Reg. No. \_\_\_\_\_\_\_\_\_\_\_\_\_



**End Semester Examination – Nov / Dec – 2019**

|  |  |  |  |
| --- | --- | --- | --- |
| **Code :** | **14CS2001** | **Duration :** | **3hrs** |
| **Sub. Name :** | **ANALYSIS OF ALGORITHMS** | **Max. Marks :** | **100** |

**ANSWER ALL QUESTIONS (5 x 20 = 100 Marks)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Q. No.** | **Sub Div.** | **Questions** | **Course**  **Outcome** | **Marks** |
| 1. | a. | Solve the following recurrence relations:   1. x(n)=x(n-1)+5 for n>1, x(1)=0. 2. x(n)=x(n/2)+n for n>1, x(1)=1(solve for n=2k ) | CO1 | 15 |
| b. | Construct an algorithm for element uniqueness problem and assess its worst case time complexity. | CO1 | 5 |
| **(OR)** | | | | |
| 2. | a. | Design string matching algorithm using Brute-Force method and analyze its efficiency. | CO1 | 10 |
| b. | Explain asymptotic notations with suitable examples for each notation. | CO1 | 10 |
|  |  |  |  |  |
| 3. |  | Demonstrate quick sort algorithm with suitable example. Estimate it’s worst-case, best case and average case time complexity. | CO2 | 20 |
| **(OR)** | | | | |
| 4. | a. | Illustrate Bubble sort algorithm with example and examine it’s time efficiency for best and worst case inputs. | CO2 | 10 |
| b. | Illustrate insertion sort algorithm with example and examine it’s time efficiency for best and worst case inputs. | CO2 | 10 |
|  |  |  |  |  |
| 5. | a. | Explain how dynamic programming is applied to solve travelling  sales person problem. | CO2 | 10 |
| b. | Apply Kruskal’s algorithm to find a minimum spanning tree of the graph. | CO2 | 10 |
| **(OR)** | | | | |
| 6. |  | Solve the following instance of 0/1 Knapsack problem using dynamic programming technique:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Item** | 1 | 2 | 3 | 4 | | **Value** | 100 | 20 | 60 | 40 | | **Weight** | 3 | 2 | 4 | 1 | | CO2 | 20 |
|  |  |  |  |  |
| 7. | a. | Find an optimal binary merge pattern for merging ten sorted files whose lengths are 28, 32, 12, 5, 84, 53, 91, 35, 3 and 11. Also, compute the minimum number of moves required. | CO2 | 10 |
| b. | Explain warshall’s algorithm to find the transitive closure of a digraph. Give example. | CO2 | 10 |
| **(OR)** | | | | |
| 8. | a. | Construct Huffman tree for the following data and obtain its Huffman code:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Symbol | A | B | C | D | E | | Frequency | 24 | 12 | 10 | 8 | 8 |   Decode the text whose encoding is 10011110101110. | CO3 | 10 |
| b. | How will you find the shortest path between two given vertices using Dijkstra’s algorithm? Explain with example. | CO3 | 10 |
|  | | **Compulsory**: |  |  |
| 9. | a. | Apply Horspool’s algorithm to an example pattern and demonstrate by constructing a shift table for it. | CO3 | 10 |
| b. | Apply Backtracking algorithm to solve 4-queen’s problem. | CO3 | 10 |