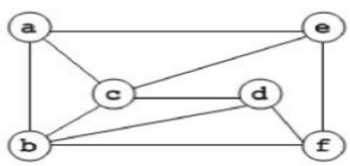
ABES Engineering College, Ghaziabad

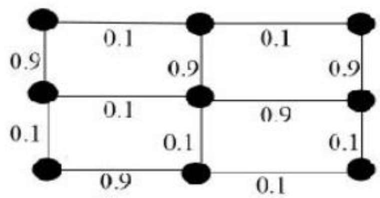
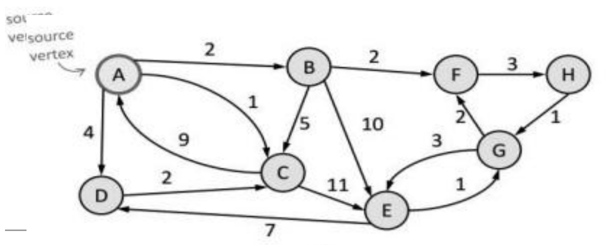
B.Tech Odd Semester Sessional Test-2

Printed Pages: 2
Session:2022-2023
Roll No.
Date of Exam:30-11-2022

Course Code:KCS503
Course Name: Design & Analysis of Algorithm
Maximum Marks:75
Time: 2 hrs

Instructions:
1. Attempt All sections.
2. If require any missing data, then choose suitably.

Q.No.	Question	Marks	CO	KL	PI
	Section-A	**Total Marks : 5*2=10**			
1	**Attempt ALL Parts**				
a)	Consider a graph G=(V,E) where V= (v1, v2, v3,……v100), E = { (vi , vj)| 1<=i<j<=100}, and weight of the edge (v_i ,v_j) is | i- j|. The weight of minimum spanning tree is? [**GATE 2018**]	**2**	**CO3**	K2	1.4.1
b)	Which algorithm can be used to find the solution of the problem in minimum time: Given a set of non-negative integer, and a value K, determine if there is a subset of the given set with sum equal to K. Explain that algorithm. [**ISRO 2018**]	**2**	**CO4**	K2	2.2.3
c)	The Chromatic number of the following graph is 	**2**	**CO4**	K2	1.4.1
d)	Differentiate between P and NP problem.	**2**	**CO5**	K2	2.2.4
e)	When RANDOMIZED-QUICKSORT runs, how many calls are made to the random number generator RANDOM in the worst case? How about in the best case? Give your answer in terms of Θ notation.	**2**	**CO5**	K2	2.2.4
	Section-B	**Total Marks : 3*5=15**			
2	**Attempt ANY ONE part from the following**				
a)	What is an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers? a:1 b:1 c:2 d:3 e:5 f:8 g:13 h:21	**5**	**CO3**	K3	2.2.3
b)	Consider 5 items along with their respective weights and value: I= < I1, I2, I3, I4. I5> W=<5,10,20,30,40> V=<30,20,100,90,160> The capacity of knapsack W=60. Find the solution to fractional knapsack problem.	**5**	**CO3**	K3	2.2.3
3	**Attempt ANY ONE part from the following**				
a)	Consider product of three matricesM1, M2, M3 having wrows and x columns, x rows and y columns, and y rows and z columns. Under what condition will it take less time to compute the product as (M1M2)M3 than to compute M1(M2M3)? **[GATE 2020]**.	**5**	**CO4**	K2	2.2.3
b)	Define Sum of Subset Problem. Let S={4,6,7,8} and m=18. Find all possible subsets of S that sums to m. Generate the state space tree.	**2+3**	**CO4**	K2	2.2.3
4	**Attempt ANY ONE part from the following**				
a)	Compute the prefix function π for the pattern P = a b a c a b using KNUTH -MORRIS-PRATT algorithm.	**5**	**CO5**	K3	2.2.3
b)	Determine the running time of recursive FFT, each invocation take time Θ(n),exclusive of recursive call where n is the length of input vector.	**5**	**CO5**	K3	2.2.3

Section-C		Total Marks : 5*10=50			
5	**Attempt ANY ONE part from the following**				
a)	Consider the following undirected graph with edge weight as shown. Find the minimum weight spanning tree,along with algorithm & Complexity.[**GATE 2020**]  	**5+3+2**	**CO3**	K3	1.4.1
b)	Apply greedy single source shortest path algorithm on following graph.Also write the same Algorithm with complexity.  	**5+3+2**	**CO3**	K3	1.4.1
6	**Attempt ANY ONE part from the following**				
a)	A thief enters a house for robbing it. He can carry a maximal weight of 13 kg into his bag. There are 4 items in the house with the following weights and values. Item Mirror Painting Vase earphone Value (Rs.) 28 18 23 9 Weight (kg) 7 5 6 2 What items should thief take if he either takes the item completely or leaves it completely?	**10**	**CO4**	K3	2.2.3
b)	Write Recursive formula to compute Longest Common Subsequence of two Sequences X = {7, 12 15, 14, 21, 13, 14, ,11, 10, 9} and Y = {20, 7, 15, 1, 14, 3, 6, 13, 11, 18, 10, 9}. Also Show the steps used to print the LCS of X and Y.	**2+6+2**	**CO4**	K3	2.2.3
7	**Attempt ANY ONE part from the following**				
a)	Find an optimal parenthesization of a matrix chain product whose sequence of dimensions is {10, 5, 3, 12, 6}.	**10**	**CO4**	K3	2.2.3
b)	Show the solution of' 4 queen problem using backtracking approach with state space tree.	**10**	**CO4**	K3	2.2.3
8	**Attempt ANY ONE part from the following**				
a)	Write Rabin-Karp algorithm. For string matching working modulo q = 11, how many spurious hits does the Rabin-Karp matcher encounter in the text T =3141592653589793 when looking for pattern P = 26?	**4+6**	**CO5**	K3	2.2.3
b)	Write Boyer-Moore string matching algorithm. Consider a Text T= XYXZXXYXTZXYXZXXYXXYXXYY and Pattern P= XYXZXY.Find the pattern in the string.	**4+6**	**CO5**	K3	2.2.3
9	**Attempt ANY ONE part from the following**				
a)	What is an approximation algorithm? What is meant by P (n) approximation algorithms? Discuss approximation algorithm for Travelling Salesman Problem	**3+3+4**	**CO5**	K2	2.2.3
b)	Define NP hard and NP complete problems. What are the steps involved in proving a problem NP complete? Specify the problems are already proved to be NP complete	**3+3+4**	**CO5**	K2	2.2.4

CO Course Outcomes mapped with respective question

KL Bloom's knowledeg Level (K1, K2, K3, K4, K5, K6)

K1- Remember, K2- Understand, K3-Apply, K4- Analyze, K5: Evaluate, K6- Create

Solution ST-2

DAA - KCS 503

Section - A

Q1 (a) Answer is 99

There are 100 vertices numbered from 1 to 100

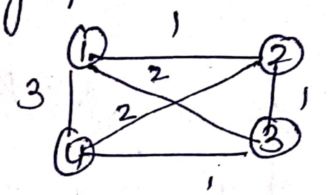
Edges are present in this graphs

$E = \{v_i, v_j \mid 1 \leq i_1, V_2), (V_2, V_3) \text{ -- } (V_{99}, V_{100})$

will have minimum weight i.e 1 for each edge.

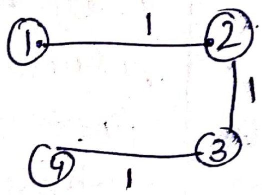
As, we know that, there are 99 edges are possible in MST of 100-vertices graph.

eg:-



graph

MST =

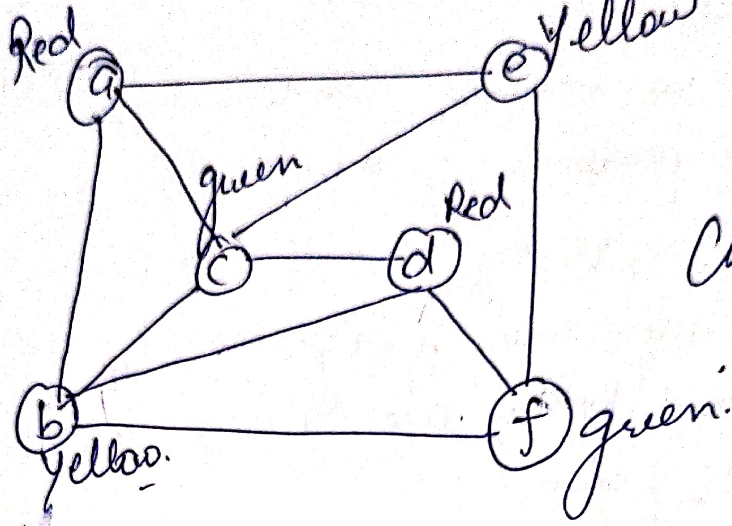


→ (2 Marks)

Q1(b) Given problem is subset-sum problem in which a set of non-negative integers & a value sum is given to determine, if there is a subset of the given set with sum equal to given sum. With recursion technique, time complexity of the problem is exponential. We can solve the problem in pseudopolynomial time using dynamic programming

→ (2 Marks)

Q1(c) Graph is planar, so chromatic number should be less than or equal to 4, can not be less than 3 because of odd length cycle. In other words, if graph is planar & has odd length cycle. then chromatic number can be either 3 or 4



Chromatic no = 3

→ 2 (Marks)

Q1(d)

P Problem	NP Problem
1. P Problem are set of problems which can be solved in polynomial time by deterministic algo	1. NP Problems are the problems which can be solved in non-deterministic polynomial time.
2. P Problems can be solved & verified in polynomial time.	2. Solution to NP Problems can not be obtained in polynomial time, but if the sol^n is given, it can be verified in polynomial time.
3. P Problems are subset of NP Problem	3. NP problems are super set of P Problem
4. Selection sort, Linear Search.	4. TSP, Knapsack problem.

(1 + 1) Marks
2 diff 2 diff

Q1(c) $\Theta(n)$ Calls are made to RANDOM in both cases. ③

Worst Case behaviour occurs when the partitioning produces one subproblem of size $n-1$ & one of size 0 each time it is called.

Recurrence Rel^n

$$T(n) = T(n-1) + T(0) + \Theta(1)$$ (1 Mark)

$T(0)$ = Randomized-Partition is not Called on subproblem of size '0'

$$\Rightarrow T(n) = T(n-1) + \Theta(1)$$

$$\therefore \Rightarrow \Theta(n)$$

Best Case : when partition is of size atmost $n/2$.

$$T(n) \leq 2T\left(\frac{n}{2}\right) + \Theta(1)$$

by Master Theorem.

$$\therefore T(n) = \Theta(n)$$

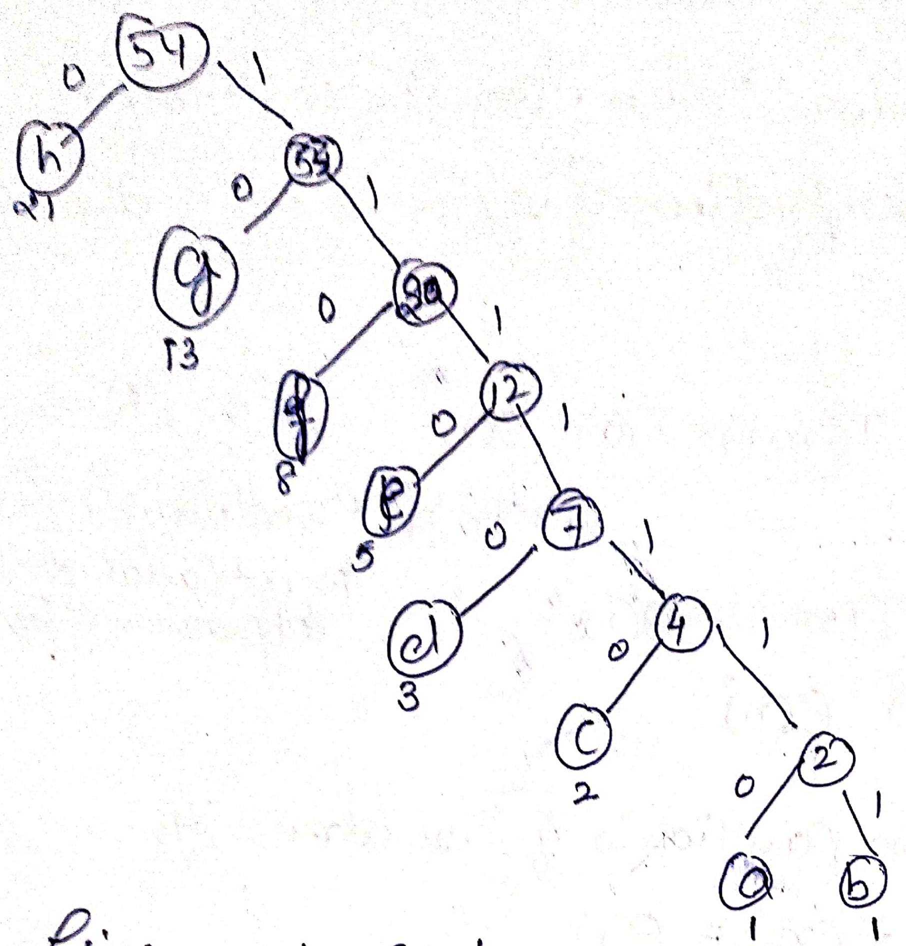
(1 Mark)

Section - B

Q2(a) a: 1, b: 1, c: 2, d: 3, e: 5, f: 8, g: 13, h: 21

1) Data is arranged in increasing order.
2) The Two least frequent characters are taken as the Children of newly made node & the frequency of the newly made node is equal to the sum of those two child nodes. Repeat the same procedure, till all nodes are finished.

Scanned with OKEN Scanner



(4 Mark)

Given string can be written as.

a = 1111110

b = 1111111

c = 111110

d = 11110

e = 1110

f = 110

g = 10

h = 0

(1 Mark)

Scanned with OKEN Scanner

Q2(b)

	W	V	P=V/W
I_1	5	30	6.0
I_2	10	20	2.0
I_3	20	100	5.0
I_4	30	90	3.0
I_5	40	160	4.0

Now arrange it in decreasing order

	W	V	P
I_1	5	30	6.0
I_3	20	100	5.0
I_5	40	160	4.0
I_4	30	90	3.0
I_2	10	20	2.0

(3 Mark)

Total Weight, W = 60

Now, fill the knapsack, acc. to decreasing value of P_i

$$\text{Max. Value} = 30 \times \frac{5}{5} + 100 \times \frac{20}{20} + 160 \times \frac{35}{40}$$

$$= 30 + 100 + 140$$

$$= 270 \text{ (min. Cost)}$$

$W = 5 \times \frac{5}{5} + 20 \times \frac{20}{20} + 40 \times \frac{35}{40}$

$\Rightarrow 5 + 20 + 35 = 60$ (2 Marks)

Q3 (a) Order of $M_1 = \omega * x$

order of $M_2 = x * y$

order of $M_3 = y * z$

for Cost of $(M_1 M_2) M_3 = \omega x y + \omega y z$

$\rightarrow$ Cost of $M_1 M_2 = \omega * x * y$

$\rightarrow$ new matrix = M

$\rightarrow$ Order of $M = \omega * y$

$\rightarrow$ Cost of $M M_3 = \omega * y * z$

$\rightarrow$ Total Cost = $M_1 M_2$ Cost + $M M_3$ Cost

Illy Cost of $M_1 (M_2 M_3) = xyz + \omega x z$

for $(M_1 M_2) M_3$ to take less time than $M_1 (M_2 M_3)$

$(\omega x y + \omega y z) < (xyz) + \omega x z$

Dividing both sides of above eqn.

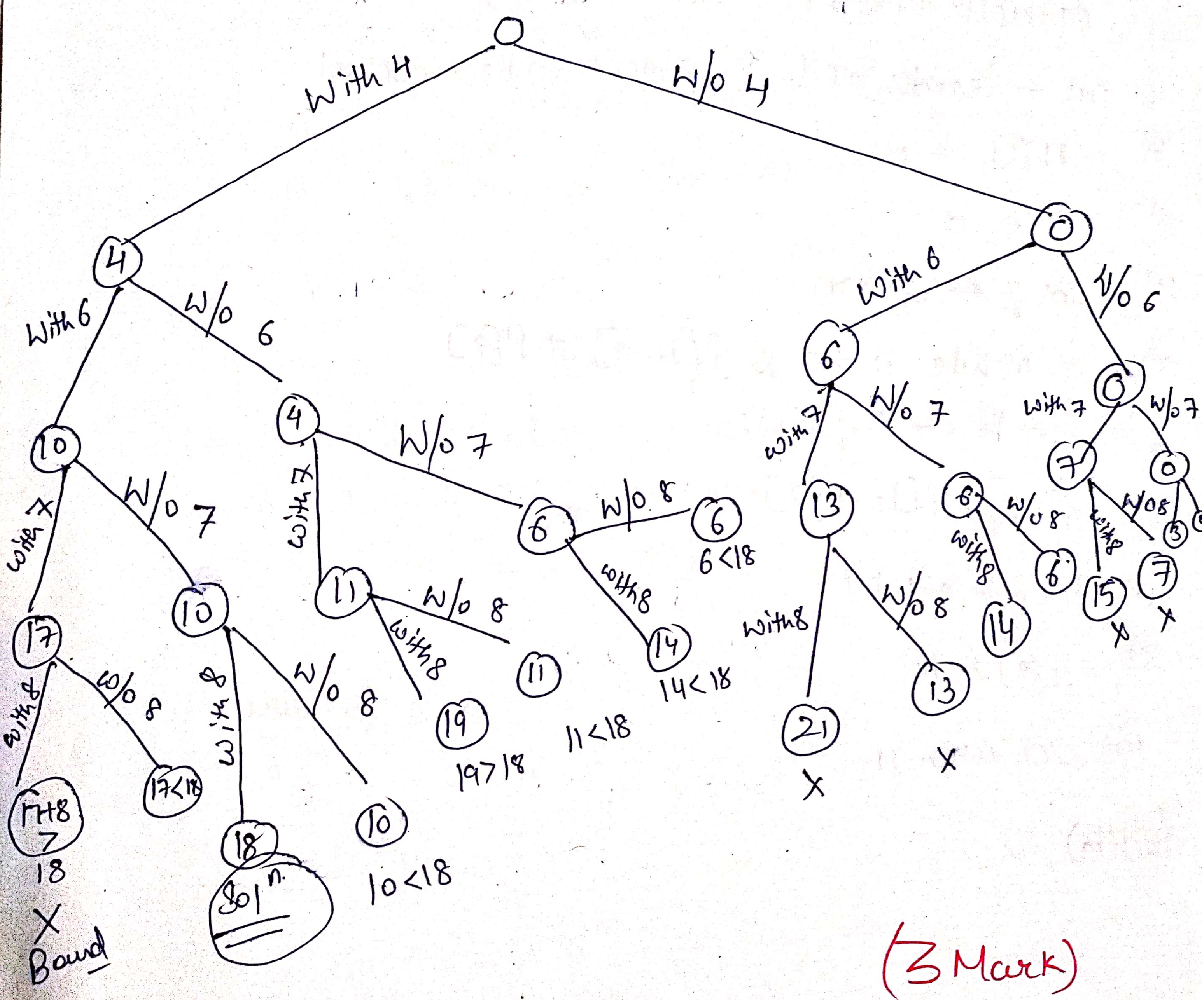
$\left(\frac{1}{x} + \frac{1}{z}\right) < \left(\frac{1}{\omega} + \frac{1}{y}\right)$

(5 Mark)

Q3(b) Subset-Sum problem : is finding a subset of given set S = {s1, s2 -- sn} of n positive integers whose sum is equal to a given positive integer d.

(2 Marks)

Let S = {4, 6, 7, 8}, m = 18



(3 Mark)

Q4(a) Prefix funcⁿ 'π' for the pattern.

$P = a\ b\ a\ c\ a\ b$

1	2	3	4	5	6
a	b	a	c	a	b
0	0	1	0	1	2

(2 Mark)

KMP Algo for Computing prefix funcⁿ.

COMPUTE-PREFIX-FUNCTION(P)

1) $m \leftarrow$ length[P] // 'P' pattern to be matched
2) $\pi[1] \leftarrow 0$
3) $k \leftarrow 0$
4) for $q \leftarrow 2$ to m
5) do while $k > 0$ & $P[k+1] \neq P[q]$
6) do $k \leftarrow \pi[k]$
7) if $P[k+1] = P[q]$
8) then $k \leftarrow k+1$
9) $\pi[q] \leftarrow k$
10) return π

(3 Mark)

~~Q4(b)~~

Q4(b) To determine Running time of procedure RECURSIVE-FFT, we know that exclusive of the recursive calls, each invocation takes time $\theta(n)$

n = length of input vector.

Recurrence for the running time is

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

$$a = 2, \; b = 2$$

by Master Theorem.

$$n^{\log_b a} = n^{\log_2 2}, \Rightarrow n^1$$

$$f(n) = n^{\log_b a} = n$$

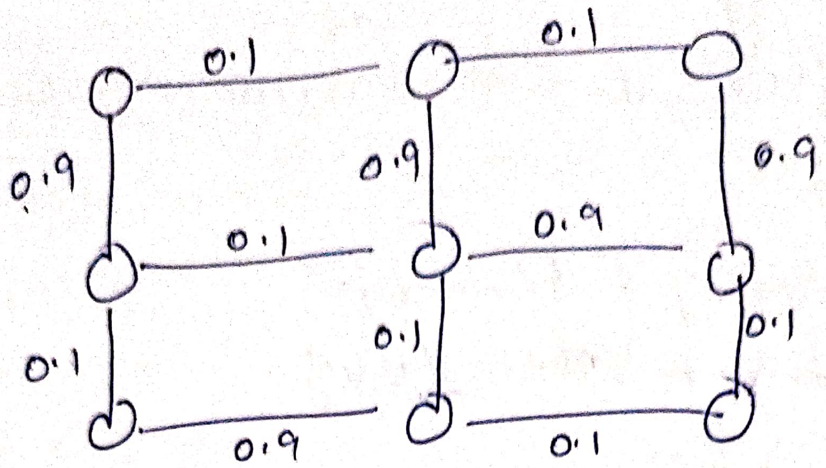
$$\therefore \; \Theta(n \log n)$$

So, we can evaluate a polynomial of degree-bound 'n' at the complex n^{th} root of unity in time $\Theta(n \log n)$

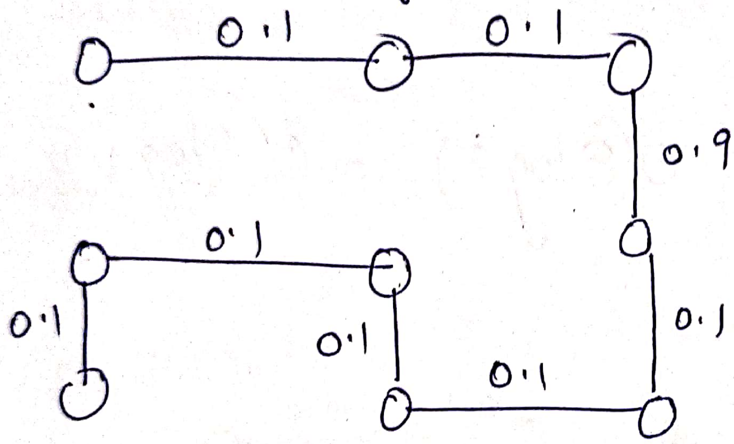
(5 Mark)

Section - C

Q5 (a)



Acc. to Kruskal Algo.



We, will include minimum weight edges first if there is no cycle. resultant

But we need only one edge to form spanning tree & we have 3 options for one edge

$\therefore$ MST = 3 (5 Marks)

Algo. Kruskal.

MST - KRUSKAL (G, W)

1) $A \leftarrow \phi$

2) for each vertex v _ V[G]

3) do Make - set [v]

(11)

(4) Sort the edges of E in to non-decreasing order by weight w.

(5) for each edge (u,v) - E taken in non-decreasing order by weight

(6) do if find-set[u] ≠ find-set[v]

(7) then A ← A Union {(u,v)}

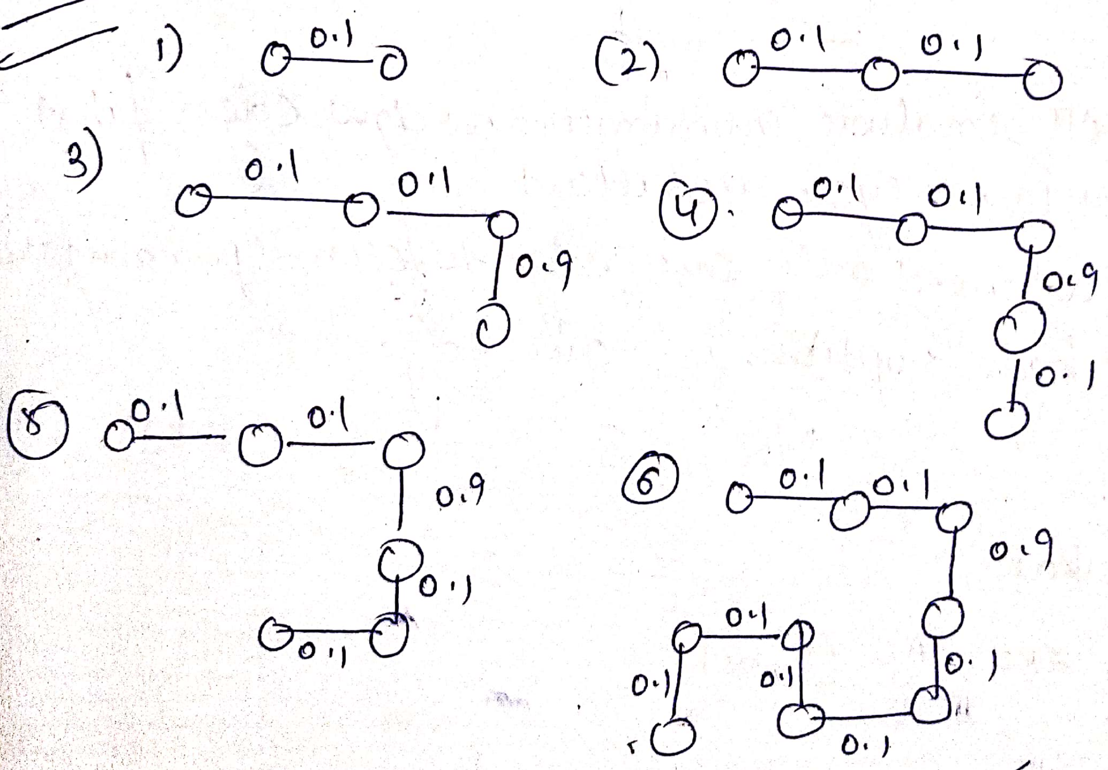
(8) UNION(u,v)

(9) return A

(3 Mark)

Complexity $= O(E \log E) = O(E \log V)$ (2 Mark)

Or Prim's.



(5 Mark)

MST-PRIM (G, w, r):

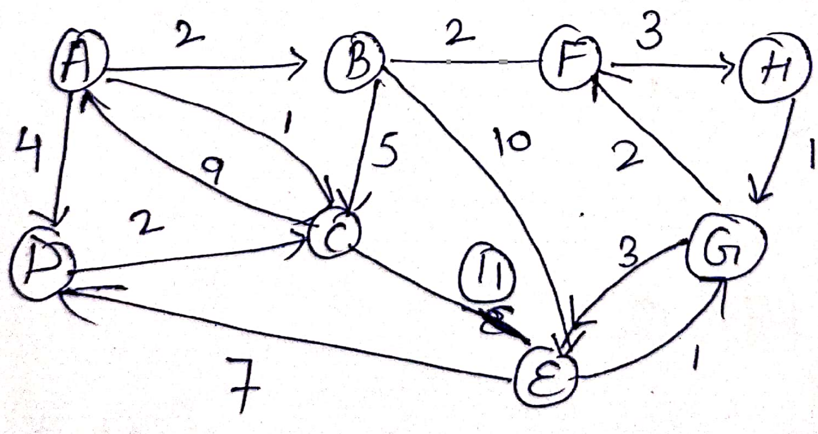
1) for each $u \in V[G]$
2) do $Key[u] \leftarrow \infty$
3) $\pi[u] \leftarrow NIL$
4) $Key[r] \leftarrow 0$
5) $Q \leftarrow V[G]$
6) while $Q \neq \emptyset$
7) do $u \leftarrow$ Extract-MIN (Q) 8) for each $v \in Adj[u]$
9) do if $v \in Q$ & $w(u,v) < Key[v]$
10) then $\pi[v] \leftarrow u$
11) $Key[v] \leftarrow w[u,v]$

(3 Mark)

$$\text{Complexity} = O(V\log V + E\log V)$$
$$= O(E\log V)$$

(2 Mark)

Q 5(b)



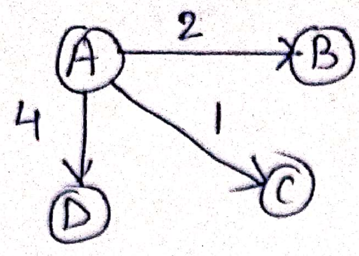
unvisitedset = {A, B, C, D, E, F, G, H}

$d[A] = 0$

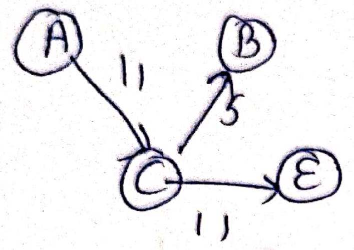
$d[B] = d[C] = d[D] = d[E] = d[F] = d[G] = d[H] = \infty$

before edge relaxation.

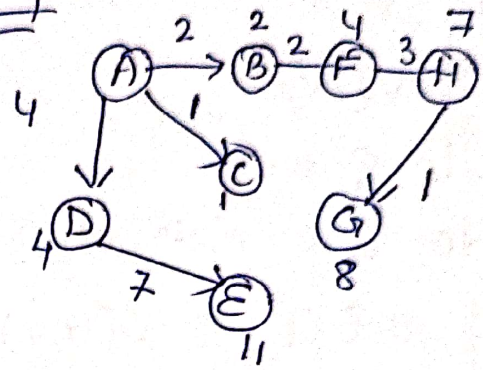
Step 1:-



Step 2:-



Step 3:-

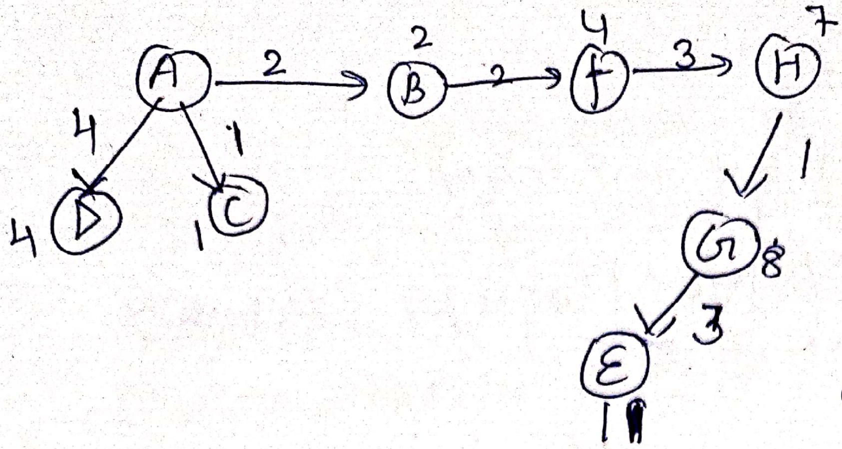


Algo Dijkstra

Dijkstra Algo (G, w, s)

1. for all the vertices $v \in V(G)$
   $d[v] \leftarrow \infty$
   $\pi[v] \leftarrow NIL$
2) $d[s] \leftarrow 0$
3) $S \leftarrow \phi$
4) $Q \leftarrow V[G]$
5) while $(Q \neq \phi)$
   $u \leftarrow$ Extract-Min Q
   $S \leftarrow S \cup \{u\}$
6) for all the vertices $v \in adj[u]$
   if $d[v] > d[u] + w[u,v]$
   $d[v] \leftarrow d[u] + w[u,v]$
   $\pi[v] \leftarrow u$

} Relax operation. (3 Mark)



Other soln

Matrix

	B	C	D	E	F	G	H
0	∞	∞	∞	∞	∞	∞	∞
A	2	1	4	∞	∞	∞	∞
A, C, B	2	1	4	11	∞	∞	∞
A, C, B, F	2	1	4	11	2	∞	∞
A, C, B, F, H	2	1	4	10	2	∞	3
A, C, B, F, H, E	2	1	4	10	2	1	3
	2	4	4	3	2	1	3

(5 Mark)

Complexity =
$= V\log V + E\log V$
$= O(|V+E|\log V)$ (2 Mark)

Q6 (a) Max Wt = 13 kg
Total item = 4.

	Item	Value	Weight
1	Mirror	28	7
2	Painting	18	5
3	Vase	23	6
4	earphone	9	2

w_i p_i	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7 28 1	0	0	0	0	0	0	0	28	28	28	28	28	28	28
5 18 2	0	0	0	0	0	18	18	18	18	18	18	18	46	46
6 23 3	0	0	0	0	0	18	23	23	23	23	~~23~~ 31	31	31	51
2 9 4	0	0	9	9	9	18	23	23	32	23	23	31	31	50

(8 Mark)

(P_i, w_i)
$(51, 13)$ 3
$[(51-23),(13-6)]$
$[28, 7] \in 1$
$\Rightarrow ((28-28),(7-7))$
$= (0, 0)$

So the item 1 & 3 is chosen by thief
$[x_1 \quad x_2 \quad x_3 \quad x_4]$
$[1 \quad 0 \quad 1 \quad 0]$

(2 Mark)

Q6(b) Let $c[i,j]$ be the length of LCS of the sequence X_i & Y_j. If either $i=0$ & $j=0$, one of the sequences has length 0, so the LCS has length 0. Optimal substructure of the LCS problem using Recurrence formula.

$$c[i,j] = \begin{cases} 0 & \text{if } i=0 \text{ or } j=0 \\ c[i-1,j-1]+1 & \text{if } i,j>0 \ \& \ x_i = y_j \\ \max(c[i,j-1], c[i-1,j]) & \text{if } i,j>0 \ \& \ x_i \neq y_j \end{cases}$$

(2 Mark)

$X = \{7, 12, 15, 14, 21, 13, 14, 11, 10, 9\}$

$Y = \{20, 7, 15, 1, 14, 3, 6, 13, 11, 18, 10, 9\}$

	0	20	7	15	1	14	3	6	13	11	18	10	9
0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1↑	←1	←1	←1	←1	←1	←1	←1	←1	←1	←1
12	0	0	1↑	1↑	1↑	1↑	1↑	1↑	1↑	1↑	1↑	1↑	1↑
15	0	0	1↑	2↖	←2	←2	←2	←2	←2	←2	←2	←2	←2
14	0	0	1↑	2↑	2↑	3↖	←3	←3	←3	←3	←3	←3	←3
21	0	0	1↑	2↑	2↑	3↑	3↑	3↑	3↑	3↑	3↑	3↑	3↑
13	0	0	1↑	2↑	2↑	3↑	3↑	4↖	←4	←4	←4	←4	←4
14	0	0	1↑	2↑	2↑	3↑	3↑	3↑	4↑	4↑	4↑	4↑	4↑
11	0	0	1↑	2↑	2↑	3↑	3↑	3↑	4↑	5↖	←5	←5	←5
10	0	0	1↑	2↑	2↑	3↑	3↑	3↑	4↑	5↑	5↑	6↖	←6
9	0	0	1↑	2↑	2↑	3	3	3	4	5	5	6	7↖

(6 Mark)

PRINT-LCS (b, x, i, j)

1) if i = 0 or j = 0
2) then return
3) if b[i, j] = '↖'
4) then print-LCS (b, x, i-1, j-1)
5) Print x_i
6) else if b[i, j] = '↑'
7) then PRINT-LCS (b, x, i-1, j)
8) else PRINT-LSS (b, x, i, j-1)

(2 Mark)

Sol^n is 7, 15, 14, 13, 11, 10, 9

Q7 (a) sequence of dimension
{10, 5, 3, 12, 6}

A1 = 10×5, A2 = 5×3, A3 = 3×12

	A1	A2	A3	A4
A1	0	150	510	546
A2		0	180	306
A3			0	216
A4				0

	1	2	3	4
1	0	1	2	2
2		0	2	2
3			0	3
4				0

$m[1,2] = m[i,k] + m[k+1,j] + P_{i-1} P_k P_j$ where 'k' must be $k \geq i$, $k < j$

$k=1 \quad = m[1,1] + m[2,2] + P_0 P_1 P_2$

$= 0 + 0 + 10 \times 5 \times 3$

$= 150$

$m[2,3] = m[2,2] + m[3,3] + P_1 P_2 P_3$

$k=2 \quad = 0 + 0 + 5 \times 3 \times 12$

$= 180$

$m[3,4] = m[3,3] + m[4,4] + P_2 P_3 P_4$

$= 0 + 0 + 3 \times 12 * 6$

$= 216$

$m[1,3]$ then $k = 1$ & 2

for $k=1$

$m[1,3] = m[1,1] + m[2,3] + P_0 P_1 P_3$

$= 0 + 180 + 10 \times 5 \times 12$

$= 180 + 600$

$= 780$

for $k=2$

$m[1,3] = m[1,2] + m[3,3] + P_0 P_2 P_3$

$= 150 + 0 + 10 \times 3 \times 12$

$= 150 + 360$

$= 510$

for $k=2$

$m[2,4] = m[2,2] + m[3,4] + P_1 P_2 P_4$

$= 180 + 0 + 5 \times 12 \times 6$

$= 180 + 360$

$= 540$

$m[1,4]$ then $k = 1, 2, 3$

for $k = 1$

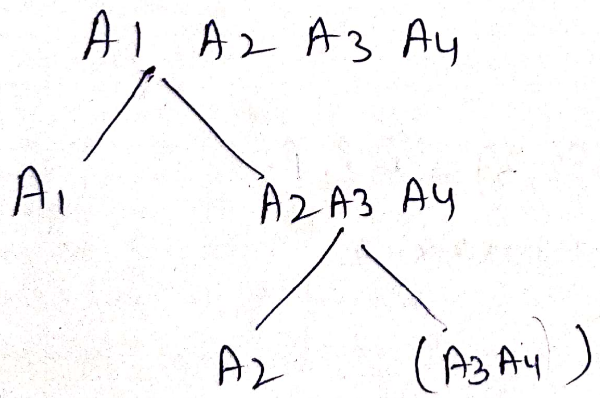
$$m[1,4] = m[1,1] + m[2,4] + P_0 P_1 P_4$$
$$= 0 + 306 + 10 \times 5 \times 6$$
$$= 306 + 300$$
$$= 606$$

$k = 2$

$$m[1,4] = m[1,2] + m[3,4] + P_0 P_2 P_4$$
$$= 150 + 216 + 10 \times 3 \times 6$$
$$= 150 + 216 + 180$$
$$= 546$$

$k = 3$

$$m[1,4] = m[1,3] + m[4,4] + P_0 P_3 P_4$$
$$= 510 + 0 + 10 \times 12 \times 6$$
$$= 510 + 720$$
$$= 1230$$

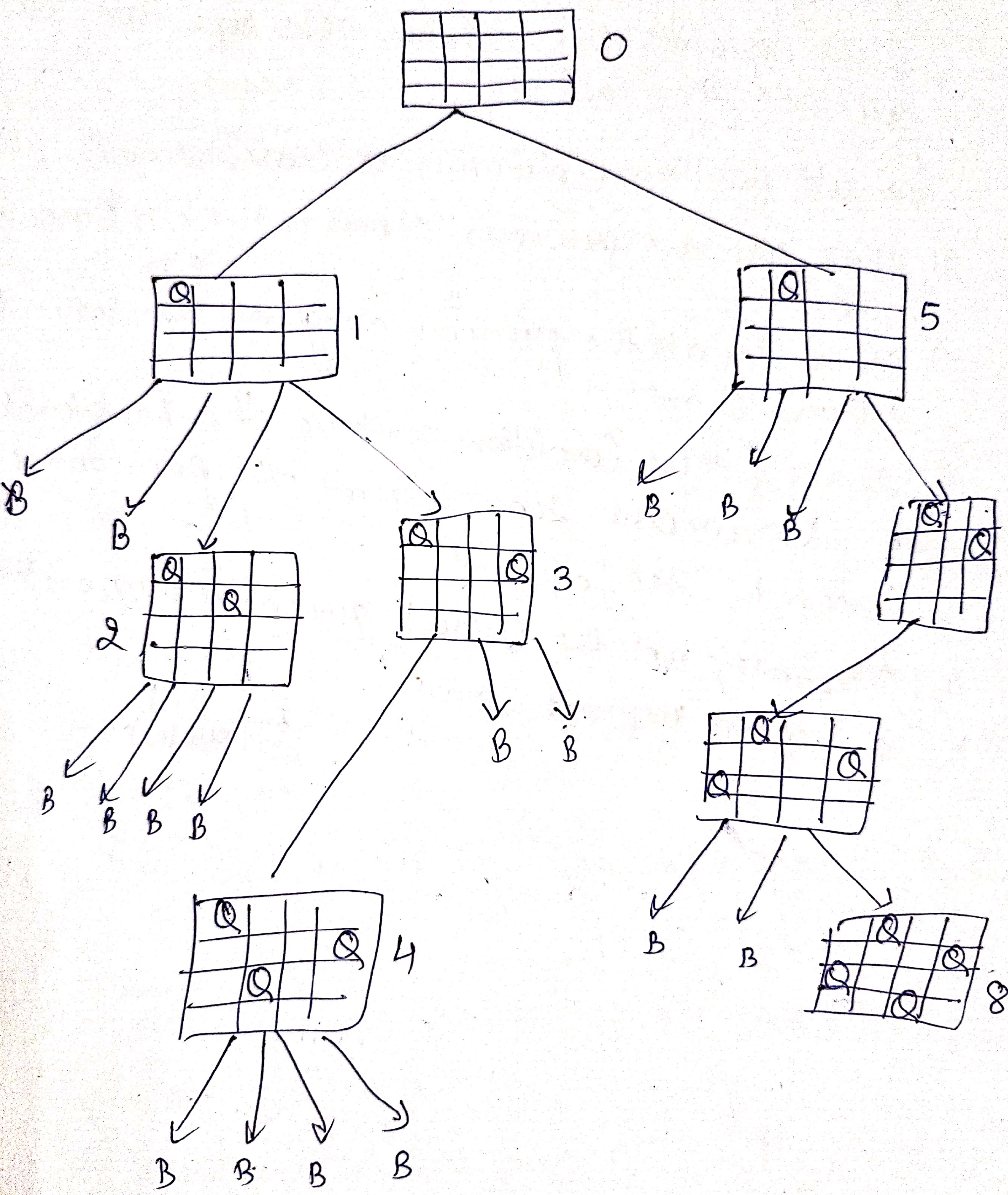


final solⁿ = $A_1(A_2(A_3 A_4))$

(10 Mark)

20

Q7(b) State space of 4-Queen



(6 Mark)

Simple logic of generating the State Space tree is to keep exploring all possible solutions using backtracking & stop exploring that solution wherever two Queens are attacking each other

for all positions (Columns) in current row:-

1) Check all the previous rows is there a Queen or not?
2) Check for all the previous diagonal Columns is there a Queen or not?
3) If any of these Conditions are true, then backtrack to the previous row & move the previous Queen 1 step forward
4) Otherwise, put the current queen in the position & move to the next row.

(4 Mark)

Q8(a) RABIN-KARP-MATCHER (T, P, d, q)

```
n ← length [T]
m ← length [P]
h ← d^(m-1) mod q
p ← 0
t0 ← 0
for i ← 1 to m
do p ← (dp + P[i]) mod q
t0 ← (dt0 + T[i] mod q
for s ← 0 to n-m.
do if P = ts
then if P[1..m] = T[s+1 ... s+m]
then "Pattern occurs with shift" s
if s < n-m
then ts+1 ← (d(ts - T[s+1] h) + T[s+m+1] mod q
```

(4 Mark)

Sol^n.

Text T = 31415926535 89793.

Pattern P = 26

Working Modulo $q = 11$

Calculate $P \bmod q = 26 \bmod 11 = 4$

Now, find the exact match of $P \bmod q$ in the given text

T =

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

P =

2	6

1st window has value 31, then the new value for the window is calculated by dropping the higher order digit i.e 3, shifting left (multiplying by 10) & then adding on the lower order digit i.e 4 gives us the new value 14 i.e.

$2 \times 10 + 6$
$\Rightarrow 20 + 6 \Rightarrow 26$

$3 \times 10 + 1$
$30 + 1$
$\Rightarrow 31 \bmod 11$

3	1	4

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

9	3	8	4	4	4	4	10	9	2	3	1	9	2	5

Spurious Hit

$m = 2$

$$t_{s+1} = 10(31 - 10^{2-1} \cdot 3) + 4$$
$$= 10(31 - 10 \cdot 3) + 4$$
$$= 10(1) + 4$$
$$\Rightarrow 14$$

Total no. of spurious Hit = 3

(5 Marks)

Q8(b) Boyer-Moore string Matching (T[0...n], P[0...m])

1) n ← length[T]
2) m ← length[P]
3) i ← m-1
4) j ← m-1
5) while i<n
6) do if P[j] = T[i]
7) then if j=0
8) then return i
9) else i ← i-1
10) j ← j-1
11) else i ← i+m - min (j, 1+last[T[i]])
12) j ← m-1
13) return -1

(4 Mark)

Bad Match Table

X	Y	Z	*
1	4	2	6

Text = XYXZXXYXTZXYXZXXYXXYXXYY

P = XYXZXY

(6 Mark)

Consider the pattern XYXZXY

Pattern	X	Y	X	Z	X	Y
Index	0	1	2	3	4	5

... Table

p1:-

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23.
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y
Pattern	X	Y	X	Z	X	Y																		

Step2:-   Shift 1.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y.
pattern		X	Y	X	Z	X	Y																	

Shift 1.

Step3:-

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23.
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y.
Pattern.			X	Y	X	Z	X	Y																

Shift 1.

Step4:-

index.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y.
Pattern.				X	Y	X	Z	X	Y															

Shift 6

Step5:-

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23.
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y.
Pattern.										X	Y	X	Z	X	Y									

Shift 1.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	~~X~~	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y
Pattern											X	Y	X	Z	X	Y								

Shift = 1.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y
Pattern												X	Y	X	Z	X	Y							

Shift = 1.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y.
Patter													X	Y	X	Z	X	Y.						

Shift = 1

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y
Pattern														X	Y	X	Z	X	Y					

Shift = 1.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23.
text	X	Y	X	Z	X	X	Y	X	T	Z	X	Y	X	Z	X	X	Y	X	X	Y	X	X	Y	Y
Pattern															X	Y	X	Z	X	Y				

and so on and there is no match.

Q9(a) **Approximation Algorithms** : Is a way of approach NP-Completeness for the optimization problem. goal of Approximation algorithm is to Come as close as possible to the optimum Value in a reasonable amount of time which is at the most polynomial time.

(3 Marks)

We say the approximate algo has an approximate ratio $P(n)$ for an input size 'n' where

$$\max\left(\frac{C}{C^*}, \frac{C^*}{C}\right) \leq P(n)$$

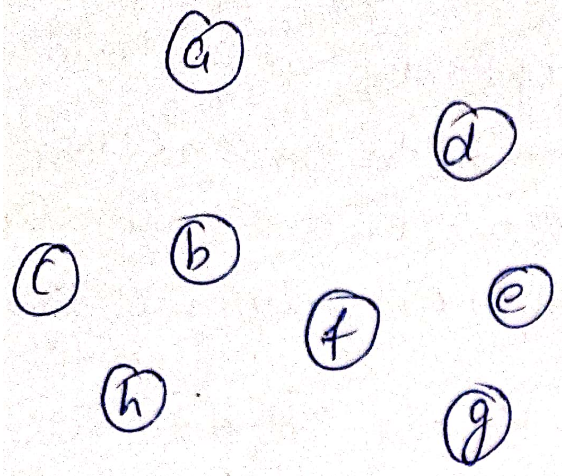
$$P(n) \geq 1$$

Some problems have polynomial-time approximation algorithm with small constant approximat ratios, while Others have best-known polynomial time approximation algorithm whose approximate ratio grow with n.

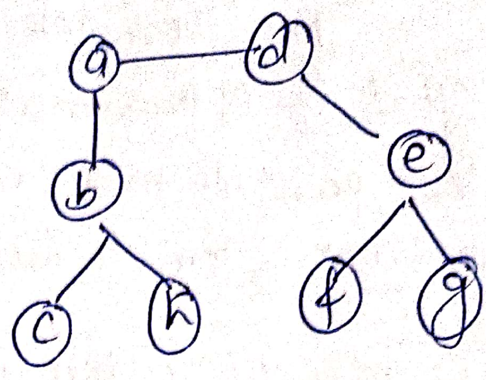
(3 Marks)

TSP

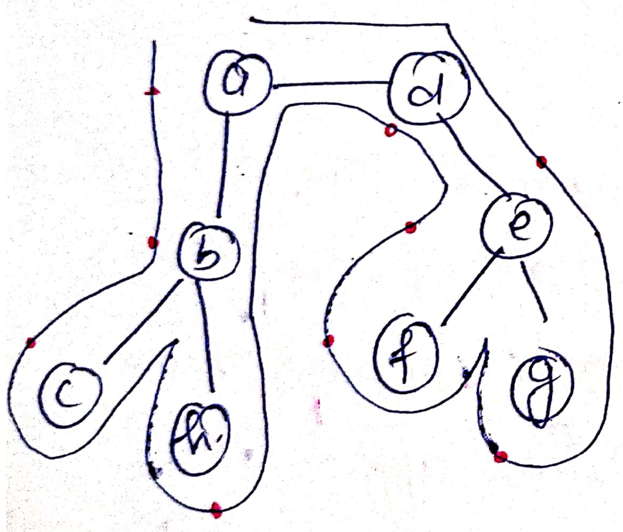
1) Compute a MST of graph G
2) Select any vertex 'r' is the root of the tree
3) Let L be the list of vertices visited in pre order tree walk of T;
4) Return the Hamiltonian cycle 'H' that visits the vertices in the Order L



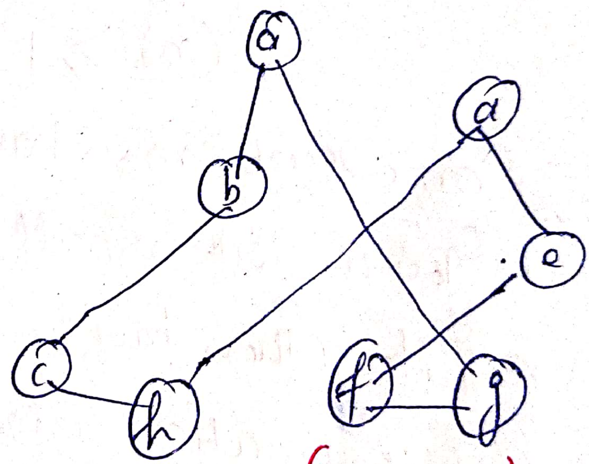
1) given set of points



② MST



3) full tree walk on T



(4 Mark)

④ A pre order sequence gives a tour H.

Intuitively, approx TSP first makes a full walk of MST, T which visits each edge exactly two times. To create a Hamiltonian cycle from the full walk, it bypasses some vertices (which corresponds to shortcut)

Q9(b)

NP-Hard :- If we can solve this problem in polynomial time, then we can solve all NP-problems in polynomial time. If you convert the issue in to one form to another form within the polynomial time

NP Complete :- A problem is in NP Complete, if it is NP & it is NP-Hard

(3 Mark)

Algo to prove that a problem is NP Complete.

NP Complete problems are the ones that are both in NP & NP-Hard,

So to prove is NP-Complete, we need to show that the problem

1) belongs to NP
2) is NP Hard

(3 Mark)

Problem are:-

1) Clique
2) Vertex-Cover
3) Subset-sum.

(4 Mark)

1) Clique : 1. clique

2. $3CNF \le P$ Clique

3. Clique $\le P$ $3CNF \le SAT$

4) Clique $\in$ NP

2) Vertex Cover

→ Vertex cover defination
→ Vertex Cover $\leq_P$ Clique
→ Clique $\leq_P$ vertex Cover
→ Vertex Cover $\in$ NP

3) Subset Cover

→ Subset Cover
→ Vertex cover $\leq_P$ Subset Cover
→ Subset Cover $\leq_P$ Vertex Cover
→ Subset Cover $\in$ NP.