**Divide and Conquer Approach**	6
	2.1	General method, Merge sort, Quick sort, Finding minimum and maximum algorithms and their Analysis, Analysis of Binary search.	

3		**Greedy Method Approach**	6
	3.1	General Method, Single source shortest path: Dijkstra Algorithm Fractional Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees: Kruskal and Prim's algorithms	
4		**Dynamic Programming Approach**	9
	4.1	General Method, Multistage graphs, Single source shortest path: Bellman Ford Algorithm All pair shortest path: Floyd Warshall Algorithm, Assembly-line scheduling Problem0/1 knapsack Problem, Travelling Salesperson problem, Longest common subsequence	
5		**Backtracking and Branch and bound**	6
	5.1	General Method, Backtracking: N-queen problem, Sum of subsets, Graph coloring	
	5.2	Branch and Bound: Travelling Salesperson Problem, 15 Puzzle problem	
6		**String Matching Algorithms**	4
	6.1	The Naïve string-matching algorithm, The Rabin Karp algorithm, The Knuth-Morris-Pratt algorithm	

Textbooks:	
1	T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005.
2	Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.

References:	
1	Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw-Hill Edition.
2	S. K. Basu, "Design Methods and Analysis of Algorithm", PHI

Course Outcomes:

Upon completion of this course students will be able to:

CSC 402.1 :Analyze the running time and space complexity of algorithms. **(Analyze)**
CSC 402.2 : Analyze the complexity of divide and conquer strategy. **(Analyze)**
CSC 402.3 : Analyze the complexity of greedy strategy. **(Analyze)**
CSC 402.4 : Analyze the complexity of dynamic programming strategy. **(Analyze)**
CSC 402.5 : Analyze backtracking, branch and bound strategy. **(Analyze)**
CSC 402.4 : Analyze string matching techniques. **(Analyze)**

Mapping of CO and PO/PSO

Relationship of course outcomes with program outcomes: Indicate 1 (low importance), 2 (Moderate Importance) or 3 (High Importance) in respective mapping cell.

Program Specific Outcomes (PSOs)
Student will have ability to
PSO1: Develop Artificial Intelligence and Machine Learning based systems.
PSO2: Apply cyber security mechanisms to ensure the protection of Information Technology assets.

	PO1 (Engg Know)	PO2 (Ana)	PO3 (De sign)	PO4 (inve stiga)	PO5 (tools)	PO6 (engg Soci)	PO7 (Env)	PO8 (Eth)	PO9 (ind Team)	PO10 (comm.)	PO11 (PM)	PO12 (life Long)
CSC402.1	1	1										
CSC402.2	1	1										
CSC402.3	1	1										
CSC402.4	1	1										
CSC402.5	1	1										
CSC402.6	1	1										
Course To PO	1	1										

CO	**PSO1**	**PSO2**
CSC402.1		
CSC402.2		
CSC402.3		
CSC402.4		
CSC402.5		
CSC402.6		
Course to PSO		

Mapping Justification:

Course Outcome	BL	Competency	Performance Indicator	PO	Mapping
CSC402.1	4	1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1
CSC402.2	4	1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1
CSC402.3	4	1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1

CSC402.4	4	1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1
CSC402.5	4	1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1
CSC402.6		1.4 Demonstrate competence in specialized engineering knowledge to the program	1.4.1 Apply theory and principles of Computer Science and engineering to solve an engineering problem	PO1	1
		2.3 Demonstrate an ability to formulate and interpret a model	2.3.1 Able to apply computer engineering principles to formulate modules of a system with required applicability and performance.	PO2	1
		2.4 Demonstrate an ability to execute a solution process and analyze results	2.4.1 Applies engineering mathematics to implement the solution.	PO2	1

CO Measurement Weightages for Tools:

	Assessment Tool Direct (weightage: 80%)				**Assessment Tool Indirect(weightage=20%)**
CO	Test 1/2	Assignment 1/2	Quiz	SEE (T)	Course Exit Survey
CSC402.1	Test 1 20%	Assignment 1 10%	10%	60%	100%
CSC402.2	Test 1 20%	Assignment 1 10%	10%	60%	100%
CSC402.3	Test 1 20%	Assignment 1 10%	10%	60%	100%
CSC402.4	Test 2 20%	Assignment 2 10%	10%	60%	100%
CSC402.5	Test 2 20%	Assignment 2 10%	10%	60%	100%
CSC402.6	Test 2 20%	Assignment 2 10%	10%	60%	100%

CO Assessment Tools:

CSC402.1: Direct Methods(80%): Unit Test 1 + Assignment 1+Quiz+SEE(T)

CO1dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO1idm

CSC402.1 = 0.8*CO1dm + 0.2* CO1idm

CSC402.2:Direct Methods(80%): Unit Test 1 + Assignment 1+Quiz+SEE(T)

CO2dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO2idm

CSC402.2 = 0.8*CO2dm + 0.2* CO2idm

CSC402.3: Direct Methods(80%): Unit Test 1 + Assignment 1+Quiz+SEE(T)

CO3dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO3idm

CSC402.3 = 0.8*CO3dm + 0.2* CO3idm

CSC404.4: Direct Methods(80%): Unit Test 2 + Assignment 2+Quiz+SEE(T)

CO4dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO4idm

CSC402.4 = 0.8*CO4dm + 0.2* CO4idm

CSC404.5: Direct Methods(80%): Unit Test 2 + Assignment 2+Quiz+SEE(T)

CO5dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO5idm

CSC402.5 = 0.8*CO5dm + 0.2* CO5idm

CSC404.6: Direct Methods(80%): Unit Test 2 + Assignment 2+Quiz+SEE(T)

CO6dm = 0.2T +0.1Assignment+0.1Quiz +0.6SEE(T)

InDirect Methods(20%): Course exit survey

CO6idm

CSC402.6 = 0.8*CO6dm + 0.2* CO6idm

Course Level Gap (if any):

Content beyond Syllabus:

Lecture Plan						
Module 1: Introduction to Analysis of Algorithms						
No.	**Date**		**Topic**	**Hrs**	**Content Delivery Method**	**Remark**
	Planned	**Actual**				
1	09-01-2023	09-01-2023	Introduction to analysis of algorithms: Introduction to subject and fundamentals of algorithms. What is meant by an efficient algorithm?	12	Chalk and board	
2	11-1-2023	11-01-2023	Efficiency of algorithms, Time and Space Complexities Fundamentals		Chalk and board	
3	12-1-2023	12-01-2023	Calculation of time complexity for code samples		Chalk and board	
4	16-1-2023	12-01-2023	Calculation of time complexity for code samples continued		Chalk and board	
5	18-1-2023	16-01-2023	Asymptotic notation big, Omega, Theta definition		Chalk and board	

6	19-1-2023	18-01-2023	Asymptotic notations examples prove that kind of sums		Chalk and board	
7	23-1-2023	18-01-2023	properties of Asymptotic notation, best worst and average case analysis of linear search and Binary search ,writing recurrence equation		Chalk and board	
8	25-1-2023	19-01-2023	back substitution method of solving recurrence		Chalk and board	
9	30-1-2023	23-01-2023	recursion tree method		Chalk and board	
10	01-02-2023	25-01-2023	Space complexity for iterative and recursive programs		Chalk and board	
11	02-02-2023	30-01-2023	Masters method		Chalk and board	
12	06-02-2023	23-01-2023 (Lab)	Analysis of Insertion sort, Selection Sort and Optimized Bubble sort.		Chalk and board	
Module 2: Divide and Conquer Approach						
13	8-2-2023	01-02-2023	Merge Sort	4	Chalk and board	
14	9-2-2023	02-02-2023	Merge sort time and space complexity		Chalk and board	
15	13-2-2023	06-02-2023	Quick Sort algorithm, Time and Space complexity		Chalk and board	
16	16-2-2023	08-02-2023	Randomized Quick Sort, Min Max Algorithm		Chalk and board	
Module 3: Greedy Method						
17	20-2-2023	09-02-2023	General Method, Fractional Knapsack Problem	4	chalk and board, PPT.	
18	22-2-2023	13-02-2023	Job Sequencing with deadline		Chalk and board	
19	23-2-2023	15-02-2023	MST- Prims, MST – Kruskal		Chalk and board	

20	1-3-2023	16-02-2023	Dijkstra's Shortest Path Algorithm (SSSP)		Chalk and board, Visualization using Animation Video.	
Module 4: Dynamic Programming						
21	2-3-2023		General Method, 0/1 Knapsack	7	Chalk and board , Lab performance	
22	6-3-2023		All pair shortest Path(Floyd Warshall Algo)		Chalk and board	
23	9-3-2023		Single Source Shortest Path (Bellman Ford)		Chalk and board ,	
24	13-3-2023		MultiStage Graph		Chalk and board	
25	15-3-2023		Traveling Salesman Problem		Chalk and board	
26	16-3-2023		Longest common subsequence		PPT	
27	20-3-2023		Assembly line scheduling, Examples on Assembly line scheduling		Chalk and board	
Module 4: Backtracking and branch and bound2						
28	23-3-2023		General Method of backtracking, n queen problem, Introduction to graph coloring	5	Chalk and board	
29	27-3-2023		Graph Coloring program and state space tree construction, Examples for practice.		Chalk and board	
30	29-03-2023		Sum of Subsets introduction, problem solving. Sum of subset program		Chalk and board	
31	3-4-2023		General Method of branch and bound, 8 puzzle problem		Chalk and board	

32	5-4-2023		15 puzzle problem, Traveling Salesman Problem		Chalk and board	
Module 5: String Matching algorithms						
33	6-4-2023		Naïve String Matching, Rabin Karp Algo	4	Chalk and board	
34	10-4-2023		KMP Algo prefix and suffix concept		Chalk and board	
35	12-4-2023		program on KMP algo		Chalk and board	
36	13-4-2023		Revision and Doubt Solving		Chalk and board	
37	20-4-2023					

Text Books:

1. **T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", 2nd Edition, PHI Publication 2005.**
2. **Ellis Horowitz, Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press.**

Reference Books:

1. **Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw-Hill Edition.**
2. **S. K. Basu, "Design Methods and Analysis of Algorithm", PHI**

Web References:

1. **https://nptel.ac.in/courses/106/106/106106131/**
2. **https://swayam.gov.in/nd1_noc19_cs47/preview**
3. **https://www.coursera.org/specializations/algorithms**
4. **https://www.mooc-list.com/tags/algorithms**

Evaluation Scheme

CIE Scheme

Internal Assessment: 20 (Average of two tests)

Internal Assessment Scheme

Module		Lecture Hours	No. of questions in			No. of questions in SEE
			Test 1	Test 2	Test 3*	
1	Introduction	8	01 (10 marks)	-	--	
2	Divide and Conquer Approach	6	01 (5 Marks)	-	--	
3	Greedy Method Approach	6	01 (5 Marks)	-	--	

4	Dynamic Programming Approach	9		01 (15 Marks)	--	
5	Backtracking and Branch and bound	6	-	01 (5 Marks)	--	
6	String Matching Algorithms	4	-	01 (5 Marks)	--	

Note: Four to six questions will be set in the Test paper

Verified by:	
Programme Coordinator	Subject Expert

Rubrics for Assignment

Indicator	Excellent	Good	Average	Below average
Timeline (2)	submitted on time or early (2)	Submitted next day (1)	Submitted in same week (0.5)	Submitted in next week (0)
Organization (2)	Well organized, neat and clear handwriting, neat diagrams with all labels. (2)	Organized to some extent, diagrams and handwriting is neat with some missing labels (1)	Organization not appropriate, diagrams are incomplete with some missing labels (0.5)	Poorly organized, diagrams incomplete (0)
Level of content (3)	All points are covered and answered accurately (3)	Some important points are omitted / addressed minimally (2)	Many important points are missing and the ones which are written are addresses in brief. (2-1)	Many important points are missing and the answers are not accurate. (1-0.5)
Knowledge about the topic (3)	All Concepts of a topic are clear and knows the application to real world problems (3)	All Concepts of a topic are mostly clear, lacks understanding about the application to real world problems (2)	Concepts of a topic are not understood clearly, lacks understanding about the application to real world problems (2-1)	Poor understanding of concepts and application to real world problems. (1-0.5)

Rubrics for Lab Experiments

Performance Indicator	**Excellent**	**Good**	**Average**	**Below Average**
Coding Standards [4M]	The code adheres to all standards. The code is exceptionally well organized and very easy to follow. Comments are complete and useful; variables' purposes are clearly communicated by their names. [4 marks]	There may be some minor failures to adhere to standards, for instance, indentation may be inconsistent, some lines may be too long, or a few variables may have unobvious names or be undocumented. [3 marks]	The code fails to adhere to standards at multiple locations indentation is inconsistent throughout the program, Many variable names are vague, comments are missing . [2 marks]	There are major problems with the program's design or coding style that would interfere with its comprehension, reuse, or maintenance. The code may be poorly formatted. [0.5-1M]
Output validation [2M]	Output is obtained for different cases of input.[2M]	Output is obtained only for some subsets of input.[1M]	Output is obtained only for some subsets of input, incorrect output for few test cases[0.5M]	no output is obtained. [0 mark]
Post Lab Questions [2M]	Answers to all questions are correct and explained in depth. [2 marks]	Answers to most of the questions are correct but not explained in depth. [1 marks]	Few answers are incorrect [0.5M]	Answers to most of the questions are incorrect. [0 mark]
Promptness [2M]	The laboratory report is submitted on time [2 mark]	The laboratory report is submitted next day. [1 marks]	-	The laboratory report is submitted in next practical session. [0 marks]

Rubrics for Asignments:

Indicator	**Excellent**	**Good**	**Below average**
Timeline (2)	submitted on time or early (2)	Submitted next day (1)	Submitted in same week (0.5)
Organization (2)	Well organized, neat and clear handwriting, neat diagrams with all labels. (2)	Organized to some extent, diagrams and handwriting is neat with some missing labels (1)	Poorly organized, diagrams incomplete (0.5)
Level of content (3)	All points are covered (3) and answered accurately	Some important points are omitted / addressed minimally (1-2)	Many important points are missing and the answers are not accurate. (1-0)
Knowledge about the topic (3)	All Concepts of a topic are clear and knows the application to real world problems (3)	All Concepts of a topic are mostly clear lacks understanding about the application to real world problems (2-1)	Poor understanding of concepts and application to real world problems. (1-0)

AOA Assignment 1

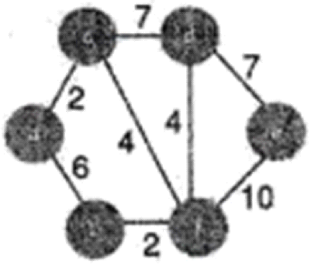
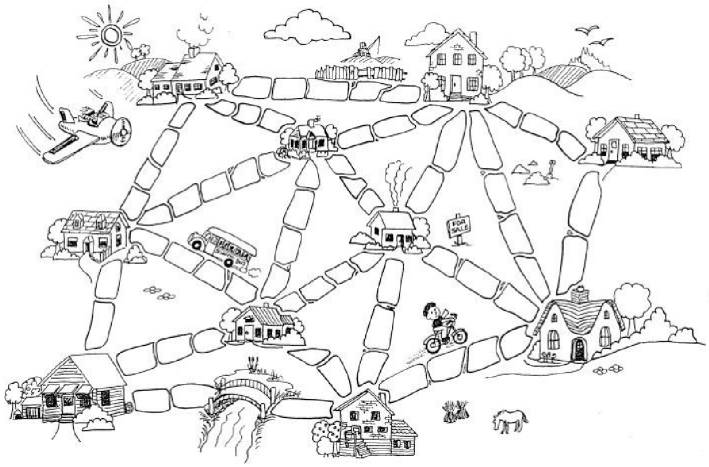
CLASS: SE COMPS (SEM IV) **YEAR: 2022-23**

Sr. No.	Question	CO	BL	PI
1	Solve following recurrence relation using back substitution method. 1. $T(n)=2T(\frac{n}{2})+1$.. 2. $T(n)=2T(\frac{n}{2})+3n^2$.	CSC402.1	3	1.4.1, 2.3.2, 2.4.1
2	Solve the following r..ecurrence by recursion tree method. 1. $T(n)=T(\frac{n}{2})+)+T(\frac{n}{4})$.+$T(\frac{n}{8})+n$ 2. $T(n)= 3T(\frac{n}{4})+cn^2$	CSC402.1	3	1.4.1, 2.3.2, 2.4.1
3	Apply Master method to derive complexity, clearly mention the cases applied $T(n) = 64T(n/8)-n^2 \log n$ $T(n) = 6T(n/4)+ n \sqrt{n}$ $T(n) = 2 T(n/2) + n/\log n$	CSC402.1	3	1.4.1, 2.3.2, 2.4.1

4	Analyze the time complexity of following code.
	a)
	int fun(int n)
	{
	int count = 0;
	for (int i = n; i > 0; i /= 2)
	for (int j = 0; j < i; j++)
	count += 1;
	return count;
	}
	b)
	for(i=2 to m-1)
	{
	for(j=3 to i)
	{
	sum=sum+A[i][j]
	}
	}
	
	c)
	void fun(int n, int arr[]){
		CSC402.1	4	1.4.1, 2.3.2, 2.4.1

	int i = 0, j = 0;
	for(; i < n; ++i)
	while(j < n && arr[i] <
	arr[j]) j++;
	}			

5	Given N events with their starting and ending times, find a schedule that includes as many events as possible using greedy strategy. It is not possible to select an event partially. Consider the below events event starting time ending time A 1 3 B 2 5 C 3 9 D 6 8	CSC402.3	4	1.4.1, 2.3.2, 2.4.1
6	Given the following jobs, their deadlines and associated profits as shown- Jobs J1 J2 J3 J4 J5 J6 J7 Deadlines 1 3 4 3 2 1 2 Profits 3 5 20 18 1 6 30 Answer the following questions- • Write the optimal schedule that gives maximum profit. • What is the maximum earned profit?	CSC402.3	3	1.4.1, 2.3.2, 2.4.1
7 a	identify an algorithm a telecommunications company will use for laying cable to a new neighborhood. If it is constrained to bury the cable only along certain paths (e.g. along roads), then there would be a graph representing which points are connected by those paths. Some of those paths might be more expensive, because they are longer, or require the cable to be buried deeper; these paths would be represented by edges with larger weights. The objective of the company is to cover all locations using a minimum path.	CSC402.3	4	1.4.1, 2.3.2, 2.4.1
7 b	Consider the following graph for the above problem statement and find the minimum path covering all locations.	CSC402.3	3	1.4.1, 2.3.2, 2.4.1

				
8	Once upon a time there was a city that had no roads. Getting around the city was particularly difficult after rainstorms because the ground became very muddy, cars got stuck in the mud and people got their boots dirty. The mayor of the city decided that some of the streets must be paved, but did not want to spend more money than necessary because the city also wanted to build a swimming pool. The mayor therefore specified two conditions:1. Enough streets must be paved so that it is possible for everyone to travel from their house to anyone else's house only along paved roads, and 2. The paving should cost as little as possible. Here is the layout of the city. The number of paving stones between each house represents the cost of paving that route. Find the best route that connects all the houses, but uses as few counters (paving stones) as possible. 	CSC402.3	4	1.4.1, 2.3.2, 2.4.1
9 a	Identify and apply the **fastest** sorting algorithm based on Divide and Conquer strategy on the following array. **Show all the steps in the first iteration.** [27, 10, 36, 18, 25, 45]	CSC402.2	3	1.4.1, 2.3.2, 2.4.1

9 b	b) Identify and apply the **stable** sorting algorithm based on Divide and Conquer strategy on the following array. **Show all the steps.** [57, 23, 89, 55, 12, 40,25]	CSC402.2	3	1.4.1, 2.3.2, 2.4.1
10	Compare the complexities of Optimized Linear search, Binary search, Bubble, Selection, Insertion, Quick, randomized quick and Merge for Best, Average and Worst case in tabular form.	CSC402.2	3	1.4.1, 2.3.2, 2.4.1
11	Apply the greedy strategy to solve minimum coin change problem. (Content beyond syllabus)	**CSC402.3**	**3**	**1.4.1, 2.3.2, 2.4.1**

Last Date of Submission:27th March 2023

FR. CONCEICAO RODRIGUES COLLEGE OF ENGG.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400 050.

I UNIT TEST

SEMESTER / BRANCH: IV/COMPUTER-Div. **A&B**
SUBJECT: Analysis of Algorithms (AOA) MAX. MARKS: 20
DATE: 03/03/2023 TIMING: 1.00 pm to 2.00 pm

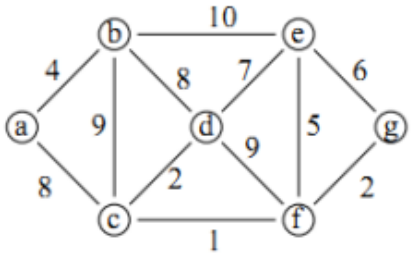
Student should be able to

CSC401.1	Analyze the running time and space complexity of algorithms.
CSC401.2	Analyze the complexity of divide and conquer strategy.
CSC401.3	Analyze the complexity of greedy strategy.

Q.NO	Questions	MARKS	CO	BL	PI
1.A	Analyze the running time complexity of following code. a.	2	CSC401.1	4	1.4.1, 2.3.2, 2.4.1

```
for ( i=1; i < n; i *= 2 ) {
  for ( j = n; j > 0; j /= 2 ) {
    for ( k = j; k < n; k += 2 ) {
      sum += (i + j * k );
    }
  }
}
```

	b.
	`int fun(int n)`
	`{`
	`  int count = 0;`
	`  for (int i = 0; i < n; i++)`
	`  for (int j = i; j > 0; j--)`
	`  count = count + 1;`
	`  return count;`
	`}`
		2	CSC401.1	4	1.4.1, 2.3.2, 2.4.1

1.B	Solve the following recurrence using the Back Substitution Method. $T(n)=T(n-1)+n^4$	2	CSC401.1	3	1.4.1, 2.3.2 2.4.1
1 C	Solve the following recurrence using Recursion Tree Method $T(n)=2T(\frac{n}{2}) + n^2$	2	CSC401.1	3	1.4.1, 2.3.2 2.4.1
1 D	Solve the following recurrence using Master's Method. Clearly mention the cases applied. $T(n)= 4T(\frac{n}{2}) + n^2\sqrt{n}$	2	CSC401.1	3	1.4.1, 2.3.2 2.4.1
2.A	Identify and apply the **fastest** sorting algorithm based on Divide and Conquer strategy on the following array. **Show all the steps in the first iteration.** [40, 11, 4, 72, 17, 2, 49]	3	CSC401.2	3	1.4.1, 2.3.2 2.4.1
2.B	Analyze the time complexity of Worst Case of the above.	2	CSC401.2	4	1.4.1, 2.3.2 2.4.1
3.A	You have a business with several offices; you want to lease phone lines to connect them up with each other, and the phone company charges different amounts of money to connect different pairs of cities. You want a set of lines that connects all your offices with a minimum total cost. Use a suitable algorithm to be used in this scenario to find the optimal solution.  	4	CSC401.3	4	1.4.1, 2.3.2 2.4.1
3.B	Analyze the time complexity of the above algorithm.	1	CSC401.3	4	1.4.1, 2.3.2 2.4.1
	OR				

<table>
<tr>
<td>3.A</td>
<td>Mr. Shashi owned a shipping company in India.One of his trucks with a weight capacity of 150 kg is about to be loaded, Awaiting shipments are the items in the following table. Each of these items has an associated value in rupees and weight.The objective is to Maximize the total value of the items loaded onto the truck without exceeding the truck’s weight capacity.
<table>
<tr><th>Item</th><th>Value</th><th>Weight in kg</th></tr>
<tr><td>1</td><td>150</td><td>50</td></tr>
<tr><td>2</td><td>145</td><td>45</td></tr>
<tr><td>3</td><td>135</td><td>30</td></tr>
<tr><td>4</td><td>145</td><td>35</td></tr>
<tr><td>5</td><td>120</td><td>40</td></tr>
<tr><td>6</td><td>70</td><td>35</td></tr>
</table>
</td>
<td>4</td>
<td>CSC401.3</td>
<td>4</td>
<td>1.4.1, 2.3.2 2.4.1</td>
</tr>
<tr>
<td>3.B</td>
<td>Analyze the time complexity of the above algorithm.</td>
<td>1</td>
<td>CSC401.3</td>
<td>4</td>
<td>1.4.1, 2.3.2, 2.4.1</td>
</tr>
</table>

*BL – Bloom’s Taxonomy Levels (1- Remembering, 2- Understanding, 3 – Applying, 4 – Analysing, 5 – Evaluating, 6 - Creating)

*CO – Course Outcomes

*PO – Program Outcomes;

*PI Code – Performance Indicator Code

BL Distribution PIE chart and CO distribution bar chart (Following diagram is just for reference purpose only)