Practical Plan

Branch: Computer Engineering

Semester: IV **Year: 2022-23**

Course Title: **Analysis of Algorithms (CSL401)**	SEE: 2 Hours – Practical
Total Contact Hours: 20 Hours	
Practical Plan Author: **Prof. Ashwini Pansare (Div. B)**	Date:10 th January 2023
Checked By: Prof. Kalpana Deorukhkar	Date:

Prerequisites: Basic knowledge of programming and data structure

Course Outcomes (CO):

On successful completion of course learner will be able to:

CSL401.1 Implement the algorithms using different approaches.
CSL401.2 Analyze the complexities of various algorithms.
CSL401.3 Compare the complexity of the algorithms for specific problems.

List of Experiments

Sr. No.	TITLE	Mapped Co
1	WAP to implement Modified bubble sort, Insertion sort, Selection sort and derive its complexity.	CSC401.1 CSC401.2
2	WAP to implement Linear search and binary search and derive its time complexity.	CSC401.1 CSC401.2
3	WAP to implement Quick sort, randomized quick sort and derive its complexity.	CSC401.1 CSC401.2
4	WAP to implement Merge sort and derive its complexity.	CSC401.1 CSC401.2
5	WAP to implement MinMax Algorithm using Divide and Conquer.	CSC401.1 CSC401.2
6	WAP to implement fractional knapsack using greedy methods.	CSC401.1 CSC401.2
7	WAP to implement Dijkstra's Shortest Path algorithm using greedy methods.	CSC401.1 CSC401.2
8*	WAP to implement Prim's algorithm of MST (*Newly added Experiment)	CSC401.1 CSC401.2
9	WAP to implement 0/1 knapsack using dynamic programming.	CSC401.1 CSC401.2
10	WAP to implement Bellman Ford Algorithm using Dynamic Programming.	CSC401.1 CSC401.2
11	WAP to implement Floyd Warshall algorithm.	CSC401.1 CSC401.2

12	WAP to implement the N queen problem using a backtracking approach.	CSC401.1 CSC401.2
13	WAP to implement sum of subset problem using backtracking approach	CSC401.1 CSC401.2
14	WAP to implement Graph Coloring using backtracking approach	CSC401.1 CSC401.2
15	WAP to implement Longest Common Subsequence using DynamicProgramming.	CSC401.1 CSC401.2

CO-PO Mapping: (BL – Blooms Taxonomy, C – Competency, PI – Performance Indicator)

CO	BL	C	PI	PO	Mapping
CSL401.1	3	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
CSL401.2	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
CSL401.3	2	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

		2.2 Demonstrate an ability to formulate a solution plan and methodology for an engineering problem	2.2.4 Compare and contrast alternative solution/methods to select the best method.	PO1	1

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO10	PO11	PO12
CSL401.1	1	1										
CSL401.2	1	1										
CSL401.3	1	1										

	PSO1	PSO2
CSL401.1	--	--
CSL401.2	--	--
CSL401.3	--	--

CO Measurement Weightages for Tools:

Course Outcomes	***Direct Methods (80%)***				***Indirect Method (20%)***
	Lab Performance	Post Lab Questions	Quizzes	End Sem Exam	Course exit survey
CSL401.1	30%	10%	10%	50%	100%
CSL401.2	30%	10%	10%	50%	100%
CSL401.3	30%	10%	10%	50%	100%

Attainment:

CO1: CSL401.1:

- **Direct Method (80%)**

CSL 401.1dm= 0. 3*$Lab\ Performance$+0. 1*$Post\ Lab$+0. 1*$Quizzes$+0. 6*SEE_O/Pr

- **In Direct Methods (20%):** Course exit survey ***CSL401.1idm***
- ***Overall Attainment*** **CSL401.1 = 0.8* *CSL 401.1dm* + 0.2* *CSL401.1idm***

CO2: CSL401.2:

- **Direct Method (80%)**

CSL 401.1dm= 0. 3*$Lab\ Performance$+0. 1*$Post\ Lab$+0. 1*$Quizzes$+0. 6*SEE_O/Pr

- **In Direct Methods (20%):** Course exit survey ***CSL401.2idm***
- ***Overall Attainment*** **CSL401.2 = 0.8* *CSL 401.2dm* + 0.2* *CSL401.2idm***

CO3: CSL401.3:

- **Direct Method (80%)**

CSL 401.1dm= 0. 3*Lab Performance+0. 1*Post Lab+0. 1*Quizzes+0. 6*SEE_O/Pr

- **In Direct Methods (20%):** Course exit survey ***CSL401.3idm***

- ***Overall Attainment*** **CSL401.3 = 0.8* *CSL 401.3dm* + 0.2* *CSL401.3idm***

Practical Session Plan

Batch	***Dates***		***Remarks***
	Planned	***Actual***	
Experiment No. 1 WAP to implement Modified bubble sort, Insertion sort, Selection sort and derive its complexity.			
A	23/01/2023	23/01/2023	
D	25/01/2023	25/01/2023	
B	26/01/2023	02/02/2023	
C	27/02/2023	27/01/2023	
Experiment No. 2 WAP to implement Linear search and binary search and derive its time complexity.			
A	23/01/2023	23/01/2023	
D	25/01/2023	25/01/2023	
B	28/01/2023	28/01/2023	
C	2/02/2023	2/02/2023	
Experiment No. 3 WAP to implement Quick sort, randomized quick sort and derive its complexity			
A	30/01/2023	06/02/2023	
D	01/02/2023	06/02/2023	
B	02/02/2023	01/02/2023	
C	03/02/2023	04/02/2023	
Experiment No. 4 WAP to implement Merge sort and derive its complexity.			
A	30/01/2023	06/02/2023	
D	01/02/2023	06/02/2023	
B	02/02/2023	01/02/2023	
C	03/02/2023	04/02/2023	
Experiment No.5 WAP to implement the MinMax algorithm using greedy methods.			

A	13/02/2023	07/02/2023	
D	15/02/2023	08/02/2023	
B	16/02/2023	09/02/2023	
C	17/02/2023	11/02/2023	

Experiment No. 6
WAP to implement Fractional Knapsack using greedy methods.

A	20/02/2023	20/02/2023	
D	22/02/2023	15/02/2023	
B	23/02/2023	16/02/2023	
C	24/02/2023	25/02/2023	

Experiment No. 7
WAP to implement Dijkstra's Shortest Path algorithm using greedy methods.

A	06/03/2023	06/03/2023	
D	08/03/2023	22/03/2023	
B	09/03/2023	23/03/2023	
C	10/03/2023	10/03/2023	

Experiment No. 8
WAP to implement Prim's algorithm using greedy methods.

A	13/03/2023	13/03/2023	
D	15/03/2023	08/03/2023	
B	16/03/2023	09/03/2023	
C	17/03/2023	17/03/2023	

Experiment No. 9
WAP to implement 0-1 Knapsack using Dynamic Programming.

A	20/03/2023	20/03/2023	
D	22/03/2023	15/03/2023	
B	23/03/2023	16/03/2023	
C	24/03/2023	24/03/2023	

Experiment No. 10
WAP to implement Bellman Ford Algorithm using Dynamic Programming.

A	20/03/2023	20/03/2023	
D	22/03/2023	15/03/2023	
B	23/03/2023	16/03/2023	
C	24/03/2023	24/03/2023	

Experiment No. 11
WAP to implement Floyd Warshall Algorithm using Dynamic Programming

A	27/03/2023	27/03/2023	
D	29/03/2023	23/03/2023	
B	30/03/2023	29/03/2023	
C	31/03/2023	31/03/2023	
Experiment No. 12 WAP to implement N-Queen problem using back tracking approach.			
A	27/03/2023	03/04/2023	
D	29/03/2023	05/04/2023	
B	30/03/2023	06/04/2023	
C	31/03/2023	11/04/2023	
Experiment No. 13 WAP to implement Sum of Subsets problem using back tracking approach.			
A	03/04/2023	03/04/2023	
D	05/04/2023	05/04/2023	
B	06/04/2023	06/04/2023	
C	07/04/2023	11/04/2023	
Experiment No. 14 WAP to implement Graph coloring problem using back tracking approach.			
A	03/04/2023	03/04/2023	
D	05/04/2023	05/04/2023	
B	06/04/2023	06/04/2023	
C	07/04/2023	07/04/2023	
Experiment No. 15 WAP to implement Longest Common Subsequence using Dynamic Programming			
A	10/04/2023	10/04/2023	
D	12/04/2023	12/04/2023	
B	13/04/2023	13/04/2023	
C	14/04/2023	14/04/2023	

Verified by:

Programme Coordinator Subject Expert