



**B.E. DEGREE EXAMINATIONS: APRIL / MAY 2023**

(Regulation 2018)

Fourth Semester

**COMPUTER SCIENCE AND ENGINEERING**

U18CST4001: Design and Analysis of Algorithms

**COURSE OUTCOMES**

- CO1:** Compare various graph traversal techniques  
**CO2:** Apply algorithm analysis techniques for a given algorithms  
**CO3:** Examine algorithm design techniques for a given application  
**CO4:** Analyse different algorithms for solving a given problem  
**CO5:** Develop application using chosen algorithm technique

**Time: Three Hours**

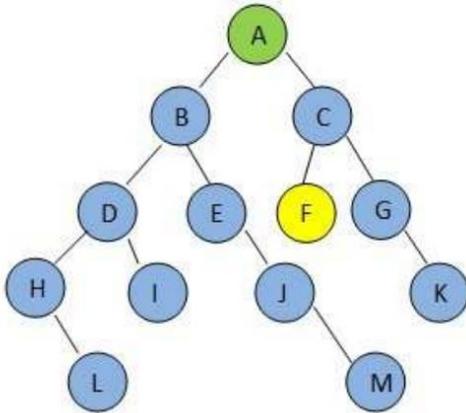
**Maximum Marks: 100**

**Answer all the Questions:-**

**PART A (10 x 2 = 20 Marks)**

**(Answer not more than 40 words)**

1. Consider the following graph, where A is the starting vertex and F is the goal vertex/node. CO1 [K<sub>3</sub>]



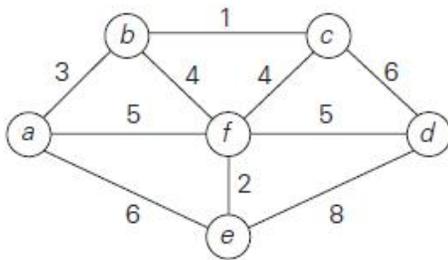
Make use of breadth first traversal and depth first traversal technique and find the order of traversal to reach goal node

2. What is topological sorting? In what kinds of graphs can topological sorting be used? CO1 [K<sub>2</sub>]
3. Find the order of growth for solutions of the following recurrences using Master Theorem CO2 [K<sub>3</sub>]
- $T(n) = 4T(n/2) + n^2, T(1) = 1$
4. Recall the basic asymptotic efficiency classes with example CO2 [K<sub>2</sub>]
5. Give the algorithm for sequential search and its best, worst, and average case complexity. CO3 [K<sub>2</sub>]

- |     |   |     |                   |
|-----|---|-----|-------------------|
| 6.  | Compute $29 * 13$ by applying the divide-and-conquer algorithm  | CO3 | [K <sub>2</sub> ] |
| 7.  | State the three requirements to be considered while constructing solution using greedy design technique | CO4 | [K <sub>2</sub> ] |
| 8.  | Enumerate the steps in finding a dynamic programming solution to a problem                              | CO4 | [K <sub>2</sub> ] |
| 9.  | State the Graph Coloring problem  | CO5 | [K <sub>2</sub> ] |
| 10. | Brief about Tractable and intractable problems  | CO5 | [K <sub>2</sub> ] |

**Answer any FIVE Questions:-  
PART B (5 x 16 = 80 Marks)  
(Answer not more than 400 words)**

- |     |    |  |   |     |                   |
|-----|----|--|---|-----|-------------------|
| 11. | a) | Explain the Dijkstra algorithm for finding single source shortest path and analyze its time and space complexity | 8 | CO1 | [K <sub>2</sub> ] |
|     | b) | Apply the Prim and Kruskal's algorithm used to identify the Minimum Spanning Tree (MST) of a graph.              | 8 | CO1 | [K <sub>3</sub> ] |



- |     |    |  |   |     |                   |
|-----|----|--|---|-----|-------------------|
| 12. | a) | Explain fundamentals of algorithmic problem solving in detail with suitable diagram.   | 8 | CO2 | [K <sub>2</sub> ] |
|     | b) | Formulate the recursive relation for Towers of Hanoi problem and find its complexity by solving the recurrence relation using backward substitution method.  | 8 | CO2 | [K <sub>3</sub> ] |
| 13. | a) | Write the Brute-force string matching algorithm and give its time complexity. Determine the number of character comparisons made by the brute-force algorithm in searching for the pattern GOAT in the text given below: | 8 | CO3 | [K <sub>3</sub> ] |

**THERE IS MORE TO LIFE THAN INCREASING ITS SPEED**

- b) Apply Strassen's matrix multiplication algorithm and compute the following. 8 CO3 [K<sub>3</sub>]

$$\begin{bmatrix} 1 & 0 & 2 & 1 \\ 4 & 1 & 1 & 0 \\ 0 & 1 & 3 & 0 \\ 5 & 0 & 2 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 & 1 \\ 2 & 1 & 0 & 4 \\ 2 & 0 & 1 & 1 \\ 1 & 3 & 5 & 0 \end{bmatrix}$$

Note: Exiting the recursion when n=2.

14. a) Solve the following Knapsack problem using Greedy technique. 8 CO4 [K<sub>3</sub>]

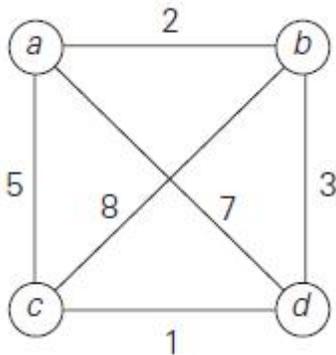
item	weight	value	
1	10	\$100	$W = 16$
2	7	\$63	
3	8	\$56	
4	4	\$12	

- b) Explain the dynamic programming algorithm for constructing an optimal search tree. 8 CO4 [K<sub>2</sub>]

15. a) Explain 4-Queens problem using backtracking . Illustrate state space tree and provide solution . 8 CO5 [K<sub>3</sub>]

- b) Write short notes on P, NP, NP-Complete and NP-Hard problems 8 CO5 [K<sub>2</sub>]

16. a) Apply branch-and-bound technique to solve the following TSP. 8 CO5 [K<sub>3</sub>]



b) Consider the following algorithm.

8 CO1 [K<sub>2</sub>]

ALGORITHM Enigma(A[0..n - 1, 0..n - 1])

//Input: A matrix A[0..n - 1, 0..n - 1] of real numbers

for i ← 0 to n - 2 do

for j ← i + 1 to n - 1 do

if A[i, j ] != A[j, i]

return false

return true

- a. What does this algorithm compute?
- b. What is its basic operation?
- c. How many times is the basic operation executed?
- d. What is the efficiency class of this algorithm?

\*\*\*\*\*