**SRM Institute of Science and Technology Set D**

**College of Engineering and Technology**

**School of Computing**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu

**Academic Year: 2021-22 (Even)**

**Test: CLA-T3**  **Date: 29-06-2022**

**Course Code & Title: 18CSC204J Design and Analysis of Algorithms** **Duration:** 2 Hour

**Year & Sem: II Year / IV Sem** **Max. Marks:** 50

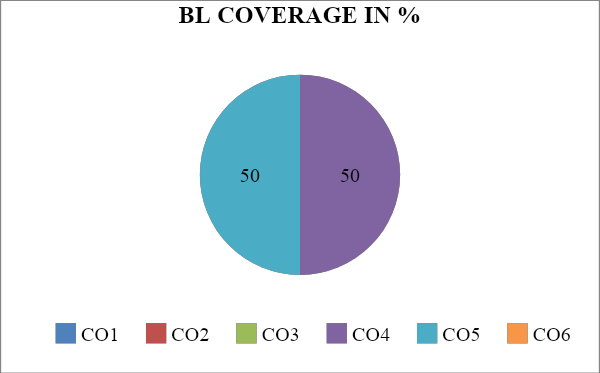
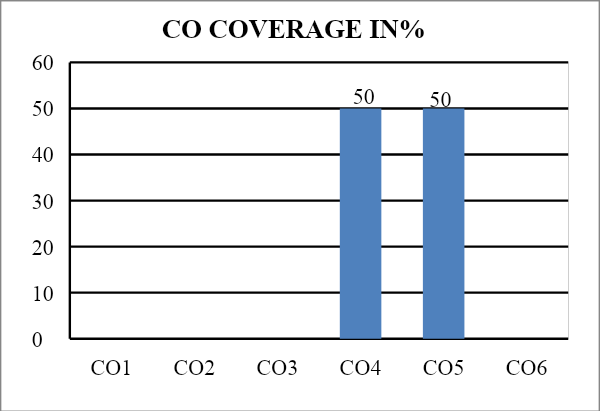
**Course Articulation Matrix:**

| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO1** | L | H |  | H | L |  |  |  | L | L |  | H |
| **CO2** | M | H | L | M | L |  |  |  | M | L |  | H |
| **CO3** | M | H | M | H | L |  |  |  | M | L |  | H |
| **CO4** | M | H | M | H | L |  |  |  | M | L |  | H |
| **CO5** | H | H | M | H | L |  |  |  | M | L |  | H |
| **CO6**. | L | H | M | H | L |  |  |  | L | L |  | H |

| **Part - A**  **(10 x 1 = 10 Marks)**  **Instructions: Answer all** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Q. No** | **Question** | **Marks** | **BL** | **CO** | **PO** | **PI Code** |
| **1** | Choose the searching method which is helpful in a backtracking algorithm for state-space tree construction.   1. **Depth-first search** 2. Breadth-first search 3. FIFO search 4. LIFO search | 1 | 3 | 4 | 3 | 2.1.1 |
| **2** | **Consider the sum-of-subset problem, n=4, sum=35, and weights={5,7,10,12,15,18,20}. How many total number of solutions for the given problem?**  **a) 4** b) 3 c) 2 d) 1 | 1 | 4 | 4 | 3 | 2.2.2 |
| **3** | A node is said to be \_\_\_\_\_\_\_\_\_\_\_\_ if it has a possibility of reaching a complete solution. a) Non-promising **b) Promising** c) Succeeding d) Preceding | 1 | 2 | 4 | 2 | 2.1.1 |
| **4** | Identify the correct choice from the following for branch and bound strategy:   1. converting the minimization problem into a maximization problem 2. **converting the maximization problem into a minimization problem** 3. changing the upper bound to lower bound 4. changing the lower bound to upper bound | 1 | 3 | 4 | 2 | 2.1.1 |
| **5** | In the Floyd Warshall Algorithm, the value of k is 0 in the following formula: A[i,j]=min{A[i,j],A[i,k],A[k,j]}. Give the correct meaning for the above statement. a) 1 intermediate vertex **b) 0 intermediate vertex** c) N intermediate vertices d) N-1 intermediate vertices | 1 | 4 | 4 | 3 | 2.1.1 |
| **6** | Randomized quick sort is identified by   1. Quick sort with random partitions 2. Quick sort with random output 3. **Quick sort with random choice of pivot** 4. Quick sort with random input | 1 | 3 | 5 | 3 | 2.1.2 |
| **7** | Find the matching time of Rabin Karp Algorithm, If the expected number of valid shifts is small and prime is larger than the length of pattern.  a Theta(m)  **b. Big-Oh(n+m)** **c.** Theta(n-m) **d.** Big-Oh(n) | 1 | 3 | 5 | 3 | 2.1.2 |
| **8** | A\_\_\_\_\_\_\_  of a [graph](https://en.wikipedia.org/wiki/Graph_(discrete_mathematics)) is a set of [vertices](https://en.wikipedia.org/wiki/Vertex_(graph_theory)) that includes at least one endpoint of every [edge](https://en.wikipedia.org/wiki/Edge_(graph_theory)) of the [graph](https://en.wikipedia.org/wiki/Graph_(discrete_mathematics)).  a. Vertex Traversal  b. Preorder Traversal  c. **Vertex cover**  d. In order cover | 1 | 2 | 5 | PO3 | 2.1.1 |
| **9** | Let X be a problem that belongs to the class NP. Then which one of the following is TRUE?.  a) There is no polynomial time algorithm for X  b) If X can be solved deterministically in polynomial time, then P=NP  c) **If X is NP-hard, then it is NP-complete**  d) X can be solved deterministically with decision | 1 | 2 | 5 | 1 | 2.2.2 |
| **10** | Let us assume the problem of 3-SAT and 2-SAT are belongs to the complexity classes. Choose the correct statement from the following:  a) both in P **b) NP-complete and P respectively** c) both NP-complete d) P-complete only | 1 | 3 | 5 | 1 | 3.1.1 |
| Part – B  ( 4 x 10 Marks = 40 Marks)  Instructions: Answer any 4 Questions | | | | | | |
| **11** | **Write the algorithm to solve the n-queen problem and Give the solution for the 4-queen problem with its state space tree.**  ALGORITHM:  1. Start in the leftmost column  2. If all queens are placed return true  3. Try all rows in the current column. Do the following for every tried row.  a. If the queen can be placed safely in this row then mark this [row, column] as part of the solution and recursively check if placing queen here leads to a solution.  b. If placing queen in [row, column] leads to a solution then return true.  c. If placing queen doesn't lead to a solution then unmark this [row, column] (Backtrack) and go to step (a) to try other rows.  4. If all rows have been tried and nothing worked, return false to trigger backtracking.  State space tree:    Solution: | 10 | 2 | 4 | 3 | 1.6.1 |
| **12** | **Consider the following graph G containing the Hamiltonian cycle of n vertices.**    **Draw the adjacency matrix of the graph G. Use a Backtracking algorithm to find the solution vertex. Compare its time complexity analysis with brute force method.**  The above graph contains the Hamiltonian cycles 1,2,3,4,5,6,7,8,1  1,3,4,5,6,7,8,2,1  1,2,8,7,6,5,4,3,1        Time complexity: T(n) = O(nn). | 10 | 3 | 4 | 4 | 2.5.3 |
| **13** | **Solve the Traveling Salesman Problem using Branch and Bound Algorithm in the following graph. Draw the graph G, the state space tree and find the optimal tour.**   |  | **A** | **B** | **C** | **D** | **E** | | --- | --- | --- | --- | --- | --- | | **A** | ∞ | 3 | 4 | 2 | 7 | | **B** | 3 | ∞ | 4 | 6 | 3 | | **C** | 4 | 4 | ∞ | 5 | 8 | | **D** | 2 | 6 | 5 | ∞ | 6 | | **E** | 7 | 3 | 8 | 6 | ∞ |   **Solution:**  **Graph:**    Answer:  Optimal path is: **A → D -> C → E → B → A**  Cost of Optimal path = **21 units** | 10 | 3 | 4 | 4 | 2.5.3 |
| **14** | **Explain in detail and compare the quicksort problem using divide and conquer and randomized algorithms with the suitable array elements.**  Solution  Divide and conquer:  -**use first, last or mid array element as pivot element**  Partitioning algorithm:  PARTITION(A, p, q) ⊳ A[p . . q] x ← A[p] ⊳ pivot = A[p] i ← p for j ← p + 1 to q do if A[j] ≤ x then i ← i + 1 exchange A[i] ↔ A[j] exchange A[p] ↔ A[i] return i ≤ x  QUICKSORT(A, p, r) if p < r then q ← PARTITION(A, p, r) QUICKSORT(A, p, q) QUICKSORT(A, q+1, r)  Best-case analysis-PARTITION splits the array evenly: T(n) = 2T(n/2) + Θ(n) = Θ(n lg n)  Randomized quicksort IDEA:   * **Use a hash function to select the pivot element**   Partition around a random element.  • Running time is independent of the input order.  • No assumptions need to be made about the input distribution.  • No specific input elicits the worst-case behavior.  • The worst case is determined only by the output of a random-number generator.  Randomized: | 10 | 3 | 5 | 5 | 2.6.3 |
| **15** | Write short note on the following:   * + - 1. Non-deterministic Polynomial class problems with NP-hard and NP-complete       2. Satisfiability problems with example | 10 | 3 | 5 | 3 | 2.6.3 |

**\*Program Indicators are available separately for Computer Science and Engineering in AICTE examination reforms policy.**

**Course Outcome (CO) and Bloom’s level (BL) Coverage in Questions**



**Approved by the Audit Professor/Course Coordinator**