PART-A

UNIT-I

1. **Give the Euclid's algorithm for computing gcd (m,n). [may-15]**

```
Algorithm GCD (m,n)
{
  While (n not equal 0) do
   {
     z=m mod n
     m= n
     n= z
      }
 return m:
}
```

2. **Design an algorithm to compute the area circumference of circle. [dec-16]**

```
Algorithm (radius)
 {
  P=3.14
  Area = pi*radius*radius
  Circum = 2* radius
  Write (Area)
  Write (Circum)
}
```

3. Compare the orders of growth of n(n-1)/2 and n^2 [May-16]

n	n(n-1)/2	n^2
1	0	1
2	1	4
3	3	9
4	6	16
5	10	25

4. The (log n) th smallest number of n unsorted numbers can be determined in o(n) average, case time(T/F). [dec-15]

False. The (log n) th smallest number of n unsorted numbers can be determined in o (n) average-case time

5. Write the recursive Fibonacci algorithm and its recurrence relation. [Dec-15]

```
Algorithm fib (n)
{if (n<=1) then
 return n:
else
 return (fib(n-1) + fib(n-2))
```

6. Define program proving and program verification. [May-14]

Program proving means proving each and every instruction of the program with the help of mathematical theorems. Program verification means checking the correctness of the program.

7. What is meant by substitution method? [Dec-14]

Substitution method is used for solving the recurrence relation. It is a kind of method in which a guess for the solution is made.

8. Write down the properties of asymptotic notations. [May-15]

Let f(n) and g(n) be two non- negative functions.

Let, n_0 and constant c are two integers such that n_0 denotes some value of input and $n > n_0$.

9. What is average case analysis? [May-14]

All the possible inputs are considered and the computing time for all inputs is calculated. The sum of all the calculated values is then divided by total number of inputs.

10. What are the components of fixed and variable part in space complexity? [Dec-13]

Space requirement s(p) is given by formula:

$s(p) = C + Sp$

Where C is constant that denotes the space of inputs and outputs.

Sp denotes the variable part of the space

11. Define algorithm validation. [Dec-12]

The process of measuring effectiveness of the algorithm actually making program or code form it, in order to know the weather algorithm, correct for valid input is known as algorithm validation.

12. Define θ-notation? [Dec-14]

A function t(n) is said to be in θ (g(n)), denoted by t(n) ε θ (g(n)), if t(n) is bounded both above & below by some constant multiple of g(n) for all large n, i.e., if there exists some positive constants c1 & c2 and some nonnegative integer n0 such that

$$c_2 g(n) <= t(n) <= c_1 g(n) \text{ for all } n >= n0$$

13. **Define Ω-notation? [May-13]**

A function t(n) is said to be in Ω (g(n)), denoted by t(n) ε Ω (g(n)), if t(n) is bounded below by some constant multiple of g(n) for all large n, i.e., if there exists some positive constant c and some non-negative integer n_0 such that

$$T(n) >= cg(n) \text{ for all } n >= n_0$$

14. **Define O-notation? [May-12]**

A function t(n) is said to be in O(g(n)), denoted by t(n) ε O(g(n)), if t(n) is bounded above by some constant multiple of g(n) for all large n, i.e., if there exists some positive constant c and some non-negative integer n_0 such that

$$T(n) <= cg(n) \text{ for all } n >= n_0$$

15. Establish relation between O and Ω. [Dec-10]

The notation O represents the upper bound of the algorithm running time. The notation Ω represent the lower bound of the algorithm running time.

16. Differentiate time complexity from space complexity. [May-10]

Time complexity is amount of time required by a program to execute. Space complexity is amount of space required by a program to execute.

17. What is an algorithm? [May-17]

An algorithm is a step by step method of solving a problem. It is commonly used for data processing, calculation and other related computer and mathematical operations.

18. Define recurrence relation. [Dec-16]

A **recurrence relation** is an **equation** that **defines** a sequence based on a rule that gives the next term as a function of the previous term(s). The simplest form of a **recurrence relation** is the case where the next term depends only on the immediately previous term..

19. What are the classical geometric problems?

- **The closest pair problem***:* given n points in a plane find the closest pair among them
- **The convex hull problem:** find the smallest convex polygon that would include all the points of a given set.

20. **What is algorithm visualization?**

Algorithm visualization is a way to study algorithms. It is defined as the use of images to convey some useful information about algorithms.

UNIT-II

1. What is meant by divide and conquer technique. (or) give general strategy of divide and conquer method ? [MAY 13, 16]

Divide and conquer is an algorithmic strategy in which the given problem is broken in to the smaller sub problems and the solutions to these sub problem is obtained .later on the solutions are combined together to obtain the solution for the original given problem

2. Write the control abstraction for divide and conquer? [DEC 12, 13]

```
Algorithm DC (p)
 {
        If p is too small then
        Return solution of p
       Else
        {
         Divide (p) and obtain p1,p2,……..pn
Where n≥1
       Apply DC to each sub problem
        Return combine (DC (P1),(DC(P2)……(DC(Pn));
    }
}
```

3.) What is the time complexity of binary search? [MAY 11, 12, DEC 16]

Time complexity of binary search

Best case	**Average case**	**Worst case**
$\theta(1)$	$\theta(\log_2 n)$	$\theta(\log_2 n)$

4.) List out two drawbacks of binary search algorithm?[DEC -07]

* In binary search the element have to be arranged either in ascending or descending order

* Each time the mid element has to computed in order to partition the list in two sub list

5.)Trace the operation of binary search algorithm for the input _15,_6,0,7,9,23,54,82,101,112,125,131,142,151 if you are searching for the element 9.[DEC-10]

* The key element i.e.the element to searched is 9

*as key =A [MID], the message "number is present in given list" can be displayed.

6.) What is difference between quick sort and merge sort? [MAY -13]

S.NO	MERGE SORT	QUICK SORT
1	The positive part of merge sort is that is it stable sort	The quick sort is not a stable sort
2	The negative part of merge sort is that it requires extra place for performing sorting It s not in place sorting algorithm	The quick sort is place sorting algorithm due to additional memory is not required

7.) Give the time efficiency and drawback of merge sort algorithm [DEC -05]

The time efficiency: the best, worst and average case time complexity of merge sort is o(n log n)

Drawback:

- The method is slower than the quick sort method
- This method is complicated to code
- The requires extra storage to execute this method

8.) Differentiate linear search and binary search techniques [DEC- 14]

S.NO	Linear search	Binary search
1	Less efficient method	Efficient method
2	Simple to implement	Additional computation is required for computing mid element

9.) What is closest pair problem? [MAY 16]

The exists a set of points on a plane which said to be convex if for any two points A and

B in the set, entire line segment with end points at A and B set

10.) Name at least two sorting techniques that make use of divide and conquer algorithm [MAY-9]

- Both **merge sort** and **quick sort** employ a common algorithmic paradigm based on **recursion**.
- This paradigm, divide-and-conquer, break a problem into **subproblems**that are similar to the original problem, recursively solves the **sub problems**, and finally combines the solutions to the **sub problems** to solve the original problem.

11. What is brute force algorithm?

A straightforward approach, usually based directly on the problem's statement and definitions of the concepts involved.

12. What is exhaustive search?

A brute force solution to a problem involving search for an element with a special property, usually among combinatorial objects such as permutations, combinations, or subsets of a set.

13. Give the general plan of exhaustive search.
Method:

- generate a list of all potential solutions to the problem in a systematic manner
- evaluate potential solutions one by one, disqualifying infeasible ones and, for an optimization problem, keeping track of the best one found so far
- when search ends, announce the solution(s) found

14. Define of feasibility

A feasible set (of candidates) is promising if it can be extended to produce not merely a solution, but an optimal solution to the problem.

15. Define Hamiltonian circuit.

A Hamiltonian circuit is defined as a cycle that passes through all the vertices of the graph exactly once.

16. List out Disadvantages of Divide and Conquer Algorithm

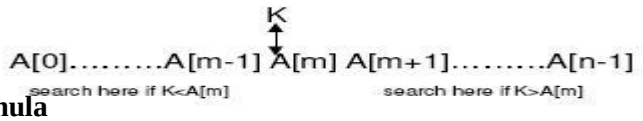
- Conceptual difficulty
- Recursion overhead
- Repeated subproblems

17. What is the difference between quick sort and merge sort?

Both quick sort and merge sort use the divide-and-conquer technique in which the given array is partitioned into subarrays and solved. The difference lies in the technique that the arrays are partitioned. For merge sort the arrays are partitioned according to their position and in quicksort they are partitioned according to the element values.

18. List out the 4 steps in Strassen's Method?
1. Divide the input matrices A and B into n/2 * n/2 submatrices, as in equation (1).
2. Using Θ(n2) scalar additions and subtractions, compute 14 n/2 * n/2 matrices A1, B1, A2, B2, …, A7, B7.
3. Recursively compute the seven matrix products Pi =AiBi for i =1, 2, 7.
4. Compute the desired submatrices r, s, t, u of the result matrix C by adding and/or subtracting various combinations of the Pi matrices, using only Θ(n2) scalar additions and subtractions.

19. What is binary search?



20. List the strassen formula

$m_1 = (a_{00} + a_{11}) * (b_{00} + b_{11})$
$m_2 = (a_{10} + a_{11}) * b_{00}$
$m_3 = a_{00} * (b_{01} - b_{11})$
$m_4 = a_{11} * (b_{10} - b_{00})$
$m_5 = (a_{00} + a_{01}) * b_{11}$
$m_6 = (a_{10} - a_{00}) * (b_{00} + b_{01})$
$m_7 = (a_{01} - a_{11}) * (b_{10} + b_{11})$

UNIT-III

1. What is the principle used behind the dynamic programming? [Dec-16]
The principle of optimality is used behind the dynamic programming.

2. State the principle of optimality. [Dec-14]

The principle of optimality state that in an optimal sequence of decision or choice each subsequence must also be optimal.

3. Define travlesman problem. [Dec-11]

If there are n cities and cost of traveling from any city to any other city is given, then we have to obtain cheapest round trip such that each city is visited exactly once and then returning of starting city completely the tour.

4. State the 0/1 knapsack problem. [Dec-14]

If we are given n objects and a knapsack are a bag in which object I that has weight w_i is to be placed. The knapsack has a capacity W.

5. Define optimal binary search tree. [May-10]

Let $\{a_1,a_2.....a_n\}$ be a set of identifier such that $a_1<a_2<a_3$ let p(i) the probability which we can search for a_i and q(i) be the probability of searching elements such that $ai< x<ai_{+1.}$

6. Define the single source shortest paths problem. [May-16]

The **problem** is also sometimes called the **single-pair shortest path problem**, to distinguish it from the following variations: 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.

7. State the assignment problem. [May-16]

Solving the state assignment problem **means finding** the optimum assignment for each state within a **sequential** digital circuit. These optimum assignments will result in decreasing the hardware realization cost and increasing the reliability of the digital circuit.

8. What is meant by principle of optimality? [Dec-16]

The principle of optimality is the basic principle of dynamic programming, which was developed by Richard Bellman: that an optimal path has the property that whatever the initial conditions and control variables (choices) over some initial period, the control (or decision variables) chosen over the remaining period must be optimal for the remaining problem, with the state resulting from the early decisions taken to be the initial condition.

9. How to calculate the efficiency of dijkstra's algorithm? [Dec-16]

- Each vertex is connected to (V-1) vertices.
- Number of adjacent edges to each vertex is V-1.
- Let E represents the total number of edges.
- Finding and uploading each adjacent vertex's weight is O(log V)

10. State the general principle of greedy algorithm. [May-17]

A **Greedy algorithm** makes **greedy** choices at each step to ensure that the objective function is optimized. The **Greedy algorithm** has only one shot to compute the optimal solution so that it never goes back and reverses the decision.

11. What do you mean by dynamic programming? [May-17]

In mathematics, management science, economics, computer science, and bioinformatics, **dynamic programming** (also known as **dynamic** optimization) is a method for solving a complex problem by breaking it down into a collection of simpler sub problems, solving each of those sub problems just once, and storing their solutions.

12. State how binomial coefficient is computed. [Dec-15]

C (n,k)= C(n-1, k-1)+ C(n-1, k)
C (n, 0) =1
C (n,n) = 1
n> k> 0

13. List out the memory functions used under dynamic programming. [May-15]

Binomial coefficient

C (n,k)= C(n-1, k-1)+ C(n-1, k)

C (n,n) = 1

Warshall algorithm

$R^{(k)}_{ij} = R^{(k-1)}_{ij}$

14. Define principle of optimality.

It states that an optimal sequence of decisions has the property that whenever the initial stage or decisions must constitute an optimal sequence with regard to stage resulting from the first decision.

15. Write the difference between the Greedy method and Dynamic programming.

Greedy method

0. Only one sequence of decision is generated.

1. It does not guarantee to give an optimal solution always.

Dynamic programming

1. Many numbers of decisions are generated.

2. It definitely gives an optimal solution always.

16. What is the Greedy choice property?

- The first component is greedy choice property (i.e.) a globally optimal solution can arrive at by making a locally optimal choice.
- The choice made by greedy algorithm depends on choices made so far but it cannot depend on any future choices or on solution to the sub problem.
- It progresses in top down fashion.

17. What are the steps required to develop a greedy algorithm?

- Determine the optimal substructure of the problem.
- Develop a recursive solution.
- Prove that at any stage of recursion one of the optimal choices is greedy choice. Thus it is always safe to make greedy choice.
- Show that all but one of the sub problems induced by having made the greedy choice are empty.

18. What are the labels in Prim's algorithm used for?

Prim's algorithm makes it necessary to provide each vertex not in the current tree with the information about the shortest edge connecting the vertex to a tree vertex. The information is provided by attaching two labels to a vertex.

- The name of the nearest tree vertex.
- The length of the corresponding edge

19. How are the vertices not in the tree split into?

The vertices that are not in the tree are split into two sets

Fringe: It contains the vertices that are not in the tree but are adjacent to at least one tree vertex.

Unseen: All other vertices of the graph are called unseen because they are yet to be affected by the algorithm.

20. What is minimum spanning tree?

Minimum spanning tree of a weighted, connected graph *G*: a spanning tree of *G* of the minimum total weight

UNIT-IV

1).Derive the iterative improvement technique [Dec-16]

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

2).Enlist various applications of iterative improvement method. [Dec-15]

Various application of iterative improvement method are-

1. Simplex method 2.matching graph vertices
3. Stable marriage problem 4.finding maximum network flow.

3).what is linear programming problem? [May-16]

The standard form of linear programming is-

P=ax+by+cz

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

4).what is bipartite graph? [Dec-16]

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 e€ E has one end point in X and other end point in Y.

5).what is maximum cardinality matching? [Dec-16]

It is a matching with largest number of matching edges.

6).what do you mean by perfect matching in bipartite graphs? [May-15, 17]

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

7).define flow 'Cut'. [May -15]

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

8).define network flow and cut. [Dec-15]

Network flow: Given a directed graph G with non –negative integer weights and two distinguished vertices s and t called source and the sink, such that the source only has out – edges and sink only has in edges, the maximum amount of commodity that can flow through network from source to sink ,is called network flow.

9).what is an articulation point in graph? [May -17]

A vertex in an undirected graph is an articulation point (cut vertex) if and only if removing it disconnects the graph.

10. What is cut and min cut?

Let X be a set of vertices in a network that includes its source but does not include its sink, and let X, the complement of X, be the rest of the vertices including the sink. The *cut* induced by this partition of the vertices is the set of all the edges with a tail in X and a head in X.

Capacity of a cut is defined as the sum of capacities of the edges that compose the cut.

Minimum cut is a cut of the smallest capacity in a given network

11. State max – flow – min – cut theorem.

The value of maximum flow in a network is equal to the capacity of its minimum cut.

12. Define optimal finish time.

Optimal finish time scheduling for a given set of tasks is a non-preemptive schedule S for which F (S) is minimum over all non-preemptive schedules S.

13. Define preemptive optimal finish time.

Preemptive optimal finish time scheduling for a given set of tasks is a p

14. What is augmentation and augmentation path?

- The length of an augmenting path is always odd
- Adding to M the odd numbered path edges and deleting from it the even numbered path edges increases the matching size by 1 [*augmentation*]

15. Write the detailed description about simplex method.

Step 0 [Initialization]

Step 1 [Optimality test]

Step 2 [Find entering variable].

Step 3 [Find departing variable]

Step 4 [Form the next tableau]

16. State Extreme point theorem.

Extreme point theorem states that if S is convex and compact in a locally convex space, then S is the closed convex hull of its extreme points: Convex set has its extreme points at the boundary. Extreme points should be the end points of the line connecting any two points of convex set.

17. **What is basic solution?**

A *basic solution* to a system of m linear equations in n unknowns ($n \geq m$) is obtained by setting $n - m$ variables to 0 and solving the resulting system to get the values of the other m variables. The variables set to 0 are called *non-basic*; the variables obtained by solving the system are called *basic*.

18. Define flow and flow conservation requirement.

- *Flow* is an assignment of real numbers x*ij* to edges (*i*,*j*) of a given network that satisfy the following:
- ***Flow-conservation requirements*:** The total amount of material entering an intermediate vertex must be equal to the total amount of the material leaving the vertex

19.Define - Integer Linear Programming

It is a Programming to find the minimum value of a linear function of several integer-valued variables subject to a finite set of constraints in the form of linear equalities and/or in equalities.

20. List the Stable marriage algorithm

Proposal The selected free man m proposes to w, the next woman on his preference list (who is the highest-ranked woman who has not rejected him before).

11.

Response If w is free, she accepts the proposal to be matched with m. If she is not free, she compares m with her current mate. If she prefers to him, she accepts m‘s proposal, making her former mate free; otherwise, she simply rejects m‘s proposal, leaving m free.

Step 2 Return the set of n matched pairs.

UNIT-V

1) What is lower bound? (May-16)

Obtaining lower bound means estimating minimum amount of work needed to solve the problem.

2) State the principle of backtracking. (or) explain the idea behind the backtracking.? (Dec-5, 10, 11, May-12)

Backtracking is a method in which the desired solution is expressed as **n** tuple ($x_1,x_2,x_3,\ldots..x_{n)}$ which is chosen from solution space, using backtrack formulation. The solution obtained i.e($x_1,x_2,x_3,\ldots x_n$) can either minimize or maximize or satisfies the criteria function.

3) What is state space tree?

A State Space tree is a rooted tree whose nodes represent partially constructed solutions to given problem. In backtracking method the state space tree is built for finding the solution. This tree is built using depth first search fashion.

4) Give the formula definition of n-queen's problem? (May-8)

Definition: Consider **n x n** Chessboard on which we have to place **n** queens such that no two queens can attack each other by being in the same row or in the same column or the same diagonal.

The diagonal conflicts can be checked by following formula-

Let, P1= (i, j) and P2 (K, l) are two positions.

Then P1and P2 is the position that is on the same diagonal, if

- i +j = K+l or
- i –j = K-l

5) Differentiate explicit and implicit constraints? (Dec-13, May-10, 12, 14)

Explicit Constraints: Explicit constraints are rules, which restrict each vector element to be chosen from given set.

Implicit Constraints: Implicit constraints are rules which determine which of the tuples in the solution space satisfy the criterion function.

6) What is the difference between live node and dead node?

Live node: The live node which is generated and whose children have not yet been generated is called live node. While tracing for the solution each internal node is a live node.

Dead node: the dead node is a generated node which is not to be expanded further or all of whose children have been generated.

7) Describe the sum of subset problem? (May-13)

Let' S={S1,…..Sn} be a set of on **n** positive integers, then we have to find a subset whose sum is equal to given positive integer d.

It is always convenient to sort the set's element in ascending order That is,

$$S1 \leq S2 \leq \ldots\ldots \leq Sn$$

8) What are the additional items required for branch and bound to compare backtracking technique? (Dec-6)

In comparison with backtracking the branch and bound requires two additional elements and those are-

i) The **value of best solution** obtained so far.

ii) A **bound on the best value** of the objective of the objective function. This value can be obtained by adding further components to the partial solution represented by the node at that instance .due to this bounding value one can traverse the state tree in certain direction and this ultimately leads to the solution for the given problem.

iii) A criteria to determine **promising sub problems** to explore.

9) Define Hamiltonian circuit problem (Dec-14, May-15)

Hamiltonian circuit problem is a problem of finding a Hamiltonian circuit. Hamiltonian circuit is a circuit that visits every vertex exactly once and return to the starting vertex.

10) What is the property of NP-complete problem? (Dec-12)

A problem D is called NP-complete if

i) It belongs to class NP.

ii) Every problem in NP can also be solved in polynomial time.

11. Define p and np problems (May-17)

The P versus NP problem is a major unsolved problem in computer science. It asks whether every problem whose solution can be quickly verified can also be solved quickly

12. What is an articulation point in a graph? (May-17)

Articulation Points (or Cut Vertices) in a **Graph**. A vertex in an undirected connected **graph** is an **articulation point** (or cut vertex) if removing it (and edges through it) disconnects the **graph**.

13. Write the formula for decision tree for searching a sorted array. (Dec-16)

$C_{worst}^{(n)} = [\log_2 n]+1$

$^{)} = [\log_{2(}n+1)]$

14. Specify the reasons for terminating the search in branch and bound. (Dec-16)

- Value of bound obtained is not better than the best solution obtained so far.
- Node represents no feasible solution because the given constraint is not satisfied by the feasible solution of that node.

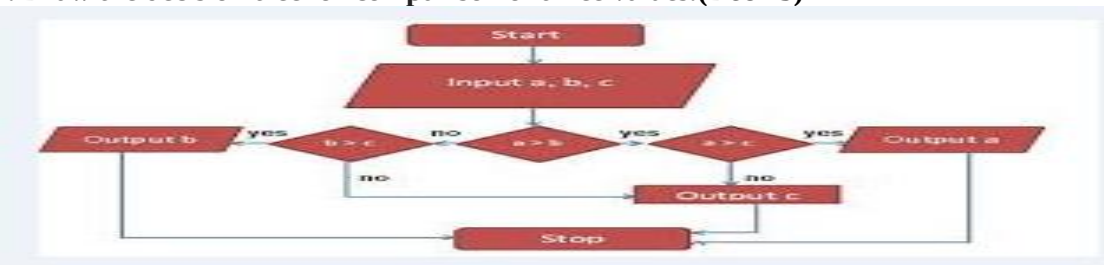
15. Give the purpose of lower bound. (May-16)

Similarly a **lower bound** is said to be a tight **lower bound**, a greatest **lower bound**, or an infimum if no greater value is a **lower bound**.

16. What is Euclidean minimum spanning tree problem? (May-16)

The **Euclidean minimum spanning tree** or EMST is a **minimum spanning tree** of a set of n points in the plane (or more generally in $\mathbb{R}^d$), where the weight of the edge between each pair of points is the **Euclidean** distance between those two points.

17. Draw the decision tree for comparison of three values.(Dec-15)



18. List out the implementation procedure of Backtracking

As usual in a recursion, the recursive function has to contain all the knowledge. The standard implementation is:

1. check if the goal is achieved REPEAT
2. check if the next step is possible at all
3. check if the next step leads to a known position - prevent circles

PART-B

16 MARK QUESTIONS

UNIT-1

1. Discuss various methods used for mathematical analysis of recursive algorithms..
2. Elaborate asymptotic analysis of an algorithm with an example.
3. Briefly explain the mathematical analysis of recursive algorithm with example.
4. Use the most appropriate notation to indicate the time efficiency class of sequential search

algorithm in the worst,best and the average case.

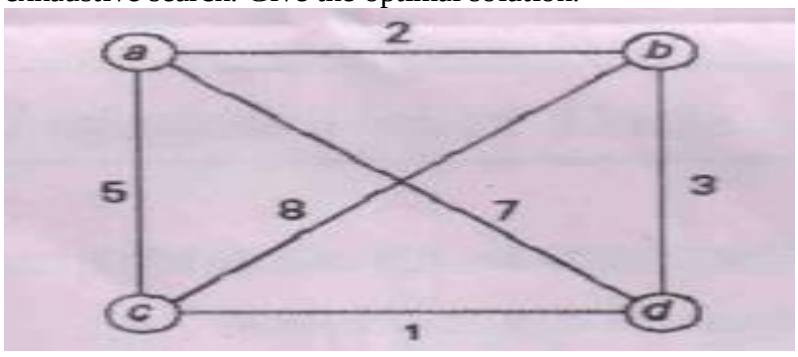
5. State the general plan for analysis the time efficiency of non recursive algorithm and explain with an example.
6. Discuss the steps in mathematical analysis for recursive algorithms. Do the same for finding the factorial of a number.
7. Give the algorithm to check whether all the elements in a given array of n elements are distinct. Find worst case complexity of the same.
8. 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.
9. Derive the worst case analysis of merge sort using suitable illustrations.
10. Write an algorithm for quick sort and write its time complexity with example list are 5,3,1,9,8,2,4,7

UNIT-II

1. State the travelling salesman problem. Elaborate the steps in solving the travelling salesman problem using brute force approach
2. Write the algorithm to perform binary search and compute run time complexity.
3. There are 4 people who need to be assigned to execute 4 Jobs (one Person per jod) and the problem is to find an assignment with the minimum total cost. The assignment cost is given below, solve the Assignment problem by exhaustive search.

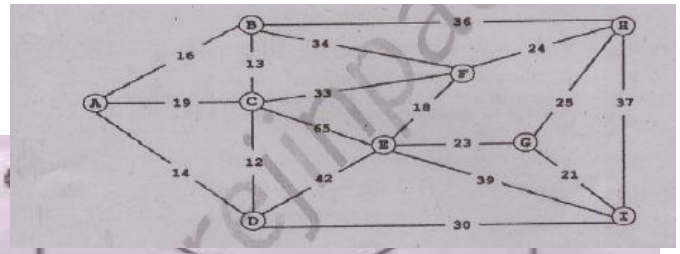
	Job 1	Job 2	Job 3	Job 4
Person 1	9	2	7	8
Person 2	6	4	3	7
Person 3	5	8	1	8
Person 4	7	6	9	4

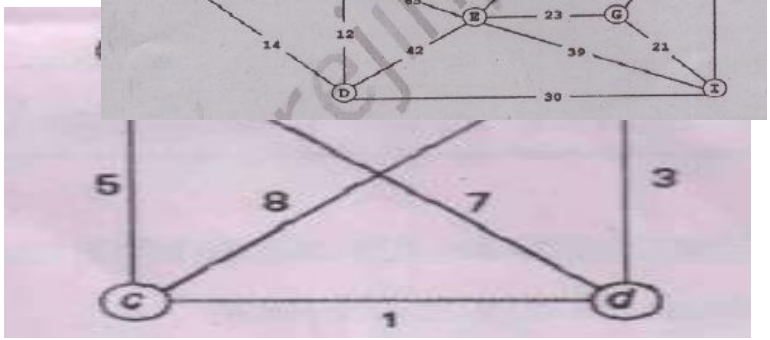
4. Explain Heap sort algorithm with an example
5. Explain merge sort algorithm with an example.
6. Write down the algorithm to construct an convex hull based on brute force technique.
7. Find all the solutions to the travelling salesman problem[cities and distances shown below] by exhaustive search. Give the optimal solution.



8. Explain about the marriage problem with suitable example.

9. Discuss about the algorithm and pseudo code to find the minimum spanning tree using prims algorithm. Find the minimum spanning tree for the graph shown below.





9.Explain Floyds war shall algorithm using dynamic programming. Trace the algorithm for the given example

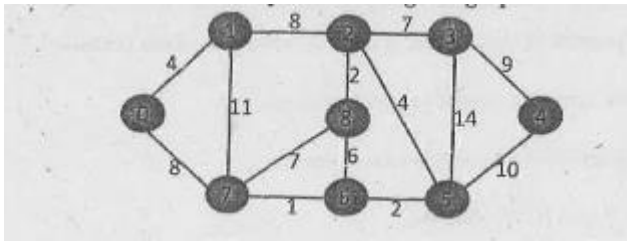
10.Explain the method used for performing multiplication of 2 integers. Explain how divide and conquer method can be used to solve the same.

UNIT-III

1.Solve the following instance of the 0/1 knapsack problem given the knapsack capacity in W=5 using dynamic programming and explain it.

Items	Weight	Value
1	4	10
2	3	20
3	2	15
4	5	25

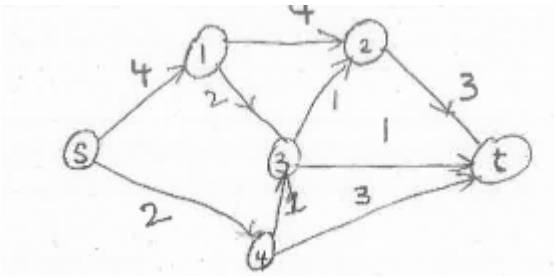
2. Explain how greedy approach is used Dijikstra's algorithm for finding the single source shortest path for the given graph.



3.Write the Huffman's algorithm. construct the Huffman tree for the following data and obtain its Huffman's code.

Character	A	B	C	D	E	–
Probability	0.5	0.35	0.5	0.1	0.4	0.2

4.Determine the max-flow in the following network.



5.Solve the following set of equations using simplex algorithm:

Max :18x1+12.5x2

Sub to :x1+x2<=20

X1<=12

X2<=16

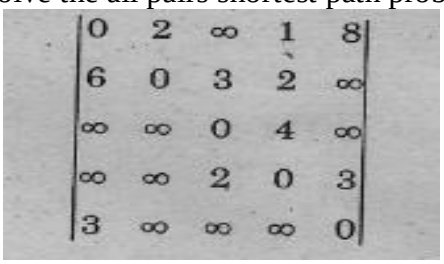
X1,x2>=0

6.What is iterative improvement? elaborate step in simplex method with an example.

7.What is bipartite graph? Is the subset of bipartite graph bipartite? Outline with an example.

8.Explain the brute force method to method to find closest points in a set of n points in k dimensional space.

9.Solve the all pairs shortest path problem for the diagraph the following weight matrix.

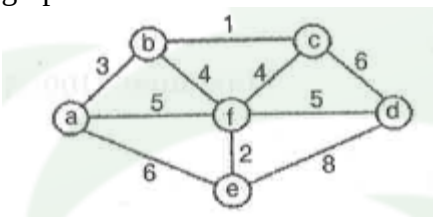


0	2	∞	1	8
6	0	3	2	∞
∞	∞	0	4	∞
∞	∞	2	0	3
3	∞	∞	∞	0

10.Explain the memory function method for the knapsack problem and give the algorithm.

UNIT-IV

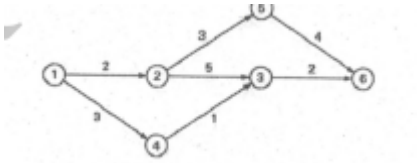
1.Give the pseudo code for prims algorithm and apply the same to find the minimum spanning tree of the graph shown below.



2.Explain multi stage graph using dynamic programming technique

3.Outline the stable marriage problem with an example.

4.Apply the shortest augmenting path problem to the network shown below.



5.Prove that the stable marriage algorithm terminates after no more then n^2 iteration with a stable marriage output.

6.Write backtracking algorithm for 4 queen's or N queens problem and discuss the possible solution.

7.Solve the following instance of knapsack problem by branch and bound algorithm.

Item	Weight	Profit	
1	5	$40	
2	7	$35	
3	2	$18	W = 15
4	4	$4	
5	5	$10	
6	1	$2	

8.Consider the travelling salesman instance defined by the following cost matrix

∞	20	30	10	11
15	∞	16	4	2
3	5	∞	2	4
19	6	18	∞	3
16	4	7	16	∞

Draw the state space tree and show the reduced matrix corresponding to the each of the node.

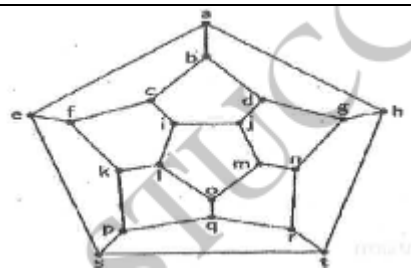
10. Find the optimal solution using branch and bound for the following assignment problem.

	Job1	Job 2	Job 3	Job 4
A	9	2	7	8
B	6	4	3	7
C	5	8	1	8
D	7	6	9	4

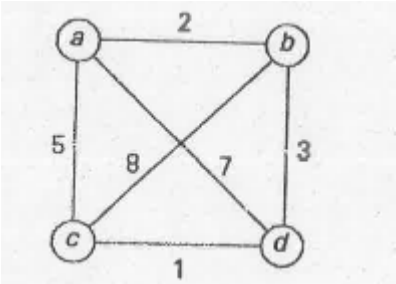
.

UNIT-V

1.Find the Hamiltonian circuit or disprove its existence in the graph given below.



2.Apply branch and bound algorithm to solve travelling saleman problem



3. What is class NP? Discuss about any five problems for which no polynomial- time algorithm has been bound.

4.Elaborate on the nearest-neighbor algorithm and multifragment heuristic algorithm for TSP problem.

5.Give any five undecidable problem and explain the famous halting problem.

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

7.Prove that the stable marriage algorithm terminates after no more then n^2 iteration with a stable marriage output.

8.Write backtracking algorithm for 4 queen's or N queens problem and discuss the possible solution.

9.Explain Heap sort algorithm with an example
10.Explain merge sort algorithm with an example.