VETRI VINAYAHA COLLEGE OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

QUESTION BANK

SUBJECT CODE: CS6402

SUBJECT NAME: DESIGN AND ANALYSIS OF ALGORITHMS

REGULATION: 2013

YEAR / SEM: II/IV

UNIT-I

INTRODUCTION

PART A

1. Define Algorithm.

An algorithm is a sequence of unambiguous instructions for solving a problem in a finite amount of time.

2. Write a short note on Algorithm Design and Analysis of Process.

o Understand the problem

o Decide onComputational DeviceExact Vs Approximate Algorithms

o Algorithm Design Techniques

o Design an algorithms

o Prove Correctness

o Analyze the Algori thm

o Code the Algorithm

3. What are the 2 kinds of Algorithm Efficiency

Time Efficiency-How fast your algorithm runs?

Space Efficiency-How much extra memory your algorithm needs?

4. How can you specify Algorithms?

Algorithms can be specified natural language or pseudo code.

5. What is Pseudo Code?

Pseudo Code is a mixture of Natural Language and Programming Language Constructs such as functions, loops, decision making statements..etc

6. What are the Important Problem Types?

• Sorting

• Searching

• String Processing

• Graph Problem

• Combinatorial Problem

• Geometric Problem

• Numerical Problem

7. How can you Classify Algorithms

Among several ways to classify algorithms, the 2 principal alternatives are
• To group algorithms according to types of problem they solve com
• To group algorithms according to underlying design techniques they are based upon

8. What is Sorting Problem?

Sorting algorithm is rearrange the items of given list descending/ascending order. Sorting algorithms classified into
• Stable Sorting Algorithm
• Non-Stable Algorithm

9. What is Searching Problem?

Finding a given value, called search key given set. Searching Algorithms needs more memory space and sorted array.

10. Write the For LOOP general format.

The general form of a for Loop is
For variable : = value 1 to value 2
step
Step do
{
statement1
statement n
}

11. What is recursive algorithm?

An algorithm is said to be recursive if the same algorithm is invoked in the body. An algorithm that calls itself is Direct recursive. Algorithm A is said to be indeed recursive if it calls another algorithm, which in turn calls A.

12. Differentiate Time Efficiency and Space Efficiency?

Time Efficiency measured by counting the number of times the algorithms basic operation is executed. Space Efficiency is measured by counting the number of extra memory units consumed by the algorithm.

13. What are the features of efficient algorithm?
• Free of ambiguity
• Efficient in execution time
• Concise and compact Completeness
• Definiteness Finiteness

14. Define Order of Algorithm

The order of algorithm is a standard notation of an algorithm that has been developed to represent function that bound the computing time for algorithms. The order of an algorithm is a way of defining its efficiency. It is usually referred as O-notation

15. Define Big Omega Notation.

Omega notation provides lower bound for the function t
A function $t(n)$ is said to Omega $(g(n))$, if there exist some. Positive constant C and some non negative integer n_0, such that $t(n) >= Cg(n)$ for all $n >= n_0$

16. What is Big 'Oh' Notation?

The Big 'Oh' notation provides an upper bound for the function t. A function $t(n)$ is said to be $O(g(n))$, if there exist some positive constant C and some non negative number, such that, $t(n) <= Cg(n)$, for all $n >= n_0$

17. What are the different types of time complexity?

The time complexity can be classified into 3 types, they are

- Worst case analysis
- Average case analysis
- Best case analysis

18.How the algorithm's time efficiency is measured.

Time efficiency indicates how fast an algorithm runs. Time taken by a program to complete its task depends on the number of steps in an algorithm. The time taken by an algorithm is the sum of compile time and execution time. The compile time does not depend on the instance characteristics.

19. Write algorithm using iterative function to fine sum of n numbers.

Algorithm
sum(a,n)
{ S : = 0.0
For i=1 to n do
S : - S + a[i];
Return S;
}

20. Write an algorithm using Recursive function to fine sum of n numbers,

Algorithm Rsum (a,n)
{
If(n≤ 0) then
Return 0.0;
Else
Return Rsum(a, n- 1) + a(n);

PART B

1. Discuss in detail about fundamentals of algorithmic problem solving.
2. Explain the important problem types in detail
3. Explain the general framework for analyzing the efficiency of an algorithm.
4. Explain in detail about asymptotic notation and its properties.
5. Explain the method of solving recurrence equation with suitable example.
6. Explain the method of solving non-recurrence equation with suitable example.

PART C

1. If you have to solve the searching problem for a list of n numbers how can you take advantages of the fact that the list is known to be sorted? Give separate answer for

i) Lists represented as arrays.

ii) Lists represented as linked lists.

2. Solve the following recurrence relations

a) x(n)=x(n-1)+5 for n>1 x(1)=0

b) x(n)=3x(n-1) for n>1 x(1)=4

c) x(n)=x(n-1)+n for n>1 x(0)=0

d) x(n)=x(n/2)+n for n>1 x(1)=1

e) x(n)=x(n/3)+1 for n>1 x(1)=1

3. Consider the following recursion algorithm

      min(A[0,….n-1])

      If n=1 return a[0]

      Else temp=min(A[0,….n-2])

            If temp<=A[n-1] return temp

            Else return A[n-1]

      a) What does this algorithm compute?

      b) Setup a recurrence relation for the algorithm basic operation count and solve it.

4. Give the recursive algorithm for finding the number of binary digits in n's binary representation where n is a positive decimal integer. Find the recurrence relation and complexity.

UNIT-II
BRUTE FORCE AND DIVIDE AND CONQUER
PART A

1. Define Brute force approach.

It is a straightforward approach of solving the problem. It is directly based on the problem statement and definitions of concepts that are directly involved in the problem.

2. What is Empirical Analysis?

It is performed by running a program implementing the algorithm on a sample of inputs and analyzing the data observed. This involves generating pseudocode and random numbers.

2. Define Convex-Hull Problem.

A set of points (finite or infinite) on the plane is called convex if for any two points P and Q in the set, the entire line segment with the end points at P and Q belongs to the set

3. What is Divide and Conquer Algorithm?

It is a general algorithm design techniques that solved problem's instance by dividing it into several smaller instance, solving of them recursively, and then combining their solutions to the original instance of the Problem.

4. What is exhaustive search?

It is a method in which solution is obtained by searching each element of given problem. It makes use of straightforward brute force approach.

5. Enlist the problems in which exhaustive search is carried out.

    1. Traveling salesman problem

    2. Knapsack problem

    3. Assignment problem

6. What is closest pair problem?

The closest pair problem is finding the two closest points from the set of n points.

7. What is travelling salesman problem?

It can be stated as follows consider that there are n cities and travelling salesman has to visit each city exactly once and has to return to the city from where he has started.

8. What is knapsack problem?

Suppose that there are n objects from i=1,2,….n. Each object I has some weight W_i and values V_i associated with each object and capacity of knapsack is W. A thief has to pickup the most valuable object to fill the knapsack to its capacity.

9. What is assignment problem?

Consider that there are n people who need to be assigned to execute n jobs i.e. only one person is assigned to execute one job at a time. Then problem is to find such assignment that gives smallest cost. The cost can br computed as cost   C[i,j] i.e. i^{th} person assigned to j^{th} job.

10. What is binary search?

It is a searching method in which the sorted list of elements is divided at the mid and the key element is searched in left and right sublist. This method is repeated until the sublist gets reduced to only one element. This method is based on divide and conquers algorithmic strategy.

11. What is the necessary precondition for the binary search?

For the binary search the list should be sorted either in ascending or descending order.

12. What is the time complexity of binary search?

Best case:$O(1)$

Average case:$O(\log_2 n)$

Worst case: $O(\log_2 n)$

13. Write down the recurrence relation used for binary search.

$C_{worst}(n)=C_{worst}([n/2]+1)$

$C_{worst}(n)=1$

14. List out the drawbacks of binary search algorithm.

1. In binary search the elements have to be arranged either in ascending or descending order.

2. Each time the mid element has to be computed in order to partition the list in two sublist.

15. What are the advantages of binary search?

1. Efficient searching method.

2. For solving nonlinear equations with one unknown this method is used.

16. Name the sorting techniques that make use of divide and conquer algorithm.

1. Merge sort

2. Quick sort

17. Define merge sort.

The merge sort is a sorting algorithm that uses the divide and conquers strategy. In this method division is dynamically carried out.

Merge sort on an input array with n elements consist of three steps:

Divide: Partition array into two sub lists s1 and s2 with n/2 elements each.

Conquer: Then sort sub list s1 and sub list s2.

Combine: Merge s1 and s2 into a unique sorted group.

18. What is the use of two recursive calls in merge sort algorithm?

In merge sort algorithm two recursive calls are made. Each recursive call focuses on n/2 elements of the list. Thus using the recursive calls the list is divided into sublists for sorting

19. Difference between quick sort and merge sort.

S.No.	Merge sort	Quick sort
1	The positive part of merge sort is that it is stable sort	The quick sort is not a stable sort
2.	It requires extra place for performing the sorting. Hence it is not in place sorting algorithm	Here additional memory is not required.

20. What is the time complexity of binary search?

Best case:$O(n\log n)$

Average case: $O(n\log n)$

Worst case: $O(n\log n)$

PART B

1. Explain the convex hull problem and solution involved behind it.
2. Explain the concept of exhaustive search with the help of an example.
3. Explain in detail about quick sort.
4. Explain in detail about merge sort with an example.
5. Explain binary search in detail.
6. Write short notes on the following
       i) Strassens matrix multiplication
       ii) Multiplication of largest integer.

PART C

1. Solve the following using brute force algorithm. Find whether the given string follows the specified pattern and return 0 or 1 accordingly.
       i) pattern="abba", input="redblueredblue" should return 1
       ii) pattern="aaaa", input="asdasdasdasd" should return 1
       iii) pattern="aabb", input="xyzabcxyzabc" should return 0

2. A pair contain two numbers and its second number is on the right side of the first one in an array. The difference of pair is the minus result while subtracting the second number from the first one . Implement a function which gets the maximal difference of all pairs in an array.(using divide and conquer method)

3. Apply strassens algorithm to compute

$$\begin{vmatrix} 1 & 2 & 1 & 1 \\ 0 & 3 & 2 & 4 \\ 0 & 1 & 1 & 1 \\ 5 & 0 & 1 & 0 \end{vmatrix} * \begin{vmatrix} 2 & 1 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 0 & 3 & 2 & 1 \\ 4 & 0 & 0 & 4 \end{vmatrix}$$

4. Find the optimal solution for the assignment problem given problem

	J1	J2	J3	J4
Person 1	5	8	4	3
Person 2	2	3	7	8
Person 3	1	7	3	2
Person 4	6	4	3	5

UNIT III
DYNAMIC PROGRAMMING AND GREEDY TECHNIQUE
PART A

1. Write the general procedure of dynamic programming.

It is typically applied to optimization problems. For each given problem, we may get any number of solutions from which we seek for optimum solution (i.e. minimum value or maximum value solution). And such an optimal solution becomes the solution to the given problem.

2. State how dynamic programming solves the complex problem.

In dynamic programming method, each subproblem is solved only once. The result of each subproblem is recorded in a table which we can obtain a solution to original problem. This method is based on principle of optimality.

3. Enlist the applications that make use of dynamic programming for the solution.

1. Multistage graphs
2. Finding shortest path
3. Optimal binary search tree
4. 0/1 knapsack problem
5. Traveling salesman problem

4. State the principle of Optimality.

The principle of optimality says that an optimal solution to any instance of an optimization problem is composed of optimal solution to its sub instances.

5. What is need for finding minimum spanning tree?

Spanning tree has many applications. Any connected graphs with n vertices much have atleast-1 edges and connected graphs with n-1 edges are trees. If the nodes of G represent cities and edges represent possible communication links connecting 2 cities, then the minimum number of links needed to connect the cities is -1. Therefore, it is necessary for finding minimum spanning tree.

6. What is spanning tree?

Let G={V,E} be an undirected connected graph. A sub graph t={V,E} of G is a spanning tree of G, if it is tree.

7. What is greedy method?

The greedy method is the most straight forward design, which is applied for change making problem. The greedy technique suggests constructing a solution to an optimization problem through a sequence of steps, each expanding a partially constructed solution obtained so far, until a complete solution to the problem is reached. On each step, the choice made must be feasible, locally optimal and irrevocable.

8. List the advantage of greedy algorithm.

1) Greedy algorithm produces a feasible solution
2) Greedy method is very simple to solve a problem
3) Greedy method provides an optimal solution directly

9. List the applications of minimum spanning tree?

Spanning tree are used to obtain independent set of circuit equations for an electric network.

Another application of spanning tree arises from the property that a spanning tree is minimal sub graph G' of G such that

V (G')=V(G) and G'

10. What is the worst case time complexity of dynamic knapsack problem?

The worst case time complexity of dynamic knapsack problem $O(2^n)$.

11. Give any two properties of dynamic programming approach.

1. Optimal substructure.

2.Overlapping subproblems

12. What is the principle difference between the two techniques?

Both divide and conquer as well as dynamic programming solves the given problem by breaking it into subproblems. But in divide and conquer the subproblems are solved independently where in dynamic programming the subproblems share the solutions among themselves. The dynamic programming works on principle of optimality to solve the problem.

13. What is digraph?

The digraph is a graph in which all the edges are directed then it is called digraph or directed graph.

14. What is transitive closure?

The transitive closure is basically a Boolean matrix in which the existence of directed paths of arbitrarily lengths between vertices is mentioned.

15. What is the purpose of Warshall's algorithm?

It is used to construct transitive closure of a given digraph.

16. What is feasible solution and optimal solution?

Feasible solution: For solving the particular problem there exists n inputs and we need to obtain a subset that satisfies some constraints. Then any subset that satisfies these constraints is called feasible solution.

Optimal solution: Optimal solution is the best choice selected from the set of feasible solution. This solution can be minimum or the maximum value of the solution.

17. Compare feasible and optimal solution.

While solving the problem using Greedy approach solutions are obtained in number of stages. These solutions satisfy problem's constraints. Such solutions are called feasible solutions. Among the feasible solutions if the best solution (either with minimum or with maximum value) is chosen then that solution is called optimal solution

18. What is objective function?

A feasible solution that either minimizes or maximizes a given objective function is called objective function.

19. What is the purpose of huffman's tree?

The huffman's trees are constructed for encoding a given text of n characters. While encoding a given text, each character is associated with some bit sequence. Such a bit sequence is called code word.

20. State the application of huffman' tree.

1. Huffman encoding is used in file compression algorithm.

2. Huffman code is used in transmission of data in the encoded form

3. This encoding is used in game playing method in which decision trees need to be formed.

PART B

1. Write an algorithm for binomial coefficient computation and analyze the efficiency of algorithm.

2. Describe the Warshall's algorithm with an example and analyze its efficiency.

3. Explain Floyd's algorithm for all pair shortest path algorithm with example and analyze its efficiency.

4. Explain OBST algorithm with an example.

5. Explain Dijkstra's algorithm in detail.

6. Explain in detail about MST with an example.

PART C

1. Given the mobile numeric keypad. You can only press buttons that are up, left, right or down to the first number pressed to obtain the subsequent numbers. You are not allowed to press bottom row corner button(i.e. * and #). Given a number N, how many key strokes will be involved to press the given number. What is the length of it? Which dynamic programming technique could be used to find solution for this? Explain each step with the help of a pseudocode and derive its time complexity.

2. The binary string below is a encoded one using Huffman codes

00110001011111011001110111011000001001110100110101

   Given the letter frequencies listed in the table below, build the Huffman code and use them to decode the title. In cases where there are multiple greedy choices, the codes are assembled by combining the first letters from left to right in the order given in the table and assign prefix 0 and 1 for left and right side of tree respectively.

Letter	a	h	v	w	' '	e	t	l	o
Frequency	1	1	1	1	2	2	2	3	3

3. Construct the following weighted graph
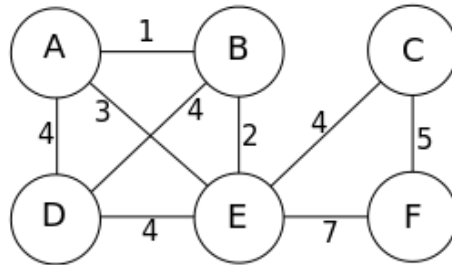
   Give the list of edges in the MST in the order that prim's algorithm insert them. Start prim's algorithm from vertex A

4. Using Algorithm OBST compute W(i,j) , R(i,j) and C(i,j) for the identifier **(end, goto, print, stop)** with **p(1)=0.05, p(2)=0.2, p(3)=0.1, p(4)=0.05, q(0)=0.2, q(1)=0.1, q(2)=0.2, q(3)=0.05, q(4)=0.05**. Using R(i,j) construct the optimal binary search tree.

5. For the given instance of problem obtain the optimal solution for the knapsack problem

Item	Weight	Values
1	5	100
2	7	63
3	8	56

and Capacity **W= 5**

UNIT-IV
ITERATIVE IMPROVEMENT
PART A

1. What is iterative improvement method?

This is a computational technique in which with the help of initial feasible solution the optimal solution is obtained iteratively until no improvement is found.

2. Enlist various application of iterative improvement method.

1. Simplex method
2. Matching graph vertices
3. Stable marriage problem
4. Finding maximum network flow.

3. What is linear programming problem?

The standard form of linear programming is

P=ax+by+cz

A linear programming (LP) problem is a problem in which we have to find the maximum or minimum value of a linear objective function.

4. What is bipartite graph?

The graph G=(V,E) in which the vertex set V is divided into two disjoint sets X and Y in such a way that every edge has one end point in X and other end point in Y.

5. What is two colorable graphs?

The two colorable graphs is a graph that can be colored with only two colors in such a way that no edge connects the same color. The bipartite graph is two colorable graph.

6. What is maximum cardinality matching?

It is a matching with largest number of matching edges.

7. What is maximum matching problem?

The maximum matching problem is a problem of finding maximum matching in a graph.

8. What is augmenting path?

The augmenting path P is a path in a graph G, such that it is an alternating path with special property that its start and end vertices are free or unmatched.

9. When can we say that the optimal solution is obtained in simplex method?

When objective function (i.e. value of Z) is largest then the optimal solution is said to be obtained in simplex method.

10. What is entering variable?

The entering variable is the smallest negative entry in the bottommost row of simplex method.

11. What are basic and non basic variables?

Consider the equation in linear programming form as

ax-by+cz+0s1=0S2=0

where s1 and s2 are basic variables and x, y, z are non basic variables.

12. What is pivot element in simplex method?

The intersection of entering variables and departing variables row is called pivot.

13. State the stable marriage problem.

The problem can be stated as

Consider two sets $M=\{m1,m2,....,m_n\}$ of n men and $W=\{w1,w2,,,,w_n\}$ of n women. Each man has a preference list of ordering the women as potential marriage partners with no ties allowed. Similarly, each woman has a preference list of the men, also with no ties. Then we have to find out the marriage matching pair(m,w) whose members are selected from these two sets based on their preferences.

14. What do you mean by perfect matching in bipartite graphs?

In a bipartite graph, a perfect matching is a matching in which each node has exactly one edge incident on it.

15. Define flow cut.

A cut is a collection of arcs such that if they are removed there is no path from s to t.

PART B

1. Summarize the steps of the simplex method.
2. State and prove max-flow min-cut theorem.
3. Explain briefly about stable marriage algorithm.
4. Determine the time efficiency class of the stable marriage problem.
5. Explain the maximum matching bipartite graph algorithm with supporting example.

PART C

1. Use simplex to solve the farmer's problem given below:

A farmer has a 320 acre farm on which he plants two crops:rice and wheat. For each acre of rice planted, his expenses are 50 and for each acre of wheat planted, his expenses are 100. Each acre of rice requires 100 quintals of storage and yields a profit of 60; each acre of wheat requires 40 quintals of storage and yields a profit of 90. If the total amount of storage space available is 19,200 quintals and the farmer has only 20,000 on hand, how many acres of each crop should he plant in order to maximize his profit? What will his profit be if he follows this strategy?
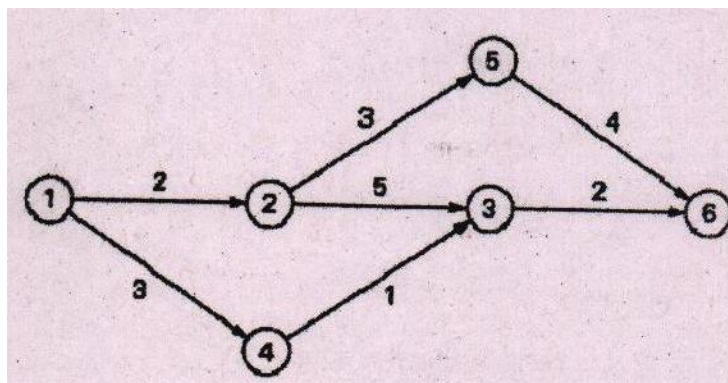
2.Maximize p=2x+3y+z

   Subject to x+y+z<=40

   2x+y-Z>=10

   -y+z>=10

   x>=0,y>=0,z>=0

3. How to you compute maximum flow for the following graph using Ford-Fulkerson method?

4. Consider that there are 4 men & 4 women .The men are named as A,B,C,D and the women are named as I,J,K,L then find the matching pairs.

Men's Preference   Women's Preference

	1st	2nd	3rd	4th
A	I	L	K	J
B	K	L	I	J
C	L	K	J	I
D	J	I	L	K

	1st	2nd	3rd	4th
I	B	A	D	C
J	D	B	C	A
K	A	C	B	D
L	B	D	C	A

UNIT V
COPING WITH THE LIMITATIONS OF ALGORITHM POWER
PART A

1. Define backtracking?

Depth first node generation with bounding function is called backtracking. The backtracking algorithm has its virtue the ability to yield the answer with far fewer than m trials.

2. What is Hamiltonian cycle in an undirected graph?

A Hamiltonian cycle is round trip along n edges of G that visits every vertex once and returns to its starting position.

3. List the application of backtracking technique?

8-Qeens problem

Sum of subset problem

Finding Hamilton problem.

Knapsack problem

4. Given an application for knapsack problem?

The knapsack problem is problem combinatorial optimization. It derives its name from the maximum problem of choosing possible essential that can fit too bag to be carried on trip. A similar problem very often appears business, combinatory, complexity theory, cryptography and applied mathematics.

5. Define subset sum problem?

Subset sum problem is problem, which is used to find a subset of a given set $S=\{S1,S2,S3,.......S_n\}$ of positive integers whose sum is equal to given positive integer d.

6. What is heuristic?

A heuristic is common sense rule drawn from experience rather than from mathematically proved assertion.

For example, going to the nearest unvisited city in the travelling salesman problem is good example for heuristic.

7. State the concept of branch and bound method?

The branch and bound method refers to all state space search methods in which all children of the E-Node are generated before any other live node can become the E-node.

8. Give the upper bound and lower bound of matrix multiplication algorithm?

Upper bound: The given algorithm does n*n*n multiplication hence at most n*n*n multiplication are necessary.

Lower bound: It has been proved in the literature that at least n*n multiplication is necessary.

9. What is state space tree?

Backtracking and branch bound are based on the construction of a state space tree, whose nodes reflect specific choices made for a solution's component .Its root represents an initial state before the search for a solution begins. The nodes of the first level the tree represent the made for the first component of solution, the nodes of the second level represent the Choices for the second components & so on

10. What is promising and non promising node?

A node state space tree is said to be promising, if it corresponds to a partially constructed solution that may still lead to complete solution. Otherwise, node is called non- promising.

11. What are the additional items are required for branch and bound compare to backtracking technique?

Compared to backtracking, branch and bound requires 2 additional items.

1) A way to proved , for every node of node of state space tree, a bound on the best value of the objective function on any solution that can be obtain d by adding further components to the partial solution represented by the node.

2) The value of the best solution seen so far.

12. Differentiate backtracking and branch bound techniques.
- Backtracking is applicable only to non optimization problems.
- Backtracking generates state space tree depth first manner.
- Branch and bound is applicable only to optimization problem.
- Branch and bound generated a node of state space tree using best first rule.

13. Define NP hard.

The NP hard problem is a class of problems in computational complexity that is as hard as the hardest problem in NP. If an NP hard problem can be solved in polynomial time then all the NP complete problems can also be solved in polynomial time.

14. Define NP completeness.

A problem is called NP-complete if
i)   It belongs to class NP
ii)   Every problem in NP can also be solved in polynomial time.

15. What do you mean by decision tree?

All the sorting and searching algorithm are based on comparison method. The comparison is usually made on input items. A model is prepared to study the performance of such algorithms which is called decision tree.

PART B

1. Draw a decision tree and find the number of key comparisons in the worst and average case for the three-element bubble sort.

2. Write backtracking algorithm for 4-queen's problem and discuss the possible solution.

3. Using the example prove that, satisfiability of boolean formula in 3 – conjunctive normal form in NP-complete.

4. Show that the Hamiltonian path problem reduces to the Hamiltonian circuit problem and vice versa.

5. Implement an algorithm for Knapsack problem using NP – Hard approach.

PART C

1. State the subset sum problem and complete state space tree of the backtracking algorithm applied to yhe instance A={3,5,6,7} and d=15 of the subset problem.

2. Explain how job assignment problem could be solved, given n tasks and n agents where each agent has a cost to complete each task, using branch and bound technique.

3. For the given instance of problem obtain the optimal solution for the knapsack problem by branch and bound algorithm.

Item	Weight	Values	and Capacity **W= 5**
1	5	100	
2	7	63	
3	8	56	