|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Roll No |  |  |  |  |  |  |  |  |  |  |  |  |

****

**Presidency University**

**Bengaluru**

**School Of Computer Science and Engineering & Information Science**

**Summer-Term End Term Examinations, Aug 2024**

**Date**: 07.08.2024

**Time**: 01.00 PM-04.00 PM

**Max Marks**: 100

**Weightage**: 50%

**Odd Semester**: 2023 - 24

**Course Code**: CSE 2007

**Course Name**: Design and analysis of algorithms

**Department: SOCSE**

**Instructions:**

1. *Read the all questions carefully and answer accordingly.*
2. *Do not write any matter on the question paper other than roll number.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Q.No** | **Questions** | **Marks** | **CO** | **RBT** |
| 1 | 1. Describe Notion of Algorithm | 4 | CO1 | L1 |
| 1. List down the steps involved in mathematical analysis of Non-Recursive Algorithms | 6 | CO1 | L2 |
| 1. Identify the time complexity (upper bound) for the below iterative functions   A()  {  Int i,j,k,n;  for(i=1;i<=n;i++)  {  for(j=1;j<=i;j++)  {  for(k=1;k<=100;k++)  {  Printf(“Ravi”);  }  }  } | 10 | CO1 | L3 |
| OR | | | | |
| 2 | 1. List down the steps involved in analyzing an algorithm | 4 | CO1 | L1 |
| 1. Explain with an example how a new variable count introduced in a program can be used to find the number of steps needed by a program to solve a problem instance. | 6 | CO1 | L2 |
| 1. Find the time complexity (upper bound) for the below recursive functions   T(n)=n + T(n-1); ;n>1  T(n)=1 ;n=1 | 10 | CO1 | L3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3 | 1. In brief explain brute force strategy of programming | 4 | CO2 | L1 |
| 1. Briefly explain Traveling Salesman Problem (TSP) using brute force strategy with example | 6 | CO2 | L2 |
| 1. List down the steps for linear search and mention its best case, worst case and average case | 10 | CO2 | L3 |

OR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1. Write an algorithm to find uniqueness of elements in an array | 4 | CO2 | L1 |
| 1. Demonstrate pattern matching algorithm with suitable example | 6 | CO2 | L2 |
| 1. List down the steps involved for bubble sort and apply the same to sort 9, 8,7, 6, 5, 4, 3, 2, 1 | 10 | CO2 | L3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 5 | 1. In brief explain Divide & conquer strategy of programming | 4 | CO3 | L1 |
| 1. Explain the general divide & conquer recurrence relation | 6 | CO3 | L2 |
| 1. State master theorem and apply the same for recurrence relation   T(n) = 2T(n/2) + 1 | 10 | CO3 | L3 |

OR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 6 | 1. List down the advantages and limitations of divide & conquer technique | 4 | CO3 | L1 |
| 1. Write and explain binary search algorithm with an example | 6 | CO3 | L2 |
| 1. Write and explain quick sort algorithm with an example. | 10 | CO3 | L3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 7 | 1. Define Dynamic programming and briefly list down its properties | 4 | CO4 | L1 |
| 1. Briefly explain steps in involved in Floyd algorithm with steps in involved in it. | 6 | CO4 | L2 |
| Briefly explain Bellman ford algorithm and Why Relaxing Edges N-1 times, gives us Single Source Shortest Path? | 10 | CO4 | L3 |

OR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 1. Bring out at least three differences between divide & conquer and dynamic programming | 4 | CO4 | L1 |
| 1. List down the applications of the greedy strategy | 6 | CO4 | L2 |
| Apply all pair shortest path algorithm (Floyd) for the below graphhttps://www.gatevidyalay.com/wp-content/uploads/2018/07/Floyd-Warshall-Algorithm-Problem-01.png | 10 | CO4 | L3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 | 1. How does backtracking algorithm work? | 4 | CO1 | L1 |
| 1. Find the minimum spanning tree (MST) by applying prims algorithm with B as source vertex.   Prim's Algorithm | 6 | CO1 | L2 |
| 1. Draw state space tree for N queens problem with 4 \*4 chess board having 4 queens Q1,Q2,Q3,Q4. | 10 | CO1 | L3 |

OR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 10 | 1. List down the steps involved in back tracking | 4 | CO2 | L1 |
| 1. Define minimum spanning tree (MST) and explain working principle of Prims algorithm. | 6 | CO2 | L2 |
| 1. How backtracking approach is used to solve sum of subset problem | 10 | CO2 | L3 |