SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Ramapuram Campus, Bharathi Salai, Ramapuram, Chennai - 600089

**FACULTY OF ENGINEERING AND TECHNOLOGY**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

****

**QUESTION BANK**

**DEGREE / BRANCH: B.Tech/CSE**

**IV SEMESTER**

**18CSC204J – Design and Analysis of Algorithms**

**Regulation – 2018**

**Academic Year 2021-2022**

# SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

**Ramapuram Campus, Bharathi Salai, Ramapuram, Chennai-600089**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**QUESTION BANK**

**SUBJECT : 18CSC204J – Design and Analysis of Algorithms**

**SEM/ YEAR: IV/ II**

**Course Outcomes**

**CO1:** Apply efficient algorithms to reduce space and time complexity of both recurrent and non-recurrent relations

**CO2:** Solve problems using divide and conquer approaches

**CO3:** Apply greedy and dynamic programming types techniques to solve polynomial time problems.

**CO4:** Create exponential problems using backtracking and branch and bound approaches.

**CO5:** Interpret various approximation algorithms and interpret solutions to evaluate P type, NP Type, NPC, NP Hard problems

**CO6:** Create algorithms that are efficient in space and time complexities by using divide conquer, greedy, backtracking technique

| **UNIT II** | | | |
| --- | --- | --- | --- |
| Introduction-Divide and Conquer-Maximum Subarray Problem-Binary Search-Complexity of binary search  Merge sort-Time complexity analysis-Quick sort and its Time complexity analysis-Best case, Worst case, Average case analysis-Strassen's Matrix multiplication and its recurrence relation-Time complexity analysis of Merge sort-Largest sub-array sum-Time complexity analysis of Largest sub-array sum-Master Theorem Proof  Master theorem examples - Finding Maximum and Minimum in an array - Time complexity analysis-Examples  Algorithm for finding closest pair problem - Convex Hull problem | | | |
| **PART-A (Multiple Choice Questions)** | | | |
| **Q.**  **No** | **Questions** | **Course Outcome** | **Competence**  **BT Level** |
| **1** | Partition and exchange sort is\_\_\_\_\_\_\_\_\_\_\_\_  A. quick sort  B. tree sort  C. heap sort  D. bubble sort  **ANSWER: A** | CO 2 | BT 1 |
| **2** | Which of the following is not the required condition for binary search algorithm?  A. The list must be sorted  B. There should be the direct access to the middle element in any sub list  C. There must be mechanism to delete and/or insert elements in list.  D. Number values should only be present  **ANSWER: C** | CO 2 | BT 1 |
| **3** | Which of the following sorting algorithm is of divide and conquer type?  A. Bubble sort  B. Insertion sort  C. Merge sort  D. Selection sort  **ANSWER: C** | CO 2 | BT 1 |
| **4** | \_\_\_\_\_\_\_\_\_\_\_\_order is the best possible for array sorting algorithm which sorts n item.  A. O (n logn)  B. O (n2)  C. O (n+logn)  D. O (logn)  **ANSWER: C** | CO 2 | BT 1 |
| **5** | The complexity of merge sort algorithm is \_\_\_\_\_\_\_\_\_\_  A. O (n)  B. O (logn)  C. O (n2)  D. O (n logn)  **ANSWER: D** | CO 2 | BT 1 |
| **6** | Binary search algorithm cannot be applied to \_\_\_\_\_\_\_\_  A. sorted linked list  B. sorted binary trees  C. sorted linear array  D. pointer array  **ANSWER: A** | CO 2 | BT 1 |
| **7** | Which of the following is not a limitation of binary search algorithm?  A. must use a sorted array  B. requirement of sorted array is expensive when a lot of insertion and deletions are needed  C. there must be a mechanism to access middle element directly  D. binary search algorithm is not efficient when the data elements more than 1500.  **ANSWER: D** | CO 2 | BT 1 |
| **8** | Which of the following is an external sorting? A. Insertion Sort B. Bubble Sort C. Merge Sort D. Tree Sort  **ANSWER: B** | CO 2 | BT 1 |
| **9** | Merging k sorted tables into a single sorted table is called \_\_\_\_\_\_\_ A. k way merging B. k th merge C. k+1 merge D. k-1 merge  **ANSWER: A** | CO 2 | BT 1 |
| **10** | The operation that combines the element is of A and B in a single sorted list C with n=r+s element is called \_\_\_\_\_\_\_\_ A. Inserting B. Mixing C. Merging D. Sharing  **ANSWER: C** | CO 2 | BT 1 |
| **11** | What is the worst case time complexity of a quick sort algorithm? a) O(N) b) O(N log N) c) O(N2) d) O(log N)  **ANSWER: C** | CO 2 | BT 1 |
| **12** | What is the average running time of a quick sort algorithm? a) O(N2) b) O(N) c) O(N log N) d) O(log N)  **ANSWER: C** | CO 2 | BT 1 |
| **13** | How many sub arrays does the quick sort algorithm divide the entire array into? a) one b) two c) three d) four  **ANSWER: B** | CO 2 | BT 1 |
| **14** | Quick sort is a \_\_\_\_\_\_\_\_\_\_ a) greedy algorithm b) divide and conquer algorithm c) dynamic programming algorithm d) backtracking algorithm  **ANSWER: B** | CO 2 | BT 1 |
| **15** | Which one of the following sorting algorithm is best suited to sort an array of 1 million elements? a) Bubble sort b) Insertion sort c) Merge sort d) Quick sort  **ANSWER: D** | CO 2 | BT 1 |
| **16** | Strassen’s algorithm is a/an\_\_\_\_\_\_\_\_\_\_\_\_\_ algorithm. a) Non- recursive b) Recursive c) Approximation d) Accurate  **ANSWER: B** | CO 2 | BT 1 |
| **17** | What is the running time of naïve matrix multiplication algorithm? a) O(n2.81) b) O(n4) c) O(n) d) O(n3)  **ANSWER: D** | CO 2 | BT 1 |
| **18** | \_\_\_\_\_\_\_\_\_\_\_ is a method of constructing a smallest polygon out of n given points. a) closest pair problem b) quick hull problem c) path compression d) union-by-rank  **ANSWER: B** | CO 2 | BT 1 |
| **19** | What is the other name for quick hull problem? a) convex hull b) concave hull c) closest pair d) path compression  **ANSWER: A** | CO 2 | BT 1 |
| **20** | Master’s theorem is used for? a) solving recurrences b) solving iterative relations c) analysing loops d) calculating the time complexity of any code  **ANSWER: A** | CO 2 | BT 1 |
| **21** | What is the result of the recurrences which fall under first case of Master’s theorem (let the recurrence be given by T(n)=aT(n/b)+f(n) and f(n)=nc? a) T(n) = O(n^logba) b) T(n) = O(nc log n) c) T(n) = O(f(n)) d) T(n) = O(n2)  **ANSWER: A** | CO 2 | BT 1 |
| **22** | Under what case of Master’s theorem will the recurrence relation of merge sort fall? a) 1 b) 2 c) 3 d) It cannot be solved using master’s theorem  **ANSWER: B** | CO 2 | BT 1 |
| **23** | Find the maximum sub-array sum for the given elements. {2, -1, 3, -4, 1, -2, -1, 5, -4} a) 3 b) 5 c) 8 d) 6  **ANSWER: B** | CO 2 | BT 1 |
| **24** | What is the time complexity of the divide and conquer algorithm used to find the maximum sub-array sum? a) O(n) b) O(logn) c) O(nlogn) d) O(n2)  **ANSWER: C** | CO 2 | BT 1 |
| **25** | What is the advantage of recursive approach than an iterative approach? a) Consumes less memory b) Less code and easy to implement c) Consumes more memory d) More code has to be written  **ANSWER: B** | CO 2 | BT 1 |
| **PART B (4 Marks)** | | | |
| **1** | Formulate the Binary Search Algorithm and analyze its worst, best and average case time complexity. | CO2 | BT 3 |
| **2** | Consider that there were two list of sorted array. Propose the algorithm for combining the elements of two arrays into single array in sorted order. | CO2 | BT 3 |
| **3** | Derive the algorithm for Quick sort and comment on the time complexity of the same. | CO2 | BT 3 |
| **4** | State the objective of Strassen Matrix Multiplication and list the steps involved in the process. | CO2 | BT 3 |
| **5** | Describe the control abstraction for Divide and Conquer strategy. | CO2 | BT 2 |
| **6** | List and describe the approaches involved in the Largest Sub Array problem solving process. | CO2 | BT 2 |
| **7** | Write short notes on Master theorem usage in the time complexity analysis. | CO2 | BT 2 |
| **8** | Develop the algorithmic steps to find the maximum and minimum element in the given list. | CO2 | BT 3 |
| **PART C (12 Marks)** | | | |
| **1** | Consider example of your choice and find the Closest pair among the considered number. Devise an algorithm and analyze its efficiency. | CO2 | BT 4 |
| **2** | Explain the Convex Hull problem and analyze the time complexity of the algorithm. | CO2 | BT 4 |
| **3** | Design the recursive and iterative binary search algorithm. And analyze the time complexity of the two approaches. | CO2 | BT 4 |
| **4** | Devise an algorithm for merging the two sorted sub arrays using auxiliary storage and conclude the time complexity of the algorithm. | CO2 | BT 4 |
| **5** | Derive the time complexity of the Stassen Matrix multiplication process using recurrence relation. | CO2 | BT 4 |

**Note:**

1. **BT Level –** Blooms Taxonomy Level
2. **CO – Course Outcomes**

BT1 – Remember BT2 – Understand BT3 – Apply BT4 – Analyze BT5 – Evaluate BT6 – Create