INSTITUTE OF AERONAUTICAL ENGINEERING

## (Autonomous)

Dundigal, Hyderabad - 500043

## INFORMATION TECHNOLOGY

DEFINITIONS AND TERMINOLOGY QUESTION BANK

| Course Name | $:$ | DESIGN AND ANALYSIS OF ALGORITHMS |
| :--- | :--- | :--- |
| Course Code | $:$ | AITB05 |
| Program | $:$ | B.Tech |
| Semester | $:$ | IV |
| Branch | $:$ | Information Technology |
| Section | $:$ | A $\mid$ B |
| Academic Year | $:$ | 2019 - 2020 |
| Course Faculty | $:$ | Dr.M Purushotham Reddy, Associate Professor |

## COURSE OBJECTIVES:

The course should enable the students to:
I Assess how the choice of data structures and algorithm design methods impacts the performance of programs.
II Solve problems using data structures such as binary search trees, and graphs and writing programs for these solutions.
III Choose the appropriate data structure and algorithm design method for a specified application.
IV Solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking, and branch and bound and writing programs for these solutions.

## DEFINITIONS AND TERMINOLOGY QUESTION BANK

| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| MODULE-I |  |  |  |  |  |  |
| 1 | What is meant algorithm? | An algorithm is set of instructions that if followed accomplishes a particular task. | Remember | CO1 | CLO1 | AITB05.01 |
| 2 | List the characteristics that any algorithm satisfies. | - Input <br> - Output <br> - Definiteness <br> - Finiteness <br> - Effectiveness | Remember | CO1 | CLO1 | AITB05.01 |
| 3 | What are the types of algorithm? | - Recursive algorithms. <br> - Dynamic programming algorithm. <br> - Backtracking algorithm. <br> - Divide and conquer algorithm. <br> - Greedy algorithm. <br> - Brute Force algorithm. | Remember | CO1 | CLO1 | AITB05.01 |


| S.No | QUESTION | ANSWER Rloms | CO | CLO | CLO Code |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  |  | Level |  |  |  |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | size grows. It represents worst case time complexity. |  |  |  |  |
| 12 | What is Omega notation? | Omega tells us the lower bound of the runtime of a function. | Remember | CO1 | CLO3 | AITB05.03 |
| 13 | What does asymptotic mean? | The definition of asymptotic is a line that approaches a curve but never touches. | Understand | CO1 | CLO3 | AITB05.03 |
| 14 | What is performance analysis of algorithm? | Analysis of algorithm is the process of analyzing the problem-solving capability of the algorithm in terms of the time and size required (the size of memory for storage while implementation). | Remember | CO1 | CLO2 | AITB05.02 |
| 15 | What is the time complexity of algorithm? | The amount of time taken by a set of code or algorithm to process or run as a function of the amount of input. | Remember | CO1 | CLO2 | AITB05.02 |
| 16 | What do you mean by space complexity of an algorithm? | Space complexity of an algorithm quantifies the amount of space or memory taken by an algorithm to run as a function of the length of the input. | Remember | CO1 | CLO2 | AITB05.02 |
| 17 | What <br> amortized <br> algorithm? is <br> cost | The amount we charge an operation is called its amortized cost. | Remember | CO1 | CLO3 | AITB05.03 |
| 18 | What is the basic principle of divide and conquer? | A divide-and-conquer algorithm works by recursively breaking down a problem into two or more sub-problems of the same or related type, until these become simple enough to be solved directly. The solutions to the subproblems are then combined to give a solution to the original problem. | Understand | $\mathrm{CO} 1$ | CLO4 | AITB05.04 |
| 19 | What are the advantages of divide and conquer? | With the divide and conquer method, it reduces the degree of difficulty since it divides the problem into sub problems that are easily solvable, and usually runs faster than other algorithms would. It also uses memory caches effectively. | Understand | CO1 | CLO4 | AITB05.04 |
| 20 | What is binary search and its algorithm? | Binary search is an efficient algorithm for finding an item from a sorted list of items. It works by repeatedly dividing in half the portion of the list that could contain the item, until you've narrowed down the possible locations to just one. | Understand | CO1 | CLO4 | AITB05.04 |
| 21 | What is time complexity of binary search? | $\mathrm{O}(\log \mathrm{n})$ | Understand | CO1 | CLO4 | AITB05.04 |
| 22 | What is the <br> algorithm for <br> quick sort?  | The algorithm starts by picking a single item which is called pivot and moving all smaller items | Remember | CO1 | CLO4 | AITB05.04 |


| S.No | QUESTION | ANSWER | Blooms <br> Level | CO | CLO | CLO Code |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  |  | before it, while all greater <br> elements in the later portion of <br> the list. |  |  |  |  |
| 23 | What is time <br> complexity of <br> quick sort? | O(n log n) | Understand | CO1 | CLO4 | AITB05.04 |
| 24 | What is the <br> technique used in <br> merge sort <br> algorithm? | Divide and Conquer. | Understand | CO1 | CLO5 | AITB05.05 |
| 25 | What is the <br> complexity of <br> merge sort? | O(nlogn) | Understand | CO1 | CLO5 | AITB05.05 |
| 26 | What is <br> Strassen's matrix <br> multiplication? | Strassen's matrix is a Divide and <br> Conquer method that helps us <br> to multiply two matrices(of size n <br> Xn). | Remember | CO1 | CLO5 | AITB05.05 |
| 27 | What is the time <br> complexity of <br> Strassen's matrix <br> multiplication? | O(n ${ }^{2.80}$ ) | Understand | CO1 | CLO5 | AITB05.05 |

## MODULE-II

| 1 | What is <br> set? | A set is a collection of distinct <br> elements. The Se can be <br> represented,forexamples, <br> asS1=\{1,2,5,10\}. | Understand | CO 2 | CLO6 | AITB05. |
| :---: | :--- | :--- | :--- | :--- | :--- | :---: |
| 2 | Define <br> Traversal? | Traversal of a binary tree <br> involves examining every node <br> in the tree. | Remember | CO 2 | CLO6 | AITB05. |
| 3 | Define Search? | Search involves visiting nodes <br> in a graph in a systematic <br> manner, and may or may not <br> result into a visit to all nodes. | Remember | CO 2 | CLO7 | AITB05. |
| 4 | What is disjoint <br> set? | A disjoint-set data structure is <br> a data structure that keeps <br> track of a set of elements <br> partitioned into a number <br> of disjoint (non-overlapping) <br> subsets. | Remember | CO 2 | CLO 7 | AITB05. |
| 5 | What are <br> disjoint set <br> operations? | The disjoint set operations are <br> 1. Union <br> 2. Find | Remember | CO 2 | CLO 8 | AITB05. |
| 6 | Define <br> Disjoint <br> setUnion | If Si and Sj are two disjoint <br> sets, then their union Si U Sj <br> consists of all the elements x <br> such that x is in Si or Sj. | Remember | CO 2 | CLO 8 | AITB05. |
| 7 | DefineDisjoint <br> Union | To perform disjoint set union <br> between two sets Si and Sj <br> can take any one root and <br> make it sub-tree of the other. | Remember | CO 2 | CLO 8 | AITB05. |
| 8 | What is find <br> operation? | To perform find operation, <br> along with the tree structure <br> we need to maintainthe name <br> of each set. So, we require <br> one more data structure to <br> store the set names. The data <br> structure contains two fields. | Understand | CO 2 | CLO 9 | AITB05. |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | One is the set name and the other one is the pointer to root. |  |  |  |  |
| 9 | Define unionfind algorithm? | A union-find algorithm is an algorithm that performs two useful operations on such a data structure: Find: Determine which subset a particular element is in. This can be used for determining if two elements are in the same subset. | Remember | CO 2 | CLO 9 | AITB05 |
| 10 | Describe Weighting rule forUnion? | If the number of nodes in the tree with root $I$ is less than the number in the tree with the root $j$, then make ' j ' the parent of i ; otherwise make ' i ' the parent of j . | Understa nd | CO 2 | CLO 9 | AITB05. |
| 11 | What are Minimum Spanning Trees? | A spanning tree for a connected graph is a tree whose vertex set is the same as the vertex set of the given graph, and whose edge set is a subset of the edge set of the given graph. i.e., any connected graph will have a spanning tree. | Understa nd | CO 2 | CLO 10 | AITB05. |
| 12 | Define BFST? | Spanning trees obtained using BFS then it called Breadth First Spanning Trees(BFST) | Remember | CO 2 | CLO 10 | AITB05. |
| 13 | What is Depth First Search | It involves exhaustive searches of all the nodes by going ahead, if possible, else by backtracking. Here, the word backtrack means that when you are moving forward and there are no more nodes along the current path, you move backwards on the same path to find nodes to traverse |  | CO 2 | $\text { CLO } 10$ | AITB05. |
| 14 | What is Graph Traversal? | Graph traversal (also known as graph search) refers to the process of visiting (checking and/or updating) each vertex in a graph.Such traversals are classified by the order in which the vertices are visited. | Understand | CO 2 | CLO 3 | AITB05. |
| 15 | What are the traversal techniques? | Preorder traversal, Inorder traversal, Postorder traversal. | Understand | CO 2 | CLO 4 | AITB05. |
| MODULE-III |  |  |  |  |  |  |
| 1 | Define a 'Greedy algorithm'? | A greedy algorithm is used to construct a Huffman tree during Huffman coding where it finds an optimal solution. | Remember | CO 3 | CLO 11 | AITB05.11 |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 2 | List applications of greedy method. | 1. Dijkstra's algorithm <br> 2. Prims algorithm <br> 3. Huffman tress | Remember | CO 3 | CLO 11 | AITB05.11 |
| 3 | Define Huffman tree? | Huffman tree is a full binary tree in which each leaf of the tree corresponds to a letter in the given alphabet. | Remember | CO 3 | CLO 11 | AITB05.11 |
| 4 | Define Job sequencing problem | A Simple Solution is to generate all subsets of given set of jobs and check individual subset for feasibility of jobs in that subset. | Remember | CO 3 | $\begin{gathered} \text { CLO } \\ 11 \end{gathered}$ | AITB05.11 |
| 5 | Define minimum cost spanning tree | A spanning tree with weight less than or equal to the weight of every other spanning tree. | Remember | CO 3 | CLO11 | AITB05.11 |
| 6 | Define optimal solution. | An optimal solution is a feasible solution where the objective function reaches its maximum (or minimum) value | Remember | CO 3 | CLO12 | AITB05.12 |
| 7 | List out the applications of greedy method. | Job Sequenced with deadline Knapsack problem Huffman coding | Remember | CO 3 | CLO12 | AITB05.12 |
| 8 | Define knapsack problem | The knapsack problem is an optimization problem used to illustrate both problem and solution. | Remember | CO 3 | CLO12 | AITB05.12 |
| 9 | Define subset paradigm | At each stage a decision is made whether a particular input is in the optimal solution. This is called subset paradigm. | Remember | CO 3 | CLO13 | AITB05.13 |
| 10 | State single source shortest path problem | The single-source shortest path problem, in which we have to find shortest paths from a source vertex v to all other vertices in the graph. | Remember | CO 3 | CLO13 | AITB05.13 |
| 1 | What is a spanning tree | A subgraph of G that connects all of the vertices and is a tree is called a spanning tree | Remember | CO 3 | CLO 11 | AITB05.11 |
| 2 | Define dynamic programming | Dynamic Programming solves problems by combining the solutions of subproblems. | Remember | CO 3 | CLO 13 | AITB05.13 |
| 3 | List characteristics of Dynamic Programming | 1.Optimal Substructure <br> 2.Overlapping subproblems | Remember | CO 3 | CLO 13 | AITB05.13 |
| 4 | List Elements of Dynamic Programming | 1. Substructure <br> 2. Table Structure <br> 3. Bottom-up Computation | Remember | CO 3 | CLO 13 | AITB05.13 |
| 5 | List components of Dynamic programming | 1. Stages <br> 2. States <br> 3. Decision <br> 4. Optimal policy | Remember | CO 3 | CLO 13 | AITB05.13 |
| 6 | List applications of dynamic programming | 1. $0 / 1$ knapsack problem <br> 2. Mathematical optimization problem <br> 3. All pair Shortest path | Remember | CO 3 | CLO 14 | AITB05.14 |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | problem |  |  |  |  |
| 7 | Define matrix chain multiplication complexity | Time Complexity: $\mathrm{O}\left(\mathrm{n}^{3}\right)$ Auxiliary Space: $\mathrm{O}\left(\mathrm{n}^{2}\right)$ | Remember | CO 3 | CLO 14 | AITB05.14 |
| 8 | Define traveling sales person problem. | The problem is to find the shortest possible route that visits every city exactly once and returns to the starting point. | Remember | CO 3 | CLO 14 | AITB05.14 |
| 9 | Define principle of optimality | It states that an optimal sequence of decisions or choices, each sub sequences must also be optimal. | Remember | CO 3 | CLO 15 | AITB05.15 |
| 10 | Define the time complexity of all pairs shortest paths problem. | time complexity of this algorithm is $\mathrm{O}\left(\mathrm{n}^{3}\right)$ | Remember | CO 3 | CLO 15 | AITB05.15 |
| MODULE-IV |  |  |  |  |  |  |
| 1 | What is meant by backtracking? | Backtracking is an algorithmictechnique for solving problems recursively by trying to build a solution incrementally, one piece at a time, removing those solutions that fail to satisfy the constraints of the problem at any point of time | Understand | CO4 | CLO16 | AITB05.16 |
| 2 | Is backtracking same as recursion? | Backtracking is different from recursionIn recursion function calls itself until reaches a base case.where as in backtracking you use recursion in order to explore all the possibilities until you get the best result for the problem. | Remember | CO4 | $\begin{array}{\|c\|} \hline \text { CLO16 } \\ \hline \end{array}$ | AITB05.16 |
| 3 | What is 8 queen problem | The eight queens problem is the problem of placing eight queens on an $8 \times 8$ chessboard such that none of them attack one another (no two are in the same row, column, or diagonal). | Remember | CO4 | CLO16 | AITB05.16 |
| 4 | How many solutions does 8 queens problem have? | The eight queens puzzle has 92 distinct solutions | Remember | CO4 | CLO16 | AITB05.16 |
| 5 | Is subset a sum? | Input: The given set and subset, size of set and subset, a total of the subset, number of elements in the subset and the given sum. Output: All possible subsets whose sum is the same as the given sum. | Remember | CO4 | CLO17 | AITB05.17 |
| 6 | What is the symbol of subset? | The symbol "؟" means "is a subset of". The symbol " $\subset$ " means "is a proper subset of" | Understand | CO4 | CLO17 | AITB05.17 |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 7 | What do u mean by subset? | The set A is contained inside the set B. The subset relationship is denoted as $A \subset B$. | Understand | CO4 | CLO17 | AITB05.17 |
| 8 | How do you color a graph? | Graph coloring is nothing but a simple way of labelling graph components such as vertices, edges, and regions under some constraints. In a graph, no two adjacent vertices, adjacent edges, or adjacent regions are colored with minimum number of colors. | Understand | CO4 | CLO18 | AITB05.18 |
| 9 | What is graph coloring algorithm? | Graph Coloring is a process of assigning colors to the vertices of a graph. It ensures that no two adjacent vertices of the graph are colored with the same color. Chromatic Number is the minimum number of colors required to properly color any graph. | Remember | CO4 | CLO18 | AITB05.18 |
| 10 | What is chromatic number of graph? | The chromatic number of a graph is the smallest number of colors needed to color the vertices of so that no two adjacent vertices share the same color | Remember | CO4 | CLO18 | AITB05.18 |
| 11 | What is Hamiltonian graph? | Every complete graph with more than two vertices is a Hamiltonian graph | Remember | $\mathrm{CO} 4$ | CLO19 | AITB05.19 |
| 12 | What is meant by branch and Bound? | Branch and bound ( $\mathrm{BB}, \mathrm{B} \& \mathrm{~B}$, or $\mathrm{BnB})$ is an algorithm design paradigm for discrete and combinatorial optimization problems, as well as mathematical optimization | Remember | CO4 | CLO19 | AITB05.19 |
| 13 | What is difference between backtracking and branch and bound? | The main difference between backtracking and branch and bound is that the backtracking is an algorithm for capturing some or all solutions to given computational issues, especially for constraint satisfaction issues while branch and bound is an algorithm to find the optimal solution to many optimization problems, | Remember |  | CLO19 | AITB05.19 |
| MODULE-V |  |  |  |  |  |  |
| 1 | What is P class of problems | The class P consists of those problems that are solvable in polynomial time, i.e. these problems can be solved in | Remember | CO1 | CLO1 | AITB05.01 |


| S.No | QUESTION | ANSWER | Blooms Level | CO | CLO | CLO Code |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | time $\boldsymbol{O}\left(\boldsymbol{n}^{k}\right)$ in worst-case, where $\mathbf{k}$ is constant. |  |  |  |  |
| 2 | What are NP class of problems? | The class NP consists of those problems that are verifiable in polynomial time.. | Remember | CO1 | CLO1 | AITB05.01 |
| 3 | What are NP complete problems? | NP complete problems is a collection of problems in NP whose solutions may or may not polynomial time, but however, it can proved that one of them may be solved in polynomial time, then all of them can. | Remember | CO1 | CLO1 | AITB05.01 |
| 4 | What are NP complete problems? | NP-Hard problems is a collection of problems that do not have to be in NP, whose solutions are at least as hard as the NP-Complete problems. If a problem is in NP, and it's NP hard, then it is also NP-Complete. | Remember | CO1 | CLO1 | AITB05.01 |
| 5 | State vertex cover of Graph problem? | A vertex-cover of an undirected graph $\boldsymbol{G}=(\boldsymbol{V}, \boldsymbol{E})$ is a subset of vertices $\boldsymbol{V}^{\prime} \subseteq \boldsymbol{V}$ such that if edge $(\boldsymbol{u}, \boldsymbol{v})$ is an edge of $\boldsymbol{G}$, then either $\boldsymbol{u}$ in $\boldsymbol{V}$ or $\boldsymbol{v}$ in $\boldsymbol{V}^{\prime}$ or both. | Remember | CO1 | CLO1 | AITB05.01 |
| 6 | What is Clique problem? | In an undirected graph, a clique is a complete sub-graph of the given graph. | Remember | CO1 | CLO1 | AITB05.03 |
| 7 | State MAXCLIQUE problem | The Max-Clique problem is the computational problem of finding maximum clique of the graph. Max clique is used in many realworld problems. | Remember | $\mathrm{CO} 1$ | CLO1 | AITB05.01 |
| 8 | What are decision problems? | Problems for which the answer is a Yes or a No such problems are known as decision problems. | Remember | CO1 | CLO1 | AITB05.01 |
| 9 | What is an optimization problem? | Optimization problems are those for which the objective is to maximize or minimize some values. | Remember | CO1 | CLO1 | AITB05.01 |
| 10 | What is a language? | A language is the totality of inputs for which the answer is Yes. | Remember |  | CLO1 | AITB05.01 |
| 11 | Define NP- Completeness. | A language B is NP-complete if it satisfies two conditions <br> - B is in NP <br> - Every $\mathbf{A}$ in NP is polynomial time reducible to $\mathbf{B}$. | Remember | CO1 | CLO3 | AITB05.03 |
| 12 | What is NP-Hard | A problem is NP-hard if all problems in NP are polynomial time reducible to it, even though it may not be in NP itself. | Remember | CO1 | CLO3 | AITB05.03 |
| 13 | What is Satisfiability? | SAT is a problem for which there is no polynomial-time algorithm. | Understand | CO1 | CLO3 | AITB05.03 |
| 14 | What is a intractable problem? | problems for which there is no proof that there cannot be a polynomial-time algorithm are intractable. | Remember | CO1 | CLO2 | AITB05.02 |


| S.No | QUESTION | ANSWER | Blooms <br> Level | CO | CLO | CLO Code |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| 15 | What <br> Chromatic <br> Number? | chromatic number is the <br> minimum number of colors <br> needed to color the vertices of a <br> graph such that no two adjacent <br> vertices have the same color. | Remember | CO1 | CLO2 | AITB05.02 |
| 16 | What is non <br> deterministic <br> algorithm. | A nondeterministic algorithm is <br> an algorithm that, even for the <br> same input, can exhibit different <br> behaviors on different runs | Remember | CO1 | CLO2 | AITB05.02 |
| 17 | State Cooks <br> TheoremCook's theorem, states that the <br> Boolean satisfiability problem is <br> NP-complete. That is, any <br> problem in NP can be reduced in <br> polynomial time by a <br> deterministic Turing machine to <br> the problem of determining <br> whether a Boolean formula is <br> satisfiable | CO1 CLO3 | AITB05.03 |  |  |  |

## Signature of the Faculty

HOD, IT

