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

**College of Engineering and Technology**

**School of Computing**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu

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

**Test: CLA-T1** **Date: 05-04-2022**

**Course Code & Title: 18CSC204J Design and Analysis of Algorithms** **Duration:** 1 Hour

**Year & Sem: II Year / IV Sem** **Max. Marks:** 25

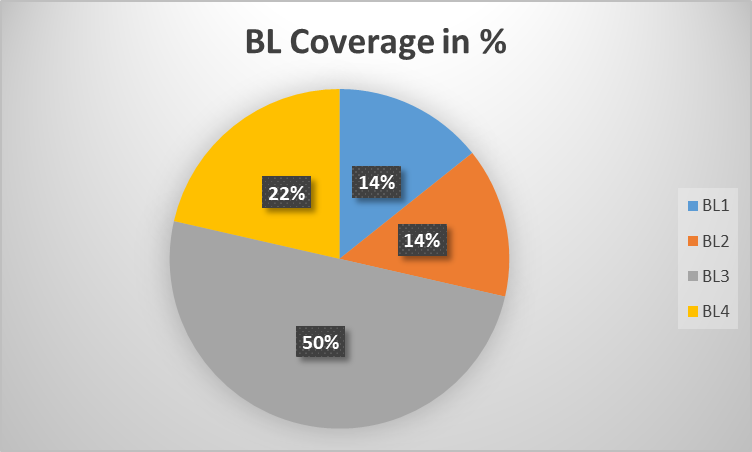
**Course Articulation Matrix:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | There are four algorithms A1, A2, A3, A4 to solve the given problem with the order log(n), nlog(n),n2, 2n respectively. Which is the best algorithm?  A. **A1**  B. A2  C. A3  D. A4 | **1** | **L1** | **1** | **2** | **2.6.4** |
| **2** | How many comparisons are required in insertion sort to sort the data of size ‘n’ if the data is sorted in reverse order?   1. n 2. **n(n-1)/2** 3. n-1 4. n/2 | **1** | **L4** | **1** | **2** | **2.6.5** |
| **3** | Given t(n) <= c.g(n) for all n >= n0, c is a positive constant, n0 is non-negative integer then   1. **t(n) is said to be in O(g(n))** 2. t(n) is said to be in Ɵ(g(n)) 3. t(n) is said to be in Ω(g(n)) 4. t(n) is said to be in O (t(n)) | **1** | **L1** | **1** | **1** | **1.7.1** |
| **4** | The function T(n) = 3n2 + 2n +3 is   1. **O(n2)** 2. O(n) 3. O(log n) 4. O(n log n) | **1** | **L3** | **1** | **2** | **2.5.3** |
| **5** | For a linear search in an array of N elements the time complexity for best, worst and average case are   1. O(n), O(1), O(n) 2. **O(1), O(n), O(n)** 3. O(n),O(1),O(n) 4. O(n),O(n),O(1) | **1** | **L4** | **1** | **1** | **1.7.1** |
| **6** | Which among the following contributes to linear growth?   1. **3n** 2. 3n2 3. 2n 4. (3/2)n | **1** | **L3** | **1** | **2** | **2.6.4** |
| **7** | Find the recurrence equation for the sequence 9,27/4,81/16,243/64,…? Given T(0)= 9   1. T(n) = T(n-1) \* (4/3) 2. T(n) = T(n-1) \* (2/3) 3. **T(n)= T(n-1) \* (3/4)** 4. T(n) = 9 \* T(n-1) | **1** | **L3** | **1** | **2** | **2.8.1** |
| **8** | The time factor when determining the efficiency of algorithm is measured by   1. Counting microseconds 2. **Counting the number of operations** 3. Counting the number of statements 4. Counting the memory required | **1** | **L2** | **1** | **1** | **1.6.1** |
| **9** | Which is the correct order of growth from slowest to fastest?(**less to high time taken)**   1. logn,nlogn,n3,n2 2. n3 , n2 ,2n, n! 3. **n,nlogn,n2,2n** 4. 1,n,logn,n2 | **1** | **L4** | **1** | **2** | **2.6.4** |
| **10** | sum = 0;  for( i = 0; i < n; i++)  for( j = 0; j < i; j++)  sum++;  The running time of the above code is   1. **O(n2)** 2. O(n/2) 3. O(3n) 4. O(nlogn) | **1** | **L3** | **1** | **2** | **2.5.2** |
| **Part – B**  **( 3 x 5 Marks = 15 Marks)**  **Instructions: Answer any 3 Questions** | | | | | | |
| **11** | Explain different design strategies for designing an algorithm.  Brief Brute Force, Divide and Conquer, Greedy Algorithm, Dynamic Programming, Backtracking, Branch and Bound | **5** | **L2** | **1** | **1** | **1.6.1** |
| **12** | Prove using substitution method that if  T(n) = 2T(n/2) + cn then T(n) = O(nlogn)  O(nlogn) | **5** | **L3** | **1** | **2** | **2.5.3** |
| **13** | Solve the following recurrence relation using recursion tree method.  T(n) = 3T(n/4) + cn2, c>0  O(n2) | **5** | **L3** | **1** | **2** | **2.5.3** |
| **14** | Derive worst case analysis of insertion sort using step count.  Pseudocode of insertion sort – 1  Step count – 2  Worst case analysis proof - 2 | **5** | **L3** | **1** | **2** | **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**