Time Complexity

① for(i= 0; i<n; i++) { → n
  a++;
}
Time complexity: O(n)

② for (i = 0; i<n; i++){ → n
  for (j = 0; j<n; j++){ → n+1
    a++; n(n+1)
  }
}
Time complexity: $O(n^2)$

(3) p = 0;
for(i=1; p<=n; i++)
{
p = p + i;
}

$p <= n$
$p > n$
$k^2 > n$
$k = \sqrt{n}$

The loop will stop when $p > n$

If $p = 0$,
$p = 0 + 1 = 1$; $p = 1 + 2 = 3$; $p = 3 + 3 = 6$
$1 + 2 + 3 \cdots k$

$$\frac{k(k+1)}{2} > n$$

$k^2 > n$
$k = \sqrt{n}$
$O(\sqrt{n})$

p=0	p=p+i = 0
1	0+1 = 1
2	= 1+2
3	= 1+2+3

$1+2+3 \cdots k$
$= \frac{k(k+1)}{2}$

$p = \frac{k^2 + k}{2}$

(4) for(i=1; i<n; i=i*2)
{
statement;
}

$i = 1 \quad i = 1 * 2 = 2$
$i = 2 \quad i = 2 * 2 = 2^2$
$i = 3 \quad i = 2 * 2 * 2 = 2^3$
$\vdots$
$i = k \quad i = 2^k$
$2^k = n$
$k = \log_2 n$

(5) for (i = n; i >= 1; i = i/2)
{
statement;
}

$i = 1 \quad i = 1/2$

$\frac{n}{2^1}$

$\frac{i}{2^2}$

$\frac{1}{2}$

$\frac{i}{2^k}$

$n = 2^k$

$\boxed{k = \log_2 n}$

(6) for (i = 0; i*i < n; i++)
{
statement;
}

$i^2 > n$

$i = \sqrt{n}$

$= O(\sqrt{n})$

(7) p = 0;
for(i = 1; i < n; i = i*2)
{
p++;
}
for(j = 1; j < p; j = j*2)
{
statement;
}

$p = \log n$

$\log p$

$\log(\log n)$

$i = 1 \quad i = 1*2 = 2$

$i = 2 \quad i = 2*2 = 2^2$

$i = 3 \quad i = 2*2*2 = 2^3$

$i = n \quad i = 2^n$

$2^n > p \qquad p = \log n$

(8)
```
for(i=0; i<n; i++)                 ← n
{
    for(j=1; j<n; j=j*2)           ← n* log(n)
    {
        stmt;
    }
}
```

$O(n \log n)$

(9)
```
Algo Mult(A, B, n)
{
    for(i=0; i<n; i++)                (n+1)
    {
        for(j=0; j<n; j++)            n*(n+1)
        {
            stmt; ————                n*n
            for(k=0; k<n; k++) ←      n*n*(n+1)
            {
                stmt; ——              n*n*n
            }
        }
    }
}
```

$2n^3 + 3n^2 + 2n + 1$

$= O(n^3)$

(10)
```
void function(int n)
{
    int count = 0;
    for(int i = n/2; i<=n; i++)
    {
        for(int j = 1; j<=n; j = 2*j)
        {
            for(int k = 1; k <= n; k = k*2)
            {
                count++;
            }
        }
    }
}
```

```
void function (int n)
{
    int count = 0
    for (int i = n/2; i<=n; i++)
    for (int j = 1; j<=n; j = 2*j)
        for (int k = 1; k<=n; k = k*2)
            count++;
}
```

Divide and Conquer

Q. Explain the need of asymptotic notation

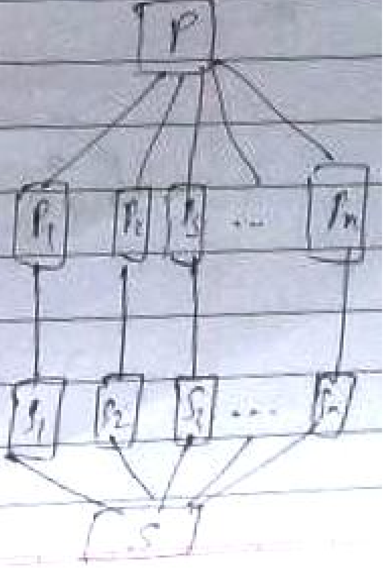
Q. Relation between Big O, Ω, Θ

1) Binary Search
2) Max Min Problem
3) Quick Sort
4) Merge Sort

Weightage for Chapter 1 : 10 Marks
Questions on Asymptotic notation and function growth

— 11 — 2 : 20 Marks
Questions on writing ~~sort~~ algorithms

Control Abstraction



Type DAndC (Problem P)
{
if small (P) return S(P);
else {
divide P into smaller unit

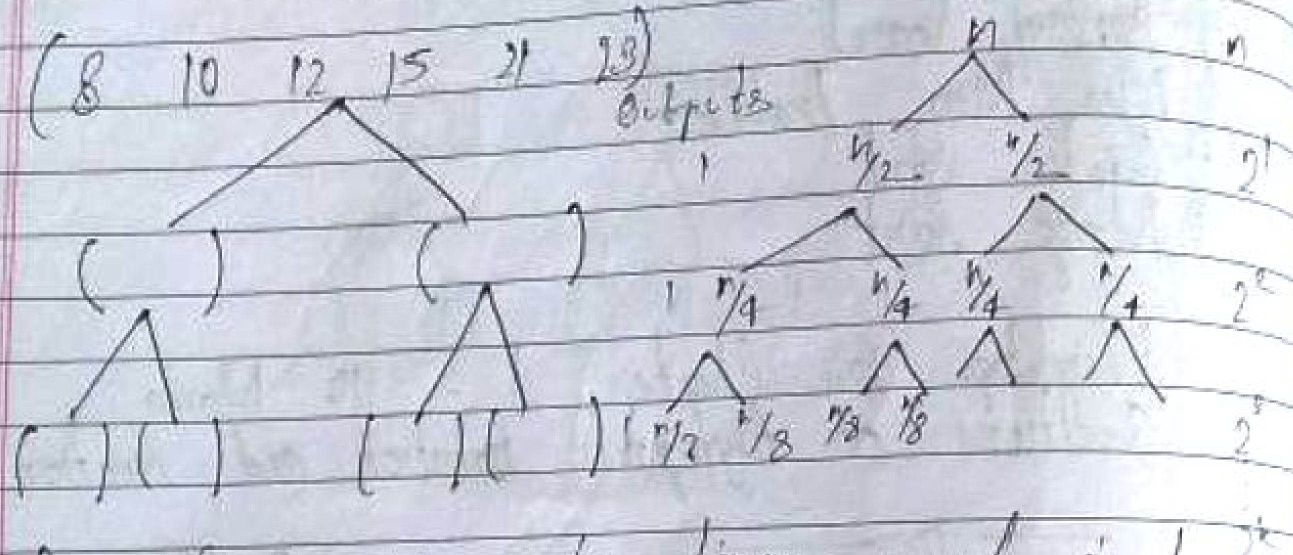
General Form

$T(n)$: Time taken to solve a ~~pro~~ problem with n inputs

a : number of subproblems

b : size of each subproblem

$$T(n) = \begin{cases} T(1) & n = 1 \\ aT(n/b) + f(n) & n > 1 \end{cases}$$



(*) Size of subproblem for binary search is 1. 2^k

$f(n)$: The function which will take n inputs (number of outputs generated after performing divide and conquer)

$n = 2^k$

$k = \log_2 n$

$1 \times \log n = \log_2 n$

Finding Maximum and Minimum

Algorithm StraightMaxMin(a, n, max, min)

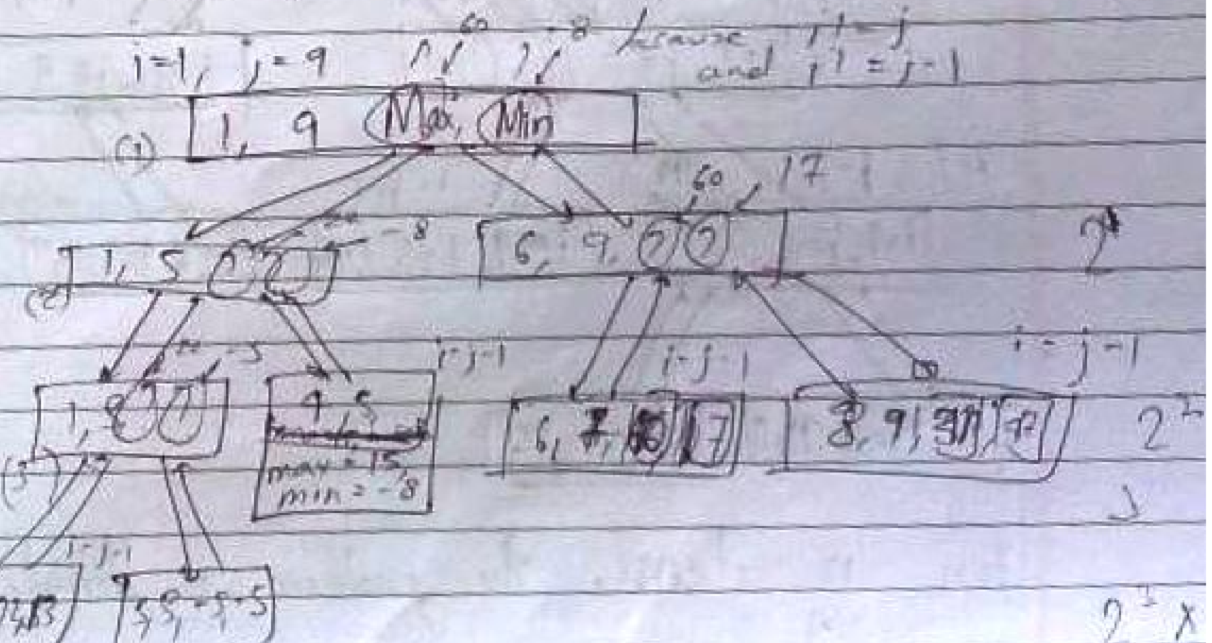
List
n elements

min, max

[i mid][Mid+1 j]

1) $n=1$ $i=j$ }
2) $n=2$ $i=j-1$ } ——— ~~(divi~~
3) $n>2$. $mid = (i+j)/2$ (divide into two)

Ex.: Find max and min in the array:
22, 13, -5, -8, 15, 60, 17, 31, 47 ($n=9$)



~~$T(n) = 2.T(n/2) + 2$~~ $n = 2^k$

$2 \times (n-1) = 2n-2 \therefore O(n)$

If $n=1$, $T(n) =$

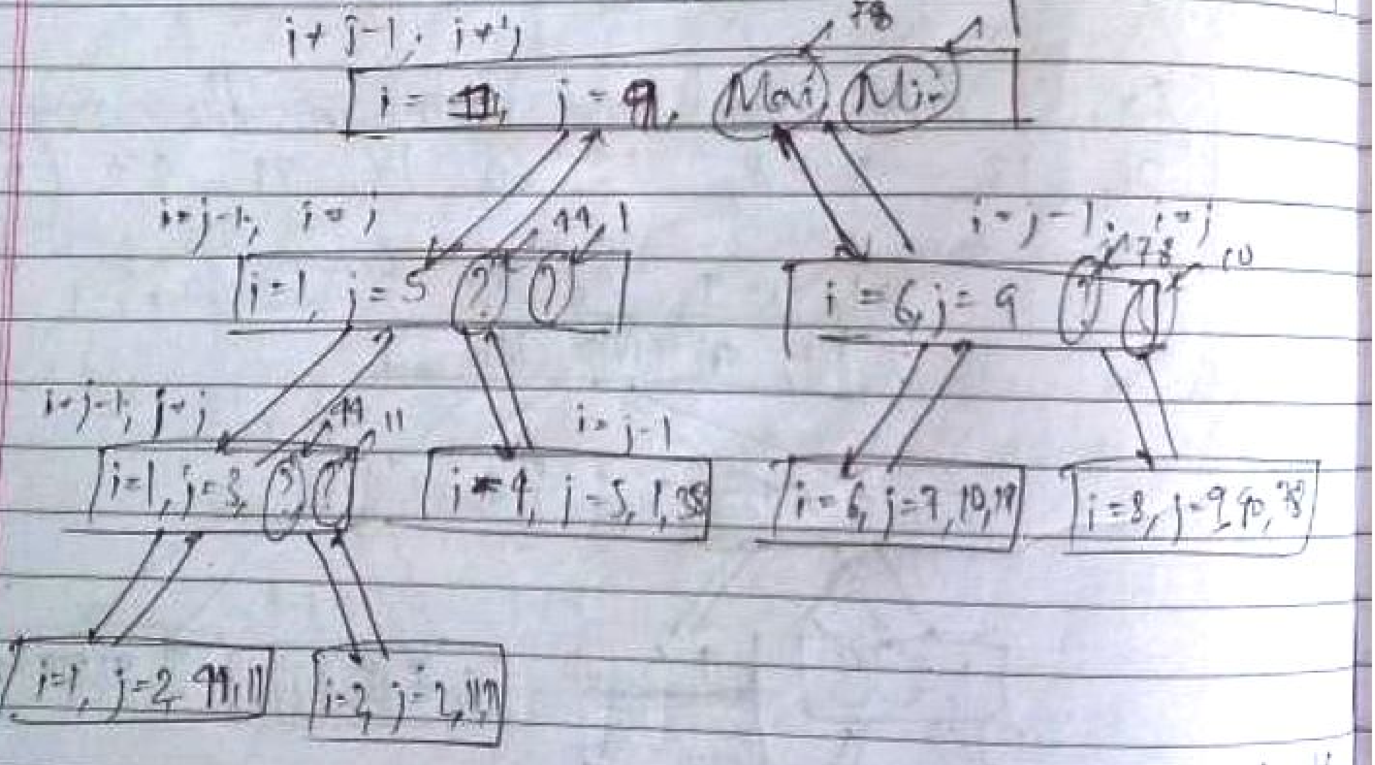
If $n=2$, $T(n) =$

If $n>2$, $T(n) =$

$$T(n) = \begin{cases} T(1) & n=1 \\ T(2) & n=2 \\ 2.T(n/2) + 2 & n>2 \end{cases}$$

i. Explain MaxMin as divide and conquer method
ii. For the given input, draw the divide and conquer tree diagram to find minimum and maximum of the given input: 44, 11, 56, 1, 38, 10, 17, 90, 78

1 2 3 4 5 6 7 8 9



For n items, the problem is divided $(n-1)$ subproblem times.
9 8

iii. Also compute complexity of the algorithm

Recurrence Equation

$T(n) = a.T(n/b) + f(n)$

Binary Search: $T(n) = 1.T(n/2) + 1 \rightarrow O(\log n)$

Min Max : $T(n) = 2T(n/2) + 2 \rightarrow O(n)$

Merge Sort : $T(n) = 2T(n/2) + n \rightarrow O(n \log n)$

Quicksort

10 6 7 8 9 2 3 11 12

Step 1: Take a pivot ~~ele~~ element.

6 7 8 9 2 3 10 11 12

Partitioning

Algorithm Partition(a, m, p)
// Within a[m], a[m+1], ..., a[p-1], the elements are
// rearranged in such a manner that if initially t = a[m],
// then after completion a[q] = t for some q between m
// and p-1, a[k] ≤ t for m ≤ k < q and a[k] ≥ t
// for q < k < p, q is returned. Set a[p] = ∞.
{
    v := a[m]; i := m; j := p;
    repeat
    {
        repeat
            i := i+1;
        until (a[i] ≥ v);
        repeat
            j := j-1;
        until (a[j] ≤ v);
        if (i < j) then Interchange (a, i, j);
    } until (i ≥ j);
    a[m] := a[j]; a[j] := v; return j;
}

a[m] — pivot element

Algorithm Interchange(a, i, j)
// Exchange a[i] with a[j]
{
    p := a[i];
    a[i] := a[j];
    a[j] := p;
}

10	16	8	12	15	6	3	9	5	∞
p	i							j	j

10	5	8	12	15	6	3	9	16

$T(n) = a \cdot T(n/b) + f(n)$

$T(n) = 2 \cdot T(n/2) + n$ for best case

Teacher's Signature:........

0. 25 12 13 5 70 23 10 j   j = j - 1

p i i j

25 12 13 5 10 23 70   i < j

p i j

23 12 13 5 | 10 25 70

p i j | p i j

25 12 13 5 70 23 10   j = j - 1

p i i i j j

25 12 13 5 10 23 70

i j

23 12 13 5 10 25 70

p i i j i j i

10 12 13 5 23 25 70

Teacher's Signature

(10 12 13 5) 23 25 70

pi i i++ j-- j j

(10 5 13 12) 23 25 70

ij

j j--

5 10 (13 12) 23 25 70

pi i++ ij j-- j

5 10 12 13 23 25 70

25 12 13 5 70 23 10

pi i++ i i++ ij j-- j j-- j

25 12 13 5 10 23 70

(23 12 13 5 10) 25 70

pi i++ ij j-- j

(10 12 13 5) 23 25 70

pi i++ i j j-- j

(10 5 13 12) 23 25 70

pi i++ j i j-- j

5 10 (13 12) 23 25 70

pi i++ ij j-- j

5 10 12 13 23 25 70

This is called Quicksort. (for Worst case wherein pivot starts from the first or the last)

Teacher's Signature:.........

Greedy Algorithms

// Algorithm Greedy (a, n)
// a[1:n] contains the n inputs.
{
solution := ϕ; // Initialize
for i:=1 to n do
{
x:

Knapsack problem — greedy & dynamic
Minimum cost spanning trees - Kruskal and prim's algorithm
Single source shortest path — greedy & dynamic
Job sequencing with deadlines

Knapsack problem
Given positive integers $P_1, P_2, \dots, P_n, W_1, W_2, \dots W_n$ and M (object each one has values P_n and W_n)
Find $X_1, X_2, \dots X_n$, $0 \le X_i \le 1$ such that
$\sum_{i=1}^{n} P_i X_i$ is maximized, subject to $\sum_{i=1}^{n} W_i X_i \le M$
profit — constraint
(fractional part of the object is allowed)
It should not exceed the given capacity

Q. $M = 20$, $(P_1, P_2, P_3) = (25, 24, 15)$; $(W_1, W_2, W_3) = (18, 15, 10)$

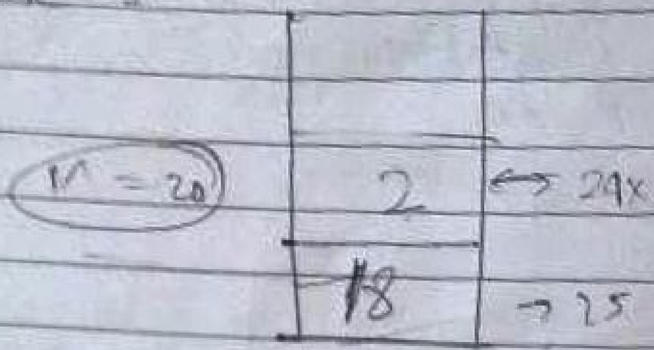
Soln → Case - I : Largest Profit

Case - II : Smallest weight

Case - III : Ratio of P_i / W_i

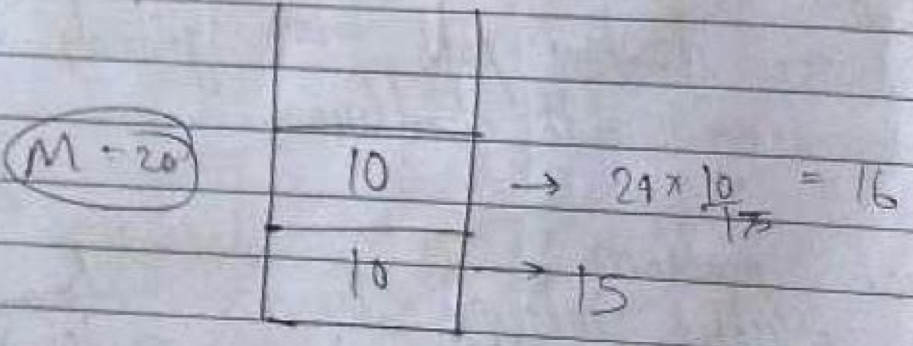
Case - I :

$\frac{P_1}{W_1} = \frac{25}{18} = 1.3$



$W = 25 + 3.2 = 28.2 \quad \left(1, \frac{2}{15}, 0\right)$

Case - II :



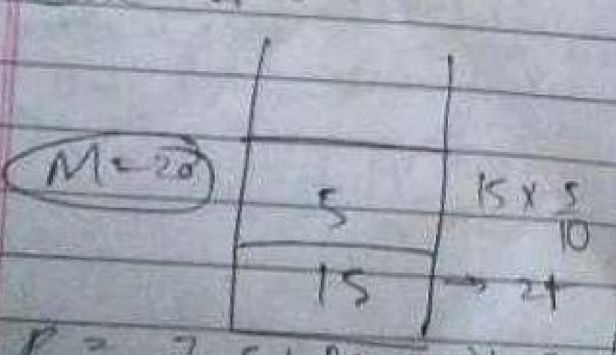
$P = 16 + 15 = 31 \quad \left(0, \frac{2}{3}, 1\right)$

Case - III :

$P_1 = 25,\ W_1 = 18,\ \frac{P_1}{W_1} = \frac{25}{18}$

$P_2 = 24,\ W_2 = 15,\ \frac{P_2}{W_2} = \frac{24}{15}$

$P_3 = 15,\ W_3 = 10,\ \frac{P_3}{W_3} = \frac{15}{10}$



$P = 7.5 + 24 = 31.5 \quad \left(0, 1, \frac{1}{2}\right)$

(*) Master Theorem is not used here as Divide and Conquer technique is not involved (for finding the complexity).

$\theta(n \log n)$

Q. Find the fractional knapsack solution for given input with capacity M = 100 and the objects are given as:

	W_i	P_i
A	90	30
B	50	50
C	20	70
D	35	20

Soln. Case - I : Largest Profit

M = 100

30	$30 \times \frac{30}{90}$
50	$\rightarrow 50$
20	$\rightarrow 70$

$P = 10 + 50 + 70$
$= 130$

$\left(\frac{1}{3}, 1, 1, 0\right)$

Case - II : Smallest Weight

M = 100

45	$\rightarrow 50 \times \frac{45}{50}$
35	$\rightarrow 20$
20	$\rightarrow 70$

$P = 45 + 20 + 70$
$= 135$

$\left(0, \frac{9}{10}, 1, 1\right)$

Case - III : Ratio of P_i/W_i

$\frac{P_1}{W_1} = \frac{30}{90} = \frac{1}{3} = 0.33$ — (1)

$\frac{P_2}{W_2} = \frac{50}{50} = 1$ — (2)

$\frac{P_3}{W_3} = \frac{70}{20} = 3.5$

$\frac{P_4}{W_4} = \frac{20}{35} = 0.57$ — (3)

M = 100

30
20
50

$\cancel{20\times}\ 20 \times \frac{30}{35} = 17.14$

$\rightarrow 70$

$\rightarrow 50$

$P = \cancel{137.14}\ 50 + 70 + 17.14$
$= 137.14$

ASTAMĪ, KRSNA, 09, PHĀLGUNA, 536

Lab

Merge Sort

0	1	2	3	4	5	6	7
9	3	7	5	6	4	8	2
lb				↑			ub

$mid = \frac{lb+ub}{2} = \frac{0+7}{2} = 3.5 = 4$

method → divide
→ merge

lb < ub

mid

Analysis - (worst case, average, best
[illegible])

It is considered to be the best algorithm [illegible] in terms of time complexity.

$$T(n) = a \cdot T(n/b) + f(n)$$

case-I $f(n) \leq n^{\log_b a}$

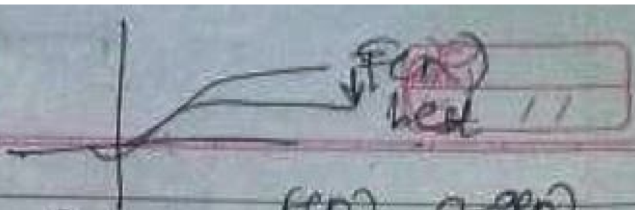
$$\boxed{T(n) = \Theta\left(n^{\log_b a}\right)}$$

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

$$\boxed{T(n) = \Theta\left(n^{\log_b a} \cdot \log n\right)}$$

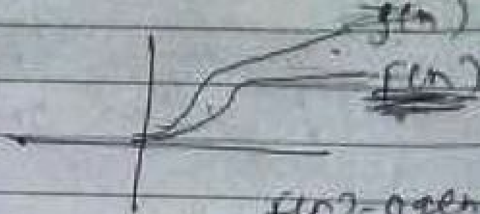
case III $f(n) > n^{\log_b a}$

$$\boxed{T(n) = \Theta(f(n))}$$



$f(n) = \Omega g(n)$

$T(n) = 4 \cdot T(n/2) + n^3$

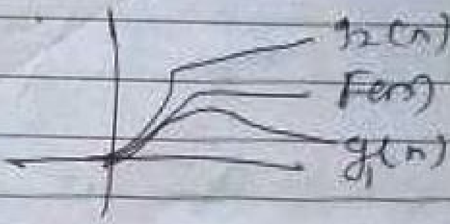


$a = 4,\ b = 2 \quad f(n) = n^3$

$f(n) = O g(n)$

$f(n) = n^3$

$\boxed{n^{\log_b a}} = n^{\log_2 4}$



$\boxed{n^2}$

$f(n) = O g(n)$

$f(n) >$

$\boxed{T(n) = \Theta(n^3)}$

f(n) = actual function return and expected time

$a = 10,\ b = 20$

g(n) = estimated time

for(i=0, i<n, i++) (n+1)
{
sum = sum + i; = n
return sum
}

Time & Space

O = Worst case

Θ = Average case

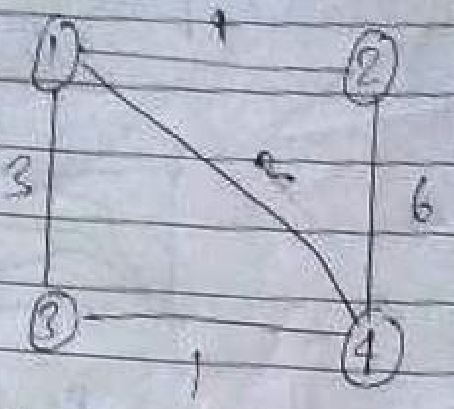
Ω = Best

a = 1 byte, b = —

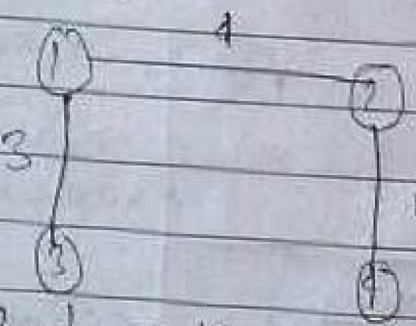
i = n byte

O(n)

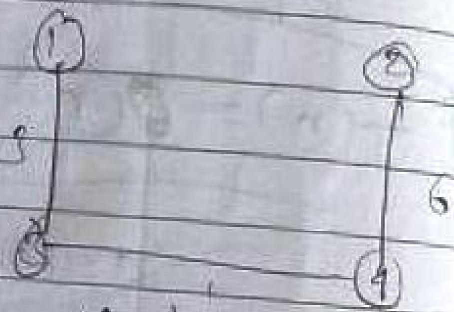
Minimum Spanning Tree (MST) Using Prim's Algorithm and Kruskal's Algorithm



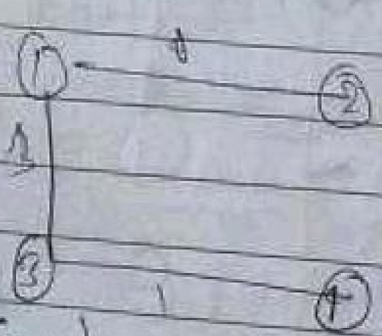
Spanning trees:



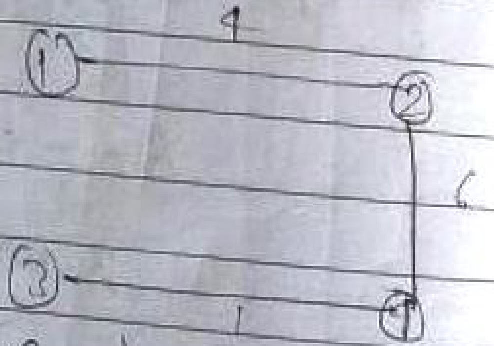
Cost = 13



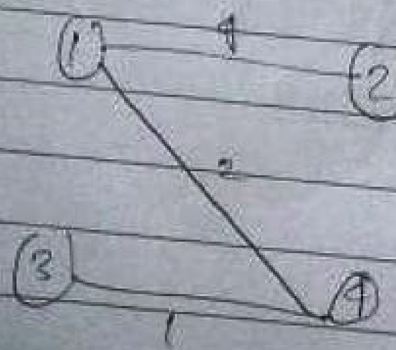
Cost = 10



Cost = 8



Cost = 11



Cost = 7 (Optimal)

Prim's Algorithm for Finding MST

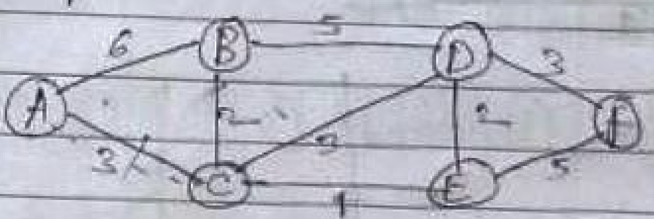
Step 1 : $x \in V$, let $A = \{x\}$, $B = V - \{x\}$ ← vertices already visited

Step 2 : Select $(u, v) \in E$, $u \in A$, $v \in B$ such that (u, v) has the smallest weight between A and B.

Step 3 : Put (u, v) in the tree. $A = A \cup \{v\}$, $B = B - \{v\}$

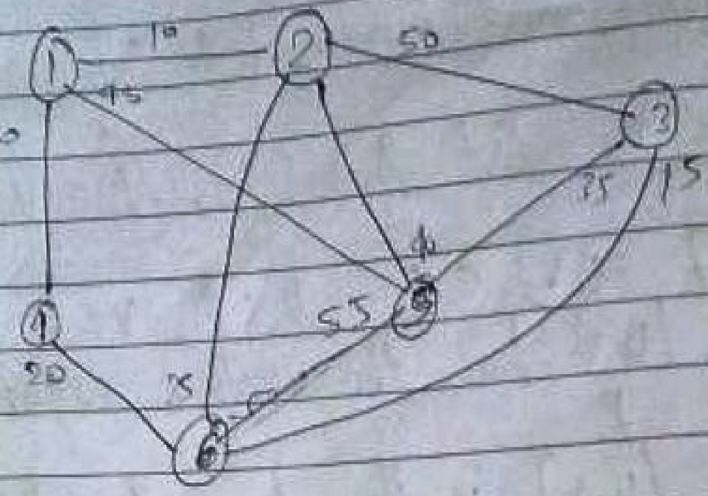
Step 4 : If $B = \phi$, stop; otherwise, go to step 2

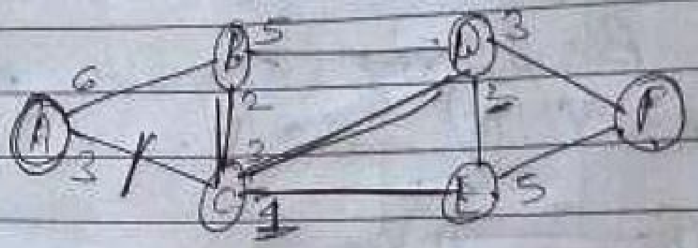
Time complexity : $O(n^2)$, $n = |V|$.

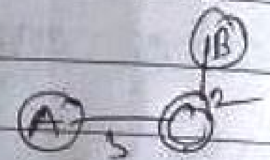
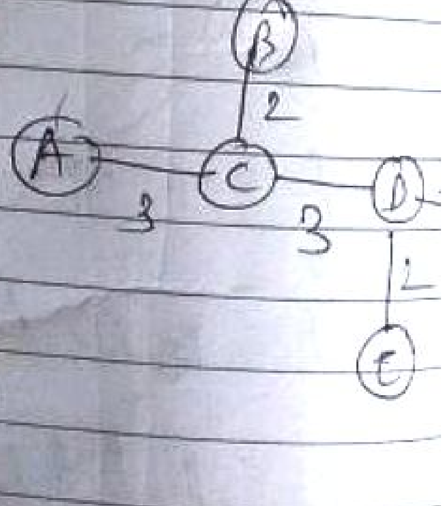


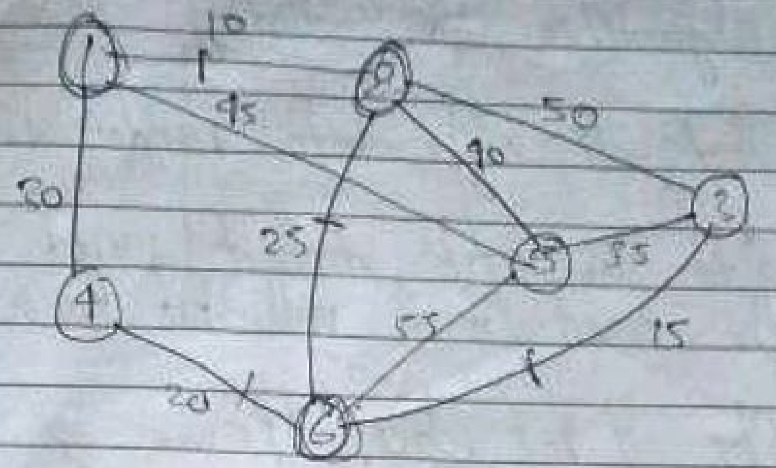
Vertices covered	Edge considered	Cost	Updated MST
(A) →	(A, B) ~~(A, C)~~	3	A —3— C
(A, C) →	(A, B), (B, D), ~~(B, C)~~	2	B —2— C, A —3— C
A, C, B →	(A, B), (B, D), (C, D), (C, E)	3	B —2— C, A —3— C —3— D
(A, C, D, B)	(A, B), (B, D), ~~(C, D)~~, (C, E)	.	
(A, C, B, D, E)	(A, B) (B, D), (C, E), (E, F)		

Order : $n(n-1)$

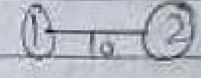
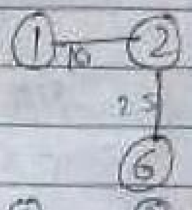
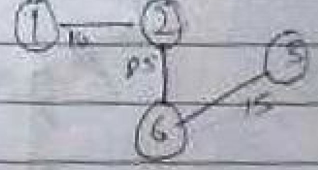
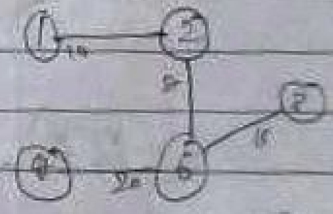




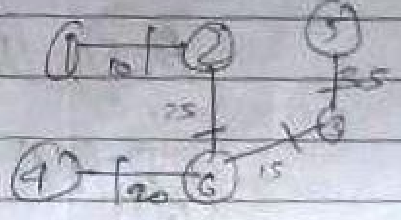
Vertices Covered	Edge Considered	Cost	Updated MST
(A) →	(A, B) ~~(A, C)~~	3	
(A, C) →	(A, B), (C, B), (C, D) (C, E)	2	
(A, C, B)	(A, B) (C, D), (C, E) (B, D)	3	
(A, C, B, D)	(D, E), (D, F), (E F)	3	
(A, C, B, D, E)		= 13	



105

Vertices	Edge Considered	Cost	Updated MST
(1) →	~~(1,2)~~, (1,4), (1,5)	10	
(1,2) →	(1,4), (1,5), (2,3), (2,5), ~~(2,6)~~	25	
(1,2,6) →	(1,4), (1,5), (2,3), (2,5), (6,5), ~~(6,3)~~, (6,4)	15	
(1,2,6,3) →	(1,4), (1,5), (2,3), (2,5), (6,5), ~~(6,4)~~, (3,5)	20	
(1,2,6,3,4) →	(1,4), (1,5), (2,3), (2,5), (6,5), (3,5)	35	

105



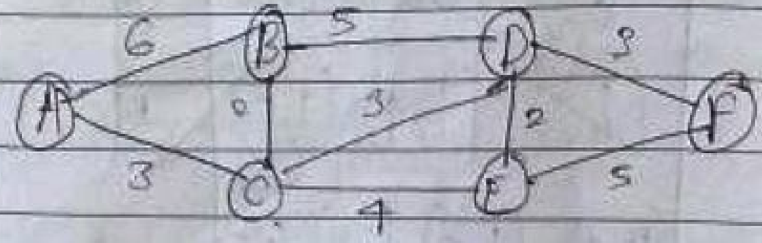
Kruskal's algorithm for finding MST

Step 1: Sort all edges into nondecreasing

Step 2: Add the next smallest weight edge to the forest if it will not cause a cy

Step 3: Stop if n-1 edges. Otherwise, go to Step

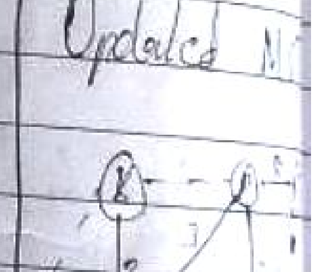
Q.



Soln

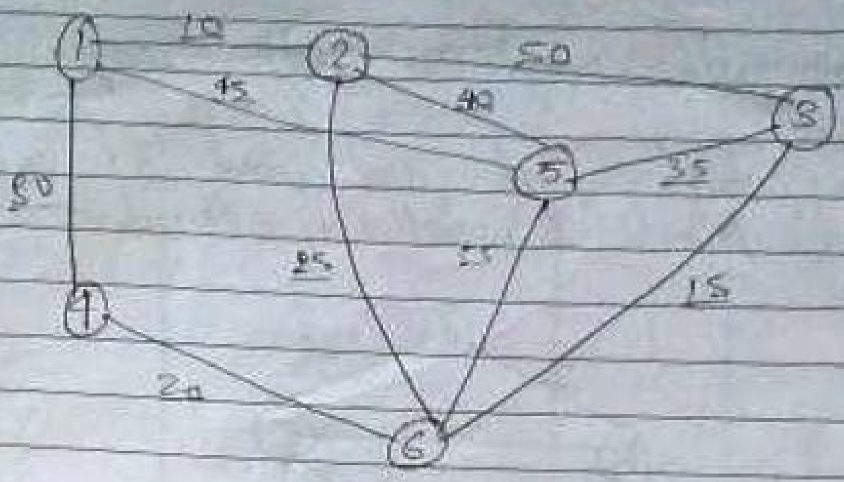
DE = 2, BC = 2, CD = 3, AC = 3, DF = 3,
CE = 4, BD = 5, EF = 5, AB = 6

Edges Covered	Edge Considered	Cost	Updated MST
(D, E)	(D, E)	2	
(D, E), (B, C)	(B, C)	2	
(D, E), (B, C), (C, D)	(C, D)	3	
(D, E), (B, C), (C, D), (A, C)	(A, C)	3	
(D, E), (B, C), (C, D), (A, C), (D, F)	(D, F)	3	



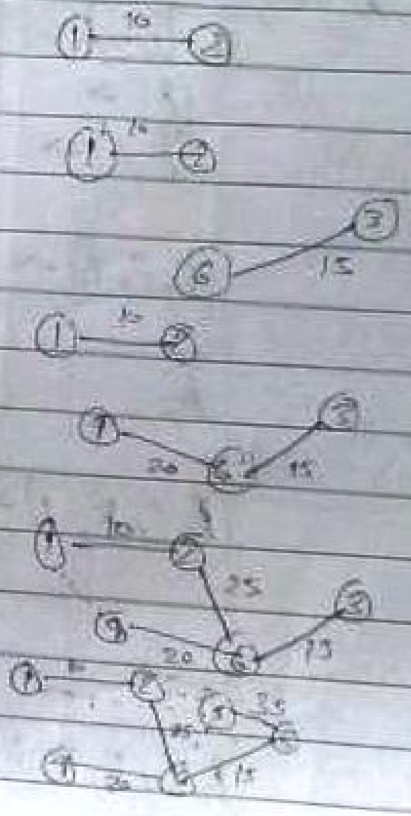
Cost = 13

Q.



Soln. (1)(2) = 10, (3)(6) = 15, (4)(6) = 20, (2)(6) = 25, (1)(4) = 30,
(3)(5) = 35, (2)(5) = 40, (1)(5) = 45, (2)(3) = 50, (5)(6) = 55

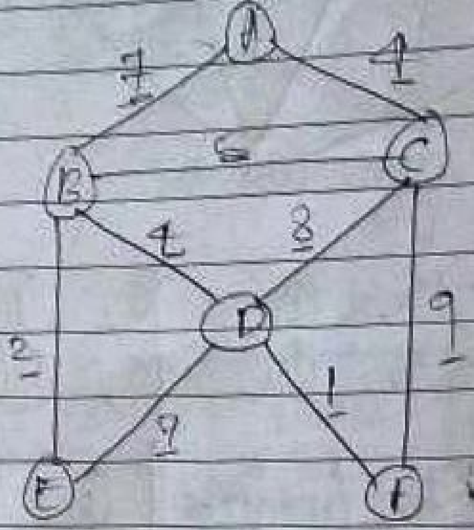
Edges Covered	Edge Considered	Cost	Updated MST
(1)(2)	(1)(2)	10	
~~(3)(6)~~ (1)(2), (3)(6)	(3)(6)	15	
(1)(2), (3)(6), (4)(6)	(4)(6)	20	
(1)(2), (3)(6), (4)(6), (2)(6)	(2)(6)	25	
(1)(2), (3)(6), (4)(6), (2)(6), (3)(5)	(3)(5)	35	



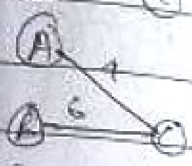
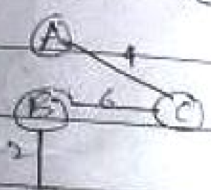
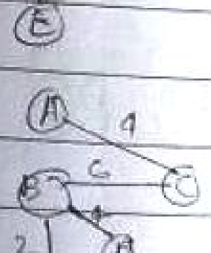
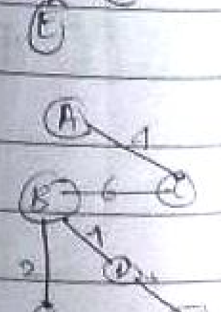
Cost = 105

Q. Explain what is MST with example

Consider foll^n graph and find out cost of MST using Prim's as well as Kruskal's algorithm

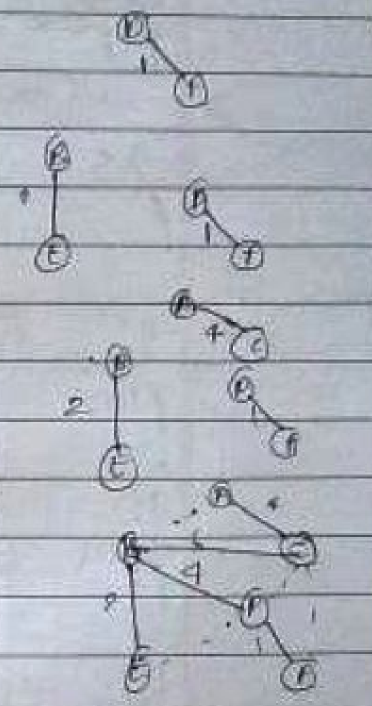


Soln.

Vertices	Edges Considered	Cost	Updated MST
(A) →	(A,B), ~~(A,C)~~	4	
(A,C) →	(A,B), ~~(C,B)~~, (C,D), (C,P)	6	
(A,C,B) →	(A,B), (C,D), (C,F), (B,D), ~~(B,E)~~	2	
A E,			
(A,C,B,E) →	(A,B), (C,D), (C,P), ~~(B,D)~~, (E,D)	4	
A C,B,E,D →	(A,B), (C,D), (C,F), (E,D), (D,F)	1	
		Cost = 17	

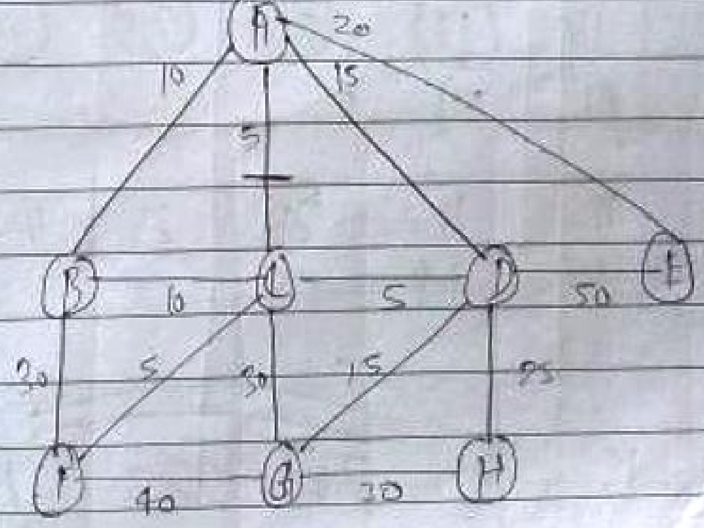
DF = 1, BE = 2, AC = 4, BD = 4, BC = 6, AB = 7,
ED = 7, CD = 8, CF = 9

Edges Covered	Edge Considered	Cost	Updated MST
(D,F)	(D,F)	1	
(D,F), (B,E)	(B,E)	2	
(D,F), (B,E), (A,C)	(A,C)	4	
(D,F), (B,E), (A,C), (B,D)	(B,D)	4	
	(B,C)	6	



Cost = 17

Q.



Ans.: 85

10 12 13 5 23 25 70

pi i i j

10 5 13 12 23 25 70

i j i i

5 10 13 12 23 25 70

pi i j

5 10 12 13 23 25 70

24 9 29 14 19 27

pi i i j

24 9 19 14 29 27

i j

j i

19 9 19 24 29 27

pi j i j

9 14 19 29 29 27

pi i j

24 9 29 14 19 27

pi j

pi j

19 9 29 14 24 27

i pj

i pj

i pj

19 9 24 14 29 27

pj i

pj

29 9 29 14 19 27

pi j

pi j

19 9 29 14 21 27

i pj

i pj

j pj

19 9 24 14 29 27

9 pi j

pi j

(19 9 14) 24 (29 27)

i pj

pj i

19 9 14

pi j

14 9 19

j pj

i pj i

(14 9) 19 24 (29 27)

19 9 | 9 14 19 24 (29 27

pi j | 29 27

9 14 | pi j 9 14 19 24

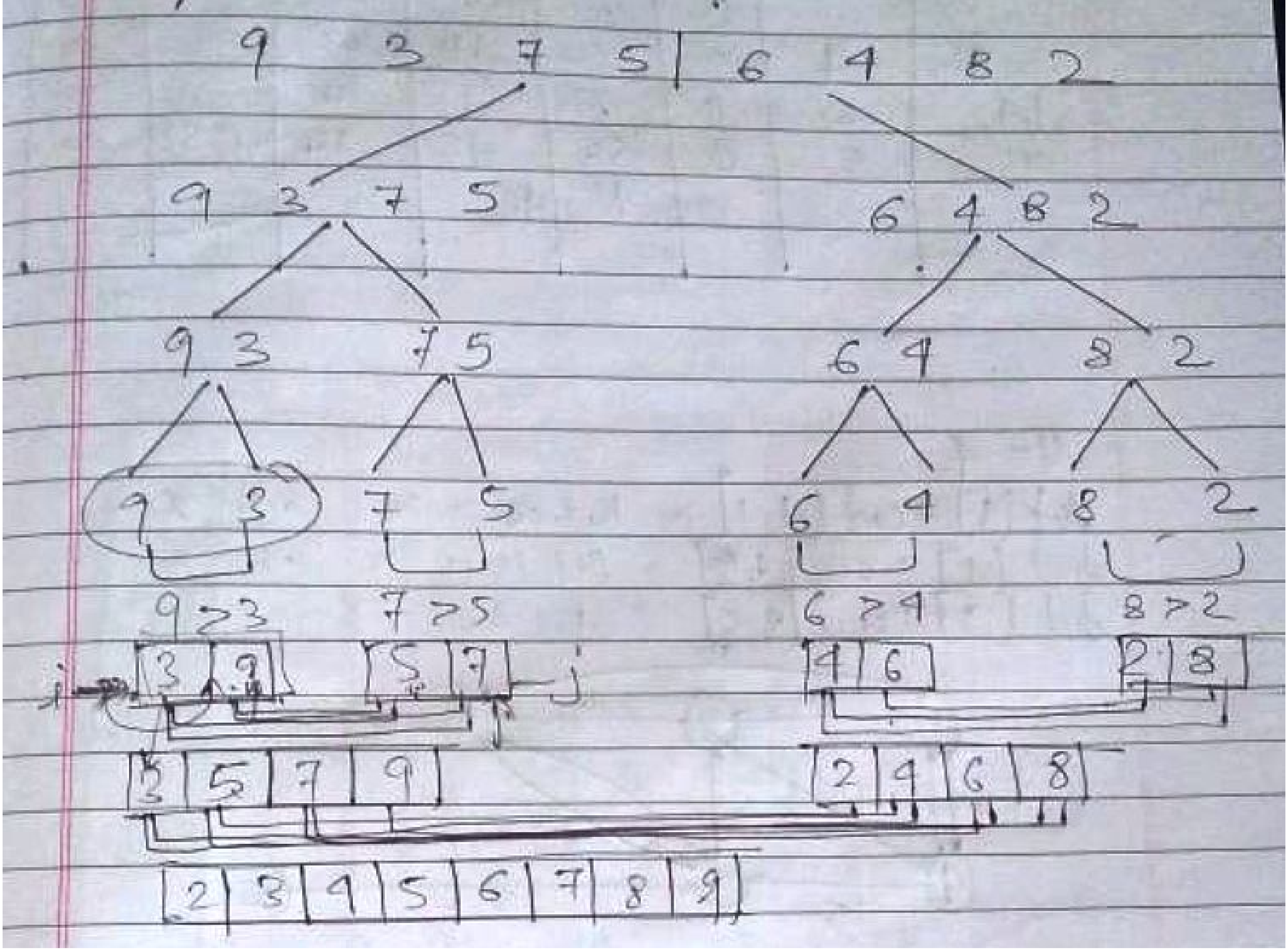
i pj | 27 29

pj i | i pj

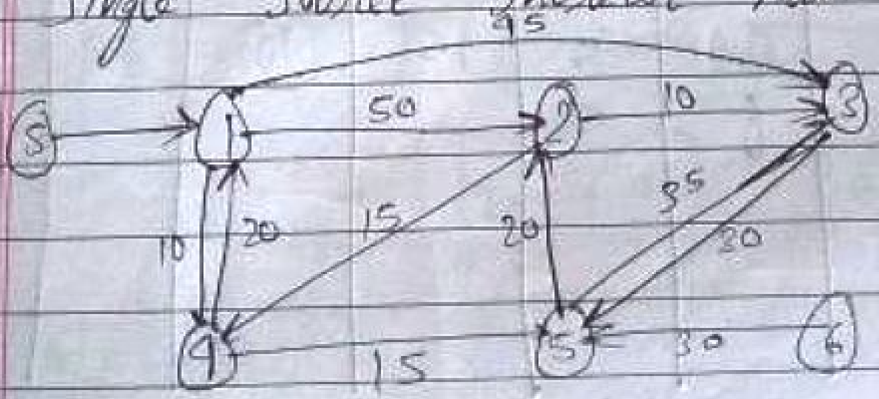
pji

Teacher's Signature:.........

Merge Sort


9 3 7 5 | 6 4 8 2
9 3 7 5
6 4 8 2
9 3
7 5
6 4
8 2
9 3
7 5
6 4
8 2
9 > 3
7 > 5
6 > 4
8 > 2
3 9
5 7
4 6
2 8
3 5 7 9
2 4 6 8
2 3 4 5 6 7 8 9

Single Source Shortest Path (SSSP)



Teacher's Signature

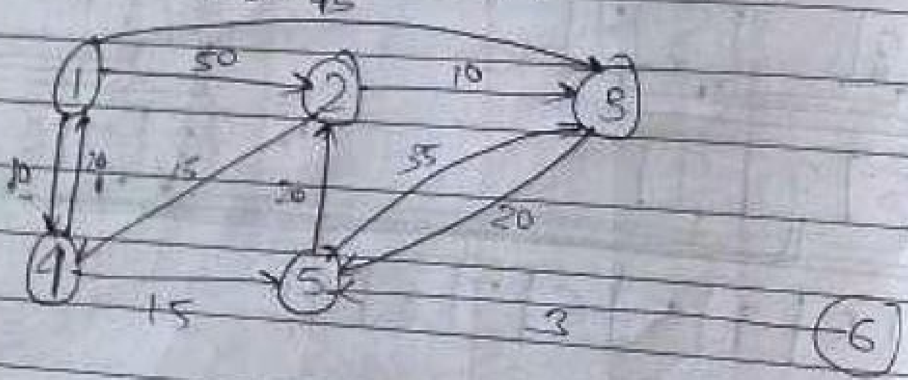
	Vertex Selected	1	2	3	4	5	6
		0	∞	∞	∞	∞	
1	4	0	50	45	10	∞	∞
1,4,	3	0	50	45	10	∞	∞
	5	0	50	45	10	25	∞
			45	45	10		

u = 4

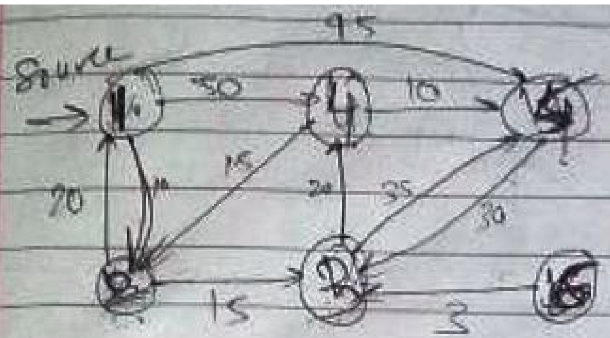
dist [4] + cost [4, 1] = 10 + 20 = 30 < 10? X

dist [4] + cost [4, 5] = 10 + 15 = 25 < 10? X

dist [3] + cost [3, 5] = 45 + 30 = 75



	Vertex Selected	1	2	3	4	5	6
		0	∞	∞	∞	∞	∞
1	4	0	50	45	10	∞	∞
1,4		0	50				



Single Source shortest path

$O(n)^2$

2

dist [w] = $\underline{\text{dist [u]}}$ + cost(u, w)

10 + cost(2, 3)

$n^2 - n$

$n*(n-1)$

S	vertex selected	[1]	[2]	[3]	[4]	[5]	[6]
	1	0	10	∞	50	45	∞
2	2	0	10	25	50	45	∞
2,3		0	10	25	45	45	∞
2,3,4		0	10	25	45	45	
2,3,4,5		0	10	25	45	45	

1-2 = 10

1-2-3 = 25

1-2-3-4 = 45

1-5 = 45

Dynamic Programming

Q. Compare Greedy, Divide and Conquer, Dynamic

Steps for Designing a Dynamic Programming Algorithm

1. Characterize optimal sub-structure
2. Recursively define the value of an optimal solution
3. Compute the value, bottom up
4. (if needed) Construct an optimal solution

Teacher's Signature:

List of algorithms

General Method
Multistage graphs
All pair shortest path
Single source shortest path
0/1 knapsack
Travelling salesman problem
Matrix chain multiplication

0/1 Knapsack

Knapsack 0-1 Example

i/W	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	3	3	3	3
2	0	0	3	4	4	7
3	0	0	3	4	5	7
4	0	0	3	4	5	7

// Initialize the base cases

for w = 0 to W

B[0, w] = 0

for i = 1 to n

B[i, 0] = 0

i \ w	0	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0
1	0	0	0	10	10	10	10	10	10
2	0	0	40	40	40	40	40	50	50
3	0	0	40	40	40	40	40	50	(change) 70
4	0	50	50	50	50	90	90	90	90

(90 → 70) ⇒ W(10 − 3 = 7), check column 7 others

(40 → 10) ⇒ W(7 − 4 = 3), check column 3, no change

Master Theorem

O = Worst Case

Θ = Average Case

Ω = Best Case

i) $f(n) < n^{\log_b a}$ then $T(n) = O(n^{\log_b a})$

ii) $f(n) = n^{\log_b a}$ then $T(n) = O(n^{\log_b a} . \log n)$

iii) $f(n) > n^{\log_b a}$, then $T(n) = O(f(n))$

e.g.

1) For Binary Search,

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

$a = 1,\ b = 2,\ f(n) = 1$

$n^{\log_b a} = n^{\log_2 1} = n^0 = 1$

$\therefore n^{\log_b a} = f(n)$ (Case 2)

$\therefore T(n) = O(n^{\log_b a} \cdot \log n) = O(\log n)$

2) For ~~MaxMin~~ a recurrence relation

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

$a = 2,\ b = 2,\ f(n) = n^2$

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

$f(n) = n^2$

$f(n) > n^{\log_b a}$ — (case 3)

$\rightarrow T(n) = O(f(n)) = O(n^2)$

Q. $T(n) = T\left(\frac{9n}{10}\right) + n$

$a = 1,\ b = \frac{10}{9},\ f(n) = n$

$n^{\log_b a} = n^{\log_{\frac{10}{9}} 1} = n^0 = 1$

$f(n) > n^{\log_b a}$ — (Case 3)

$T(n) = O(f(n)) = O(n)$

Q. $T(n) = 4 \cdot T\left(\frac{n}{2}\right) + n^3$

$a = 4, \; b = 2, \; f(n) = n^3$

$n^{\log_b a} = n^{\log_2 4} = n^2$

$f(n) > n^{\log_b a}$ (Case III)

$T(n) = O(f(n)) = O(n^3)$

Q. $T(n) = 8\,T\left(\frac{n}{4}\right) + n^2$

$a = 8, \; b = 4, \; f(n) = n^2$

$n^{\log_b a} = n^{\log_4 8} = n^{1.5}$

$f(n) > n^{\log_b a}$ (Case III)

$T(n) = O(f(n)) = O(n^2)$

Q. $T(n) = 4\,T\left(\frac{n}{2}\right) + n$

$a = 4, \; b = 2, \; f(n) = n$

$n^{\log_b a} = n^{\log_2 4} = n^2$

$f(n) < n^{\log_b a}$ (Case I)

$T(n) = O\left(n^{\log_b a}\right)$

$= O(n^2)$

Q. $T(n) = 8T\left(\frac{n}{2}\right) + 1000n^2$

$a = 8, \quad b = 2, \quad f(n) = 1000n^2$

$n^{\log_b a} = n^{\log_2 8} = n^3$

$f(n) < n^{\log_b a}$ (Case 1)

$T(n) = O\left(n^{\log_b a}\right)$

$= O(n^3)$

Q. $T(n) = 4T\left(\frac{n}{2}\right) + n^3$

$a = 4, \quad b = 2, \quad f(n) = n^3$

$n^{\log_b a} = n^{\log_2 4} = n^2$

$f(n) > n^{\log_b a}$ (Case 3)

$T(n) = O(f(n)) = O(n^3)$

Q. $T(n) = 3T\left(\frac{3n}{6}\right) + n$

$a = 3, \quad b = 3, \quad f(n) = n$

$n^{\log_b a} = n^{\log_3 3} = n$

$f(n) = n^{\log_b a}$ – (Case 2)

$T(n) = O\left(n^{\log_b a} \cdot \log n\right)$

$= O(n \log n)$

Q. $T(n) = 16\,T\left(\frac{n}{8}\right) + n^2$

$a = 16,\ b = 8,\ f(n) = n^2$

$n^{\log_b a} = n^{\log_8 16}$

$= n^{1.33}$

$f(n) = n^{\log_b a}$ (Case 3)

$T(n) = O(f(n))$

$= O(n^2)$

Substitution:

Identify a, b and f(n)

Guess the solution

Verify guess using induction

Q. $T(n) = \begin{cases} 1 & T(1) \\ T(n-1) + n & \text{if } n > 1 \end{cases}$

Let $T(n)$ be $O(n^2)$ (to prove)

$T(1) = 1$

$T(2) = T(1) + 2 = 1 + 2$

$T(3) = T(2) + 3 = 1 + 2 + 3$

$\vdots$

$T(n-1) = \Sigma(1 + 2 + \dots + n-1) + n$

$= \frac{n(n-1)}{2} + n$

$= O(n^2)$

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

($n\log n$ by Master Theorem)

Let $T(n)$ be $n\log n$

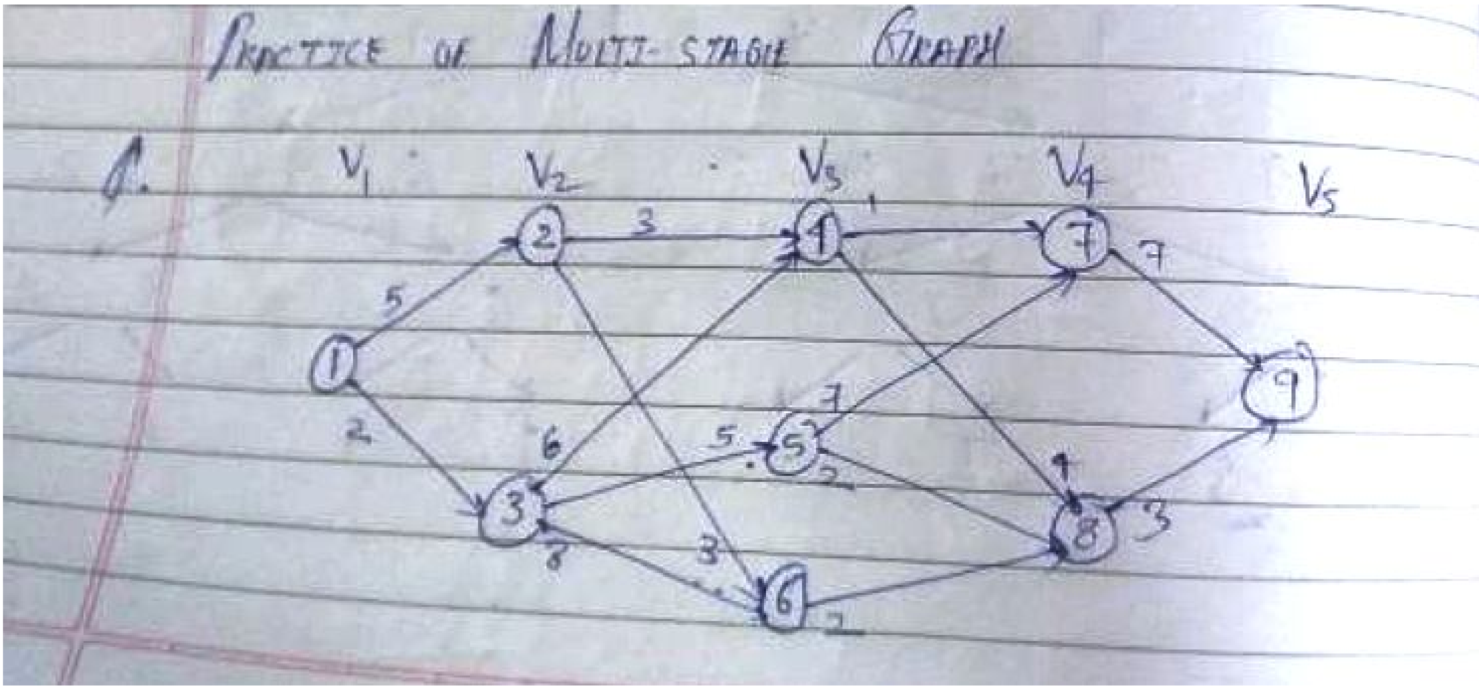
$$= 2T\left(\frac{n}{2}\right) + n$$

$$= 2 \cdot \frac{n}{2} \log\left(\frac{n}{2}\right) + n$$

$$= n[\log n - 1] + n$$

$$= \cancel{n[\log}\ n[\log n]$$

$$= O(n\log n)$$


PRACTICE OF MULTI-STAGE GRAPH
1.
V1
V2
V3
V4
V5

Forward Approach

Stage 5: $cost(5,9) = 0$

Stage 4: $cost(4,7) = 7$

$cost(4,8) = 3$

Stage 3: $cost(3,4) = \min\begin{cases} c(4,7) + cost(4,7) \\ c(4,8) + cost(4,8) \end{cases}$

$= \min\begin{pmatrix} 1+7 \\ 4+3 \end{pmatrix}$

$= 7$

$cost(3,5) = \min\begin{cases} c(5,7) + cost(4,7) \\ c(5,8) + cost(4,8) \end{cases}$

$= \min\begin{pmatrix} 7+7 \\ 2+3 \end{pmatrix}$

$= 5$

$cost(3,6) = \min\{ c(6,8) + cost(4,8)$

$= 2+3$

$= 5$

Stage 2: $cost(2,2) = \min\begin{cases} c(2,4) + cost(3,4) \\ c(2,6) + cost(3,6) \end{cases}$

$= \min\begin{pmatrix} 3+7 \\ 3+5 \end{pmatrix}$

$= 8$

$cost(2,3) = \min\begin{cases} c(3,4) + cost(3,4) \\ c(3,5) + cost(3,5) \\ c(3,6) + cost(3,6) \end{cases}$

$= \min\begin{pmatrix} 6+7 = 10 \\ 5+5 \\ 8+5 \end{pmatrix}$

Teacher's Signature:

Stage 1 : $cost(1,1) = \min \begin{cases} c(1,2) + cost(2,2) \\ c(1,3) + cost(2,3) \end{cases}$

$= \min \begin{cases} 5 + 8 \\ 2 + 10 \end{cases}$

$= 12$

$d(1,1) = 3$
$d(2,2) = 6$
$d(2,3) = 5$
$d(3,4) = 8$
$d(3,5) = 8$
$d(3,6) = 8$
$d(4,7) = 9$
$d(4,8) = 9$

$V_1 \quad V_2 \quad V_3 \quad V_4 \quad V_5$

$1 — 3 — 5 — 8 — 9$

$V_2 = d(1,1) = 3$
$V_3 = d(2, d(1,1)) = d(2,3) = 5$
$V_4 = d(3, d(2,3)) = d(3,5) = 8$
$V_5 = d(4, d(3,5)) = d(4,8) = 9$

Backward Approach

Stage 1 : $bcost(1,1) = 0$

Stage 2 : $bcost(2,2) = 5$
$bcost(2,3) = 2$

Stage 3 : $bcost(3,4) = \min \begin{cases} bcost(2,2) + cost(2,4) \\ bcost(2,3) + cost(3,4) \end{cases}$

$= \min \begin{cases} 5 + 3 \\ 2 + 6 \end{cases}$

$= 8$

$bcost(3,5) = \min \{ bcost(2,3) + cost(3,5)$

$= \min (2 + 5$

$= 7$

$bcost(3,6) = \min \begin{cases} bcost(2,2) + cost(2,6) \\ bcost(2,3) + cost(3,6) \end{cases}$

$= \min \begin{cases} 5 + 3 \\ 2 + 8 \end{cases}$

$= 8$

$bcost(4,7) = \min \begin{cases} bcost(3,4) + cost(4,7) \\ bcost(3,5) + cost(5,7) \end{cases}$

$= \min \begin{cases} 8 + 1 \\ 7 + 7 \end{cases}$

$= 9$

$bcost(4,8) = \min \begin{cases} bcost(3,4) + cost(4,8) \\ bcost(3,5) + cost(5,8) \\ bcost(3,6) + cost(6,8) \end{cases}$

$= \min \begin{cases} 8 + 4 \\ 7 + 2 \\ 8 + 2 \end{cases}$

$= 9$

$bcost(5,9) = \min \begin{cases} bcost(4,7) + cost(7,9) \\ bcost(4,8) + cost(8,9) \end{cases}$

$= \min \begin{cases} 9 + 7 \\ 9 + 3 \end{cases} = 12$

~~$d(5,9) = 8$~~
~~$d(4,8) = 5$~~
~~$d(4,7) = 4$~~
~~$d(3,6) =$~~

$d(5,9) = 8$
$d(4,8) = 5$
$d(4,7) = 4$
$d(3,6) = 2$
$d(3,5) = 3$
$d(3,4) = 3$
$d(2,3) = 1$
$d(2,2) = 1$

$$\begin{array}{ccccccccc} V_5 & & V_4 & & V_3 & & V_2 & & V_1 \\ 9 & — & 8 & — & 5 & — & 3 & — & 1 \end{array}$$

$V_4 = d(5,9) = 8$
$V_3 = d(4, d(5,9)) = d(4,8) = 5$
$V_2 = d(3, d(4,8)) = d(3,5) = 3$
$V_1 = d(2, d(3,5)) = d(2,3) = 1$

(23-03-23, TH)

(V. IMP.; REFER FOR EXAM)

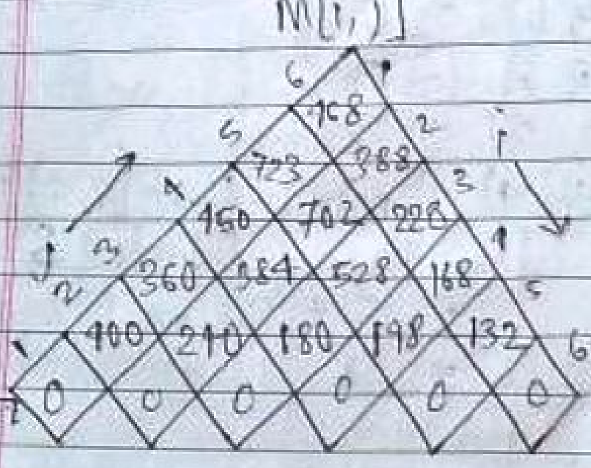
MATRIX CHAIN MULTIPLICATION

< 5, 8, 10, 3, 6, 11, 2 >

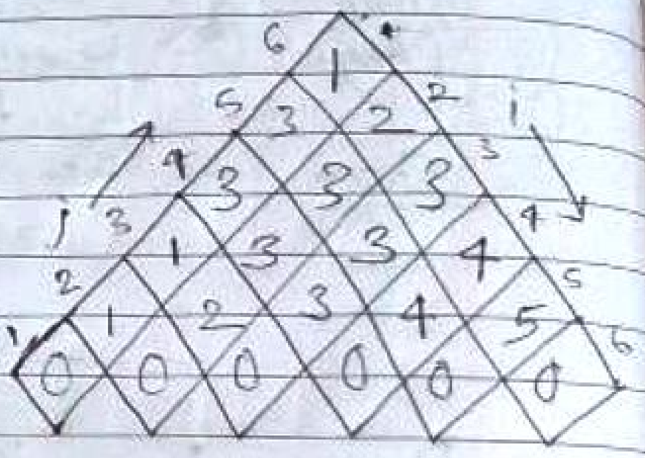
$A_1 = 5 \times 8$	$P_0 = 5$
$A_2 = 8 \times 10$	$P_1 = 8$
$A_3 = 10 \times 3$	$P_2 = 10$
$A_4 = 3 \times 6$	$P_3 = 3$
$A_5 = 6 \times 11$	$P_4 = 6$
$A_6 = 11 \times 2$	$P_5 = 11$
	$P_6 = 2$

Formula

$$m[i,j] = \begin{cases} 0 & i = j \\ \min\limits_{i \le k < j} \; m[i,k] + m[k+1,j] + P_{i-1} \cdot P_k \cdot P_j \end{cases}$$



(number of computations)



(value of k)

$i = 1$, $j = 2$

$$m[1,2] = \min_{i \le k < j} \{ m[1,1] + m[2,2] + P_0 \cdot P_1 \cdot P_2 \}$$

$(k = 1)$

$$= [0 + 0 + 5 \cdot 8 \cdot 10]$$

$$= 400$$

$$m[2,3] = \min_{i \le k < j} \{ m[2,2] + m[3,3] + P_1 \cdot P_2 \cdot P_3 \}$$

$(k = 2)$

$$= [0 + 0 + 8 \cdot 10 \cdot 3]$$

$$= 240$$

$$m[3,4] = \min_{i \le k < j} \left\{ m[3,3] + m[4,4] + P_2 \cdot P_3 \cdot P_4 \right\}$$

$(k=3)$

$= [0 + 0 + 10 \times 3 \times 6]$

$= 180$

$$m[4,5] = \min_{i \le k < j} \left\{ m[4,4] + m[5,5] + P_3 \cdot P_4 \cdot P_5 \right\}$$

$(k=4)$

$= [0 + 0 + 3 \times 6 \times 11]$

$= 198$

$$m[5,6] = \min_{i \le k < j} \left\{ m[5,5] + m[6,6] + P_4 \cdot P_5 \cdot P_6 \right\}$$

$(k=5)$

$= [0 + 0 + 6 \times 11 \times 2]$

$= 132$

$$m[1,3] = \min_{i \le k < j} \begin{cases} m[1,1] + m[2,3] + P_0 \cdot P_1 \cdot P_3 & (k=1) \\ m[1,2] + m[3,3] + P_0 \cdot P_2 \cdot P_3 & (k=2) \end{cases}$$

$= \min ([0 + 240 + 5 \times 8 \times 3], [400 + 0 + 5 \times 10 \times 3])$

$= \min ([240 + 120], [400 + 150])$

$= 360$

$$m[2,4] = \min_{i \le k < j} \begin{cases} m[2,2] + m[3,4] + P_1 \cdot P_2 \cdot P_4 & (k=2) \\ m[2,3] + m[4,4] + P_1 \cdot P_3 \cdot P_4 & (k=3) \end{cases}$$

Teacher's Signature.............

$= \min([0 + 180 + 8 \times 10 \times 6], [240 + 0 + 8 \times 3 \times ...$
$= \min([180 + 480], [240 + 144])$
$= 384$

$$m[3,5] = \min_{i \le k < j} \begin{cases} m[3,3] + m[4,5] + P_2 . P_3 . P_5 & (k = 3) \\ m[3,4] + m[5,5] + P_2 . P_4 . P_5 & (k = 4) \end{cases}$$

$= \min([0 + 198 + 10 \times 3 \times 11], [180 + 0 + 10 \times 6 \times 11])$
$= \min([198 + 330], [180 + 660])$
$= 528$

$$m[4,6] = \min_{i \le k < j} \begin{cases} m[4,4] + m[5,6] + P_3 . P_4 . P_6 & (k = 4) \\ m[4,5] + m[6,6] + P_3 . P_5 . P_6 & (k = 5) \end{cases}$$

$= \cancel{\min}\ \min([0 + 132 + 3 \times 6 \times 2], [198 + 0 + 3 \times 11 \times 2])$
$= \min([132 + 36], [198 + 66])$
$= 168$

$$m[1,4] = \min_{i \le k < j} \begin{cases} m[1,1] + m[2,4] + P_0 . P_1 . P_4 & (k = 1) \\ m[1,2] + m[3,4] + P_0 . P_2 . P_4 & (k = 2) \\ m[1,3] + m[4,4] + P_0 . P_3 . P_4 & (k = 3) \end{cases}$$

$= \min([0 + 384 + 5 \times 8 \times 6], [400 + 180 + 5 \times 10 \times 6], [360 + 0 + 5 \times 3 \times 6])$
$= \min([384 + 240], [580 + 300], [360 + 90]) = 450$

Teacher's Signature:.........

$$m[2,5] = \min_{1 \le j < k} \begin{cases} (k=2) & \{m[2,2] + m[3,5] + P_1 . P_2 . P_5\} \\ (k=3) & \{m[2,3] + m[4,5] + P_1 . P_2 . P_5\} \\ (k=4) & \{m[2,4] + m[5,5] + P_1 . P_4 . P_5\} \end{cases}$$

$= \min([0 + 528 + 8 \times 10 \times 11], [240 + 198 + 8 \times 3 \times 11], [384 + 0 + 8 \times 6 \times 11])$

$= \min([528 + 880], [438 + 264], [384 + 528])$

$= 702$

438
+264
702

88
×6
528

88
×3
264

$$m[3,6] = \min_{1 \le j < k} \begin{cases} (k=3) & \{m[3,3] + m[4,6] + P_2 . P_3 . P_6\} \\ (k=4) & \{m[3,4] + m[5,6] + P_2 . P_4 . P_6\} \\ (k=5) & \{m[3,5] + m[6,6] + P_2 . P_5 . P_6\} \end{cases}$$

$= \min([0 + 168 + 10 \times 3 \times 2], [180 + 132 + 10 \times 6 \times 2], [528 + 0 + 10 \times 11 \times 2])$

$= \min([168 + 60], [180 + 132 + 120], [528 + 220])$

$= 228$

$$m[1,5] = \min_{1 \le j < k} \begin{cases} (k=1) & \{m[1,1] + m[2,5] + P_0 . P_1 . P_5 \\ (k=2) & \{m[1,2] + m[3,5] + P_0 . P_2 . P_5 \\ (k=3) & \{m[1,3] + m[4,5] + P_0 . P_3 . P_5 \\ (k=4) & \{m[1,4] + m[5,5] + P_0 . P_4 . P_5 \end{cases}$$

Teacher's Signature: ....

$$= \min\left([0 + 702 + 5\times8\times11], [900 + 528 + 5\times10\times11], [960 + 198 + 5\times3\times11], [950 + 0 + 5\times6\times11] \right)$$

$$= \min\left([702 + 440], [928 + 550], \underset{(558)}{[960 + 198 + 165]}, [950 \ldots \right)$$

$$= 723$$

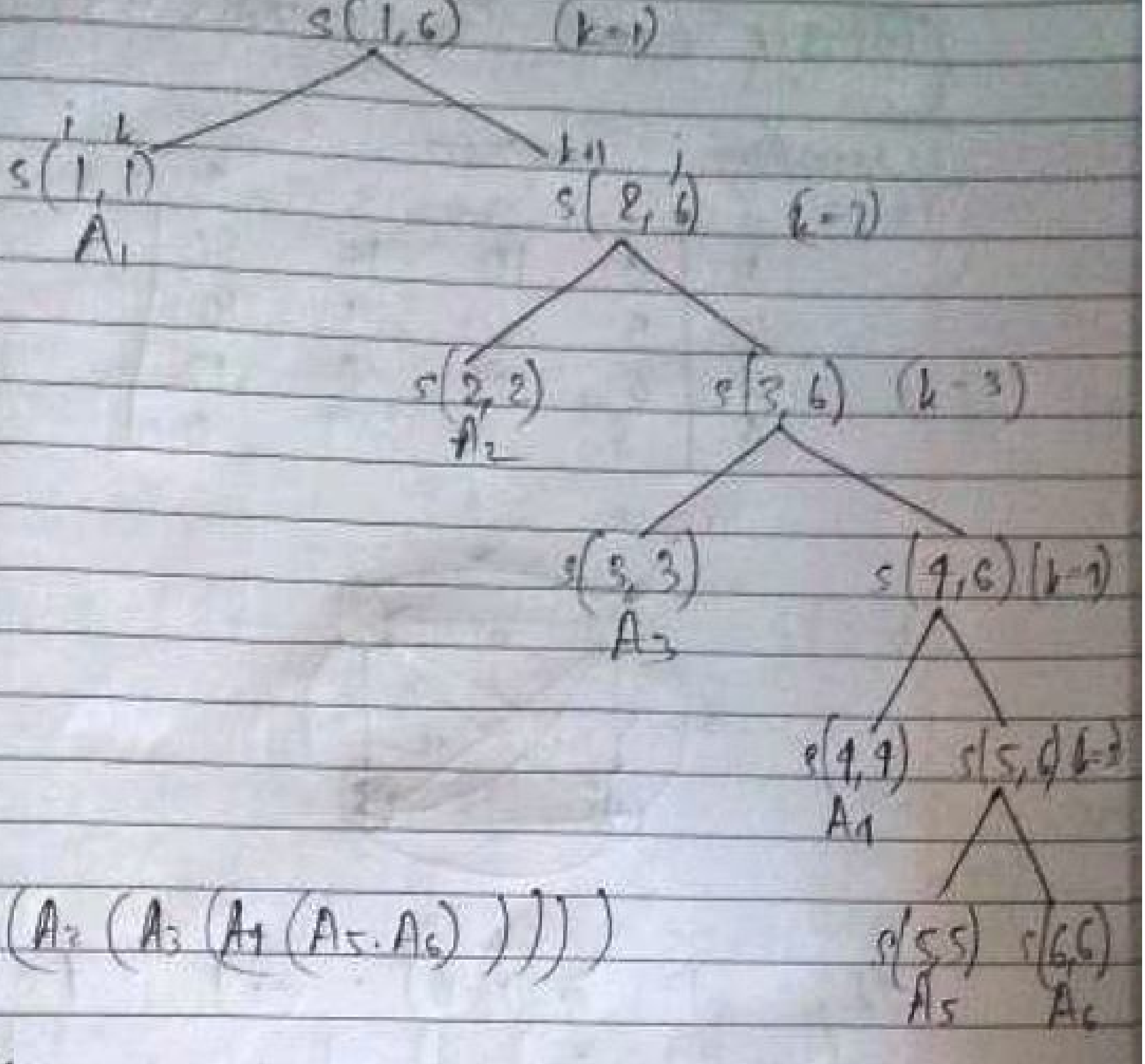
$$m[2,6] = \min_{i \le k < j} \begin{cases} m[2,2] + m[3,6] + P_1 . P_2 . P_6 & (k=2) \\ m[2,3] + m[4,6] + P_1 . P_3 . P_6 & (k=3) \\ m[2,4] + m[5,6] + P_1 . P_4 . P_6 & (k=4) \\ m[2,5] + m[6,6] + P_1 . P_5 . P_6 & (k=5) \end{cases}$$

$$= \min\left([0 + 228 + 8\times10\times2], [240 + 168 + 8\times3\times2], [384 + 132 + 8\times6\times2], [702 + 0 + 8\times11\times2] \right)$$

$$= \min\left([228 + 160], [408 + 48], [516 + 96], [702 + 176] \right)$$

$$= 388$$

$$m[1,6] = \min_{i \le k < j} \begin{cases} m[1,1] + m[2,6] + P_0 . P_1 . P_6 & (k=1) \\ m[1,2] + m[3,6] + P_0 . P_2 . P_6 & (k=2) \\ m[1,3] + m[4,6] + P_0 . P_3 . P_6 & (k=3) \\ m[1,4] + m[5,6] + P_0 . P_4 . P_6 & (k=4) \\ m[1,5] + m[6,6] + P_0 . P_5 . P_6 & (k=5) \end{cases}$$



$(A_1 (A_2 (A_3 (A_4 (A_5 . A_6)))))$

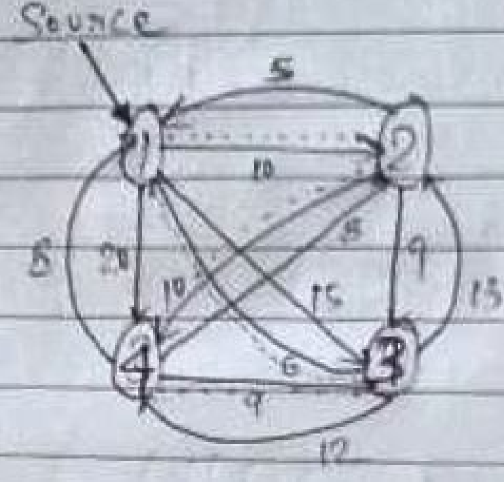
$= \min([0 + 288 + 5 \times 8 \times 2], [400 + 228 + 5 \times 10 \times 2],$
$[360 + 168 + 5 \times 3 \times 2], [450 + 132 + 5 \times 6 \times 2],$
$[723 + 0 + 5 \times 11 \times 2])$

$= \min([328 + 80], [628 + 100], [528 + 30],$
$[582 + 60], [723 + 110])$

$= 468$

Travelling Salesman

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 10 & 15 & 20 \\ 2 & 5 & 0 & 9 & 10 \\ 3 & 6 & 13 & 0 & 12 \\ 4 & 8 & 8 & 9 & 0 \end{array}$$



Step 1: $g(i, \phi) = c_{i1} \quad 1 \le i < n$

Step 2: $g(i, s) = \min\limits_{j \in s} \{ c_{ij} + g(j, s - \{j\}) \}$

Step 1: $g(2, \phi) = c_{21} = 5$

$g(3, \phi) = c_{31} = 6$

$g(4, \phi) = c_{41} = 8$

Step 2 $|s| = 1$

$g(2, \{3\}) = \min\limits_{j \in s} \{ c_{23} + g(3, \phi) \}$

$= 9 + 6 = \boxed{15}$

$g(2, \{4\}) = \min\limits_{j \in s} \{ c_{24} + g(4, \phi) \}$

$= 10 + 8 = 18$

$$g(3, \{2\}) = \min_{j \in S} \{ C_{32} + g(2, \phi) \}$$

$$= 13 + 5 = \boxed{18}$$

$$g(3, \{4\}) = \min_{j \in S} \{ C_{34} + g(4, \phi) \}$$

$$= 12 + 8 = 20$$

$$g(4, \{3\}) = \min_{j \in S} \{ C_{43} + g(3, \phi) \}$$

$$= 9 + 6$$

$$= 15$$

$$g(4, \{2\}) = \min_{j \in S} \{ C_{42} + g(2, \phi) \}$$

$$= 8 + 5 = 13$$

$|S| = 2$

$S - \{j\} \Rightarrow \{3,4\} - \{3\} - \{4\}$

$$g(2, \{3, 4\}) = \min_{j \in S} \begin{cases} C_{23} + g(3, \{4\}) \\ C_{24} + g(4, \{3\}) \end{cases}$$

$$= \min \begin{cases} 9 + 20 \\ 10 + 15 \end{cases}$$

$$= 25$$

$$g(3, \{2, 4\}) = \min_{j \in S} \begin{cases} C_{32} + g(2, \{4\}) \\ C_{34} + g(4, \{2\}) \end{cases}$$

$$= \min \begin{cases} 13 + 18 \\ 12 + 13 \end{cases}$$

$$= 25$$

$$g(4, \{2, 3\}) = \min_{j \in S} \begin{cases} C_{42} + g(2, \{3\}) \\ C_{43} + g(3, \{2\}) \end{cases}$$

$$= \min \begin{cases} 8 + 15 \\ 9 + 18 \end{cases}$$

$$= 23$$

Step III

$$g(1, \{2,3,4\}) = \min \begin{cases} C_{12} + g(2, \{3,4\}) \\ C_{13} + g(3, \{2,4\}) \\ C_{14} + g(4, \{2,3\}) \end{cases}$$

$$= \min \begin{cases} 10 + 25 \\ 15 + 25 \\ 20 + 23 \end{cases}$$

$$= 35$$

$$\boxed{\begin{aligned} g(1, \{2,3,4\}) &= 2 \\ g(2, \{3,4\}) &= 4 \\ g(4, \{3\}) &= 3 \end{aligned}}$$

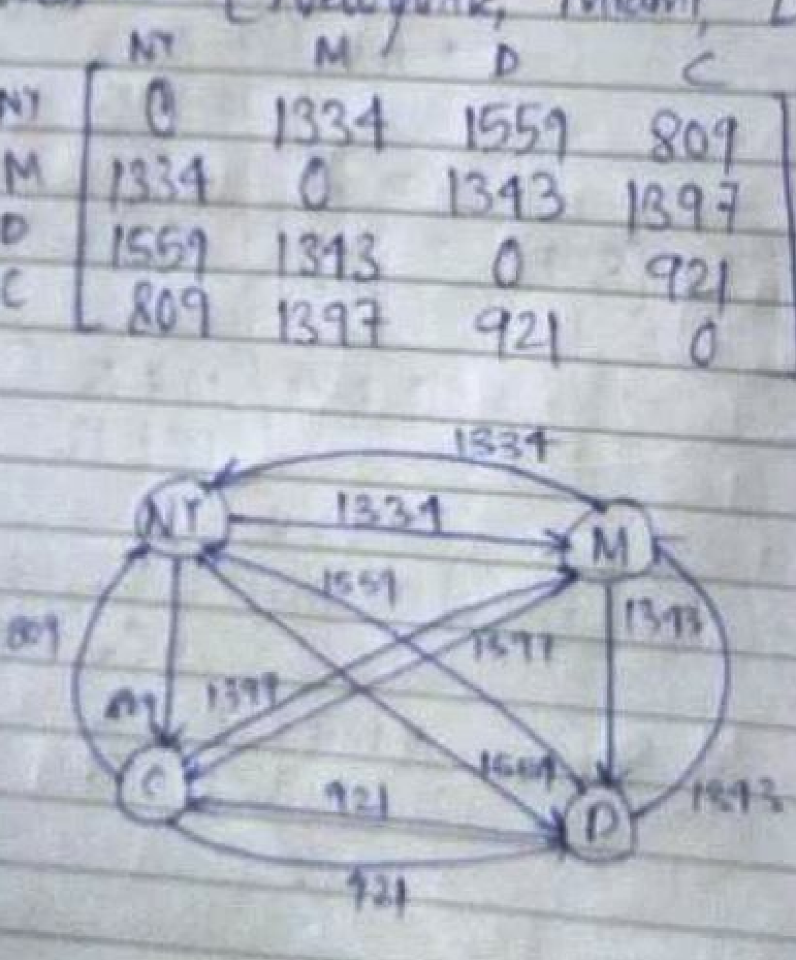
$1 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 1 = 35$

$(10 + 10 + 9 + 6)$

4. cities = {Newyork, Miami, Dallas, Chicago}

$$\begin{array}{c} \\ NY \\ M \\ D \\ C \end{array} \begin{array}{c} \begin{array}{cccc} NY & M & D & C \end{array} \\ \begin{bmatrix} 0 & 1334 & 1559 & 809 \\ 1334 & 0 & 1343 & 1397 \\ 1559 & 1343 & 0 & 921 \\ 809 & 1397 & 921 & 0 \end{bmatrix} \end{array}$$

Soln.



Step 1 : $g(M, \phi) = C_{MNY} = 1{,}334$
$g(D, \phi) = C_{DNY} = 1{,}559$
$g(C, \phi) = C_{CNY} = 809$

Step 2 : $|S| = 1$

$g(M, \{D\}) = \min_{j \in S} \{ C_{MD} + g(D, \phi)$
$= 1{,}343 + 1{,}559$
$= 2{,}902$

$g(M, \{C\}) = \min_{j \in S} \{ C_{MC} + g(C, \phi)$
$= 1{,}397 + 809$
$= 2{,}206$

$g(D, \{M\}) = \min_{j \in S} \{ C_{DM} + g(M, \phi)$
$= 1{,}343 + 1{,}334$
$= 2{,}677$

$g(D, \{C\}) = \min_{j \in S} \{ C_{DC} + g(C, \phi)$
$= 921 + 809$
$= 1{,}730$

$g(C, \{M\}) = \min_{j \in S} \{ C_{CM} + g(M, \phi)$
$= 1{,}397 + 1{,}334$
$= 2{,}731$

$g(C, \{D\}) = \min_{j \in S} \{ C_{CD} + g(D, \phi)$
$= 921 + 1{,}559$
$= 2{,}480$

$|S| = 2$

$$g(M, \{D, C\}) = \min_{j \in S} \begin{cases} C_{MD} + g(D, \{C\}) \\ C_{MC} + g(C, \{D\}) \end{cases}$$

$$= \min \begin{cases} 1,343 + 1,730 \\ 1,397 + 2,480 \end{cases}$$

$$= 3,073$$

$$g(D, \{M, C\}) = \min_{j \in S} \begin{cases} C_{DM} + g(M, \{C\}) \\ C_{DC} + g(C, \{M\}) \end{cases}$$

$$= \min \begin{cases} 1,343 + 2,206 \\ 921 + 2,731 \end{cases}$$

$$= 3,549$$

$$g(C, \{M, D\}) = \min_{j \in S} \begin{cases} C_{CM} + g(M, \{D\}) \\ C_{CD} + g(D, \{M\}) \end{cases}$$

$$= \min \begin{cases} 1,397 + 2,902 \\ 921 + 2,677 \end{cases}$$

$$= 3,598$$

Step III

$$g(NY, \{M, D, C\}) = \min_{j \in S} \begin{cases} C_{NYM} + g(M, \{D, C\}) \\ C_{NYD} + g(D, \{M, C\}) \\ C_{NYC} + g(C, \{M, D\}) \end{cases}$$

$$= \min \begin{cases} 1,334 + 3,073 \\ 1,559 + 3,549 \\ 809 + 3,598 \end{cases}$$

$$= 4,407$$

$g(NY, \{M, D, C\}) = \cancel{NY}\ M$

$g(M, \{D, C\}) = D$

$g(D, \{C\}) = C$

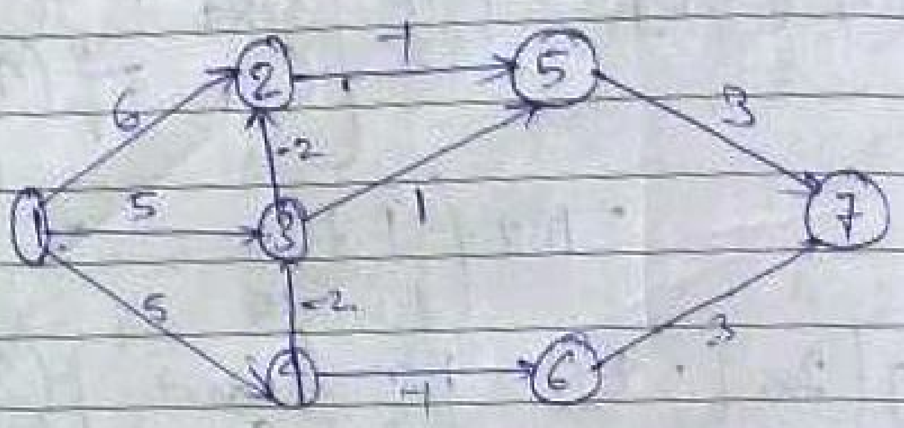
$NY \rightarrow M \rightarrow D \rightarrow C \rightarrow NY = 4,407$

MON, 27-05

Single Source Shortest Path (Bellman Ford)

$$dist^k[u] = \min\left\{ dist^{k-1}[u], \min_i \left\{ dist^{k-1}[i] + cost[i,u] \right\} \right\}$$

Q.



Soln

$$dist^k[u] = \min\left\{ dist^{k-1}[u], \min_i \left\{ dist^{k-1}[i] + cost[i,u] \right\} \right\}$$

k	1	2	3	4	5	6	7
1	0	6	5	5	∞	∞	∞
2	0	3	3	5	5	4	∞
3	0	1	3	5	5	4	7
4	0	1	3	5	0	4	5
5	0	1	3	5	0	4	3
6	0	1	3	5	0	4	3
7	0						

Teacher's Signature: ....

$$\text{dist}^2[2] = \min\left\{ \text{dist}^1[2],\ \min\left\{ \begin{array}{l} \text{dist}^1[3] + \text{cost}[3,2] \\ \text{dist}^1[4] + \text{cost}[4,2] \\ \text{dist}^1[5] + \text{cost}[5,2] \\ \text{dist}^1[6] + \text{cost}[6,2] \\ \text{dist}^1[7] + \text{cost}[7,2] \end{array} \right. \right.$$

$$= \min\left\{ 6,\ \min\left\{ \begin{array}{l} 5 + (-2) \\ 5 + \infty \\ \infty + \infty \\ \infty + \infty \\ \infty + \infty \end{array} \right. \right.$$

$$= \min(6, 3)$$
$$= 3$$

$$\text{dist}^2[3] = \min\left\{ \text{dist}^1[3],\ \min\left\{ \begin{array}{l} \text{dist}^1[2] + \text{cost}[2,3] \\ \text{dist}^1[4] + \text{cost}[4,3] \\ \text{dist}^1[5] + \text{cost}[5,3] \\ \text{dist}^1[6] + \text{cost}[6,3] \\ \text{dist}^1[7] + \text{cost}[7,3] \end{array} \right. \right.$$

$$= \min\left\{ 5,\ \min\left\{ \begin{array}{l} 6 + \infty \\ 5 + (-2) \\ \infty \\ \infty \\ \infty \end{array} \right. \right.$$

$$= \min(5, 3)$$
$$= 3$$

$$\text{dist}^2[4] = \min\left\{ \text{dist}^1[4],\ \min\left\{ \begin{array}{l} \text{dist}^1[2] + \text{cost}[2,4] \\ \text{dist}^1[3] + \text{cost}[3,4] \\ \text{dist}^1[5] + \text{cost}[5,4] \\ \text{dist}^1[6] + \text{cost}[6,4] \\ \text{dist}^1[7] + \text{cost}[7,4] \end{array} \right. \right.$$

$$= \min\left\{ 5, \min\begin{cases} 6+\infty \\ 5+\infty \\ \infty \\ \infty \\ \infty \end{cases} \right.$$

$= \min(5, \infty)$

$= 5$

$$dist^2[5] = \min\left\{ dist^1[5], \min\begin{cases} dist^1[2] + cost[2,5] \\ dist^1[3] + cost[3,5] \\ dist^1[4] + cost[4,5] \\ dist^1[6] + cost[6,5] \\ dist^1[7] + cost[7,5] \end{cases} \right.$$

$$= \min\left\{ \infty, \min\begin{cases} 6+(-1) \\ 5+(1) \\ 5+\infty \\ \infty+\infty \\ \infty+\infty \end{cases} \right.$$

$= \min(\infty, 5)$

$= 5$

$$dist^2[6] = \min\left\{ dist^1[6], \min\begin{cases} dist^1[2] + cost[2,6] \\ dist^1[3] + cost[3,6] \\ dist^1[4] + cost[4,6] \\ dist^1[5] + cost[5,6] \\ dist^1[7] + cost[7,7] \end{cases} \right.$$

$$= \min\left\{ \infty, \min\begin{cases} 6+\infty \\ 5+\infty \\ 5+(-1) \\ \infty \\ \infty \end{cases} \right.$$

$= \min(\infty, 4) = 4$

$$dist^2[7] = \min \left\{ dist^1[7],\ \min \begin{cases} dist^1[2] + cost[2,7] \\ dist^1[3] + cost[3,7] \\ dist^1[4] + cost[4,7] \\ dist^1[5] + cost[5,7] \\ dist^1[6] + cost[6,7] \end{cases} \right.$$

$$= \min \left\{ \infty,\ \min \begin{cases} 6 + \infty \\ 5 + \infty \\ 5 + \infty \\ \infty \\ \infty \end{cases} \right.$$

$$= \min(\infty, \infty) = \infty$$

$$dist^3[2] = \min \left\{ dist^2[2],\ \min \begin{cases} dist^2[3] + cost[3,2] \\ dist^2[4] + cost[4,2] \\ dist^2[5] + cost[5,2] \\ dist^2[6] + cost[6,2] \\ dist^2[7] + cost[7,2] \end{cases} \right.$$

$$= \min \left\{ 3,\ \min \begin{cases} 3 + (-2) \\ 5 + \infty \\ 5 + \infty \\ 4 + \infty \\ \infty \end{cases} \right.$$

$$= \min(3, 1) = 1$$

$$dist^3[3] = \min \left\{ dist^2[3],\ \min \begin{cases} dist^2[2] + cost[2,3] \\ dist^2[4] + cost[4,3] \\ dist^2[5] + cost[5,3] \\ dist^2[6] + cost[6,3] \\ dist^2[7] + cost[7,3] \end{cases} \right.$$

$$= \min \left\{ 3,\ \min \begin{cases} 3 + \infty \\ 5 + (-2) \\ 5 + \infty \\ 4 + \infty \\ \infty \end{cases} \right. = \min\{3, 3\} = 3$$

$(1,2) = (1-4-3-2) = 1 \qquad k = 1$

$(1,3) = (1-4-3) = 3$

$(1,4) = (1-4) = 5$

$(1,5) = (1-3-2-5) = 7$

$(1,6) = (1-4-6) = 2$

$(1,7) = (1-4-3-2-5-7) = 3$

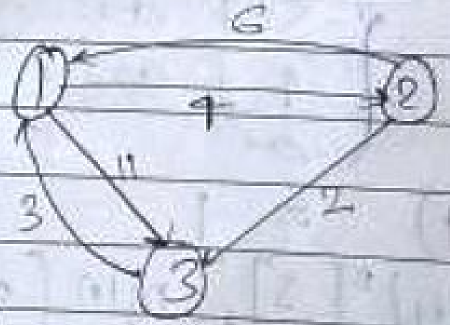
(FRI, 31-03)

Bellman Ford - Dynamic Approach
Dijkstra - Greedy Approach

All Pair Shortest Path

Recursive Formula

$A^k(i,j) = \min\{A^{k-1}(i,j),\ A^{k-1}(i,k) + A^{k-1}(k,j)\}$ $k, k=$



$i = j$ then 0
i, j weight
i, j no weight ∞

$k=0$

$$A^0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \left[\begin{array}{ccc} 0 & 4 & 11 \\ 6 & 0 & 2 \\ 3 & \infty & 0 \end{array}\right] \end{array}$$

$k = 1$

$$A^1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array}\begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 0 & 4 & 11 \\ 6 & 0 & 2 \\ 3 & 7 & 0 \end{bmatrix} \end{array}$$

(i, j)
(1, 2)

i k j
1 — 1 — 2

(i, j)
(1, 3)

i k j
1 — 1 — 3

(i, j)
(2, 1)

i k j
2 — 1 — 1

(i, j)
(2, 3)

i k j
2 — 1 — 3
(6 + 11 = 17, not accepted)

(i, j)
(3, 1)

i k j
3 — 1 — 1

(i, j)
3, 2

i k j
3 — 1 — 2

$k = 2$

$$A^2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array}\begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 0 & 4 & 6 \\ 6 & 0 & 2 \\ 3 & 7 & 0 \end{bmatrix} \end{array} \qquad A^3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \end{array}\begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 0 & 4 & 6 \\ 5 & 0 & 2 \\ 3 & 7 & 0 \end{bmatrix} \end{array}$$

(i, j)
(1, 2)

i — k — j
1 2 2
(4)

(i, j)
(1, 3)

i — k — j
1 2 3
(6)

(i, j) $(2, 1)$

i — k — j
2 — 2 — 1
(6)

(i, j) $(3, 1)$

i — k — j
3 — 2 — 1
(∞)

k = 3
(i, j) $(1, 2)$

Not possible

(i, j) $(2, 1)$

~~Not possible~~

i — k — j
2 — 3 — 1
(5)
Accepted

(i, j) $(2, 3)$

i — k — j
2 — 2 — 3
(2)

(i, j) $(3, 2)$

i — k — j
3 — 2 — 2
(∞)

(i, j) $(1, 3)$

i — k — j
1 — 8 — 3
(11 — not accepted)

(i, j) $(2, 3)$

i — k — j
2 — 3 — 3
(2)

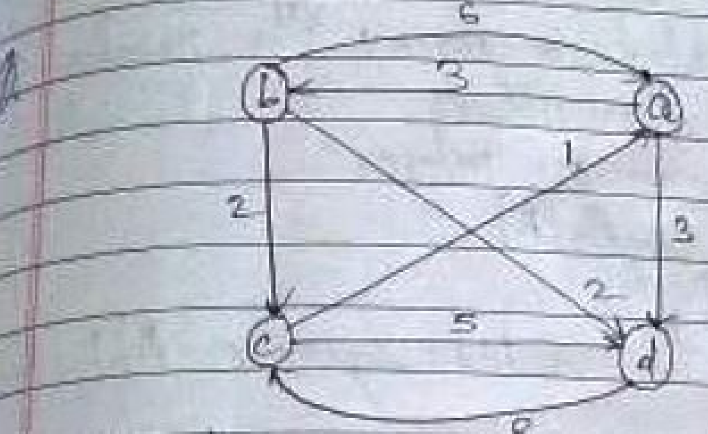
(1, 1)
(3, 1)

(1, 1)
(3, 2)

i k j
3 — 3 — 1
(3)

i k j
3 — 3 — 2
(∞)

(1,2) → (1 → 2) = 4 , (1,3) → (1 → 2 → 3) = 6

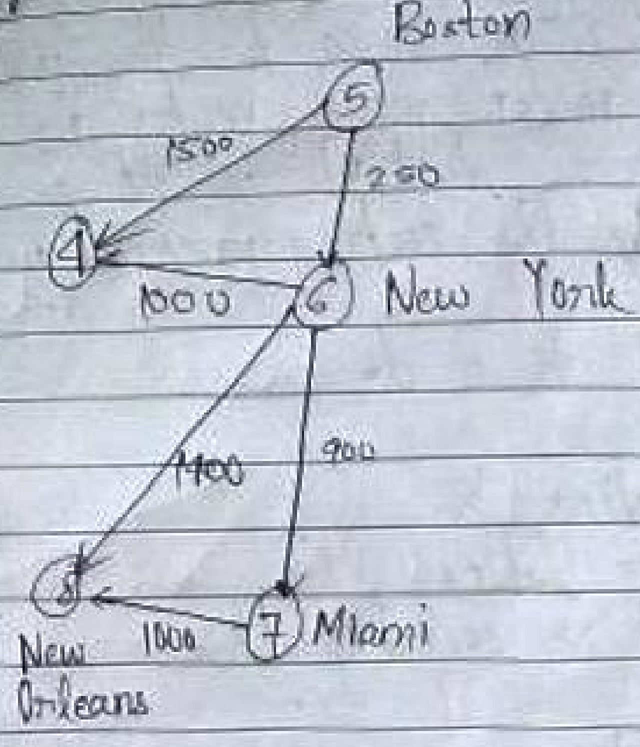


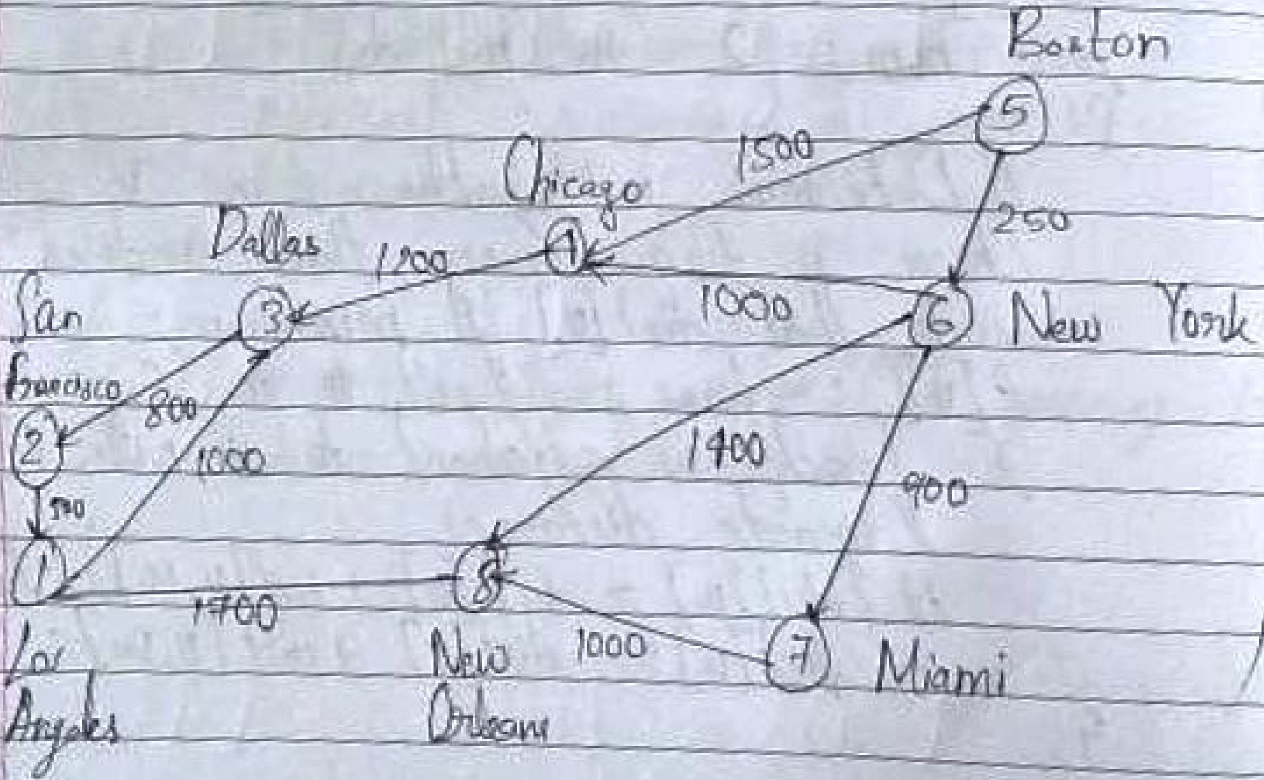
$A^k(i,j) = \min\{A^{k-1}(i,j),\ A^{k-1}(i,k) + A^{k-1}(k,j)\},\ k=1$

$k = 0$

$$A^0 = \begin{array}{c|cccc} & a & b & c & d \\ \hline a & 0 & 3 & \infty & 3 \\ b & 6 & 0 & 2 & 2 \\ c & 1 & \infty & 0 & 5 \\ d & \infty & \infty & 8 & 0 \end{array}$$

Expt- 4


Boston
5
1500
250
4
1000
6
New York
1400
900
8
7
1000
Miami
New Orleans


Boston
5
1500
Chicago
250
Dallas
1200
4
1000
San Francisco
3
6
New York
800
2
1000
1400
900
500
1
1700
8
1000
7
Miami
Los Angeles
New Orleans

Assuming the starting vertex to be 5 (Boston)

S	Vertex Selected	5	1	2	3	4	6	7	8

	S	Vertex Selected	5	1	2	3	4	6	7	8
1	5	6	0	∞	∞	∞	1500	250	∞	∞
2	5,6	7	0	∞	∞	∞	1250	250	1,150	1,650
3	5,6,7	4	0	∞	∞	2,450	1,250	250	1,150	1,650
4	5,6,7,4	8	0	∞	∞	2,450	1,250	250	1,150	1,650
5	5,6,7,4, 8	3	0	3,350	∞	2,450	1,250	250	1,150	1,650
6	5,6,7,4, 8,3	2	0	3,350	3,250	2,450	1,250	250	1,150	1,650
7	5,6,7,4, 8,3,2	1	0	3,350	3,250	2,450	1,250	250	1,150	1,650
8	5,6,7,4, 8,3,2,1	-	0	3,350	3,250	2,450	1,250	250	1,150	1,650

for iteration 1,

dist [4] = dist[5] + cost [5,4]
= 0 + 1500
= 1500

dist [6] = dist [5] + cost [5,6]
= 0 + 250
= 250

for iteration 2,

dist[9] = dist[6] + cost[6,9]
= 250 + 1000
= $\boxed{1,250}$ < 1500

dist[7] = dist[6] + cost[6,7]
= 250 + 900
= 1,150

dist[8] = dist[6] + cost[6,8]
= 250 + 1,400
= 1,650

for iteration 3,

dist[3] = dist[9] + cost[9,3]
= 1,250 + 1,200
= 2,450

dist[8] = dist[7] + cost[7,8]
= 1,150 + 1,000
= 2,150 > 1,650

for iteration 4,

dist[3] = dist[9] + cost[9,3]
= 1,250 + 1,200
= 2,450

For iteration 5,

dist[1] = dist[8] + cost[8, 1]
= 1,650 + 1,700
= 3,350

For iteration 6,
dist[2] = dist[3] + cost[3, 2]
= 2,450 + 800
= 3,250

For iteration 7,
dist[1] = dist[2] + cost[2, 1]
= 3,250 + 300
= 3,550 > 3,350

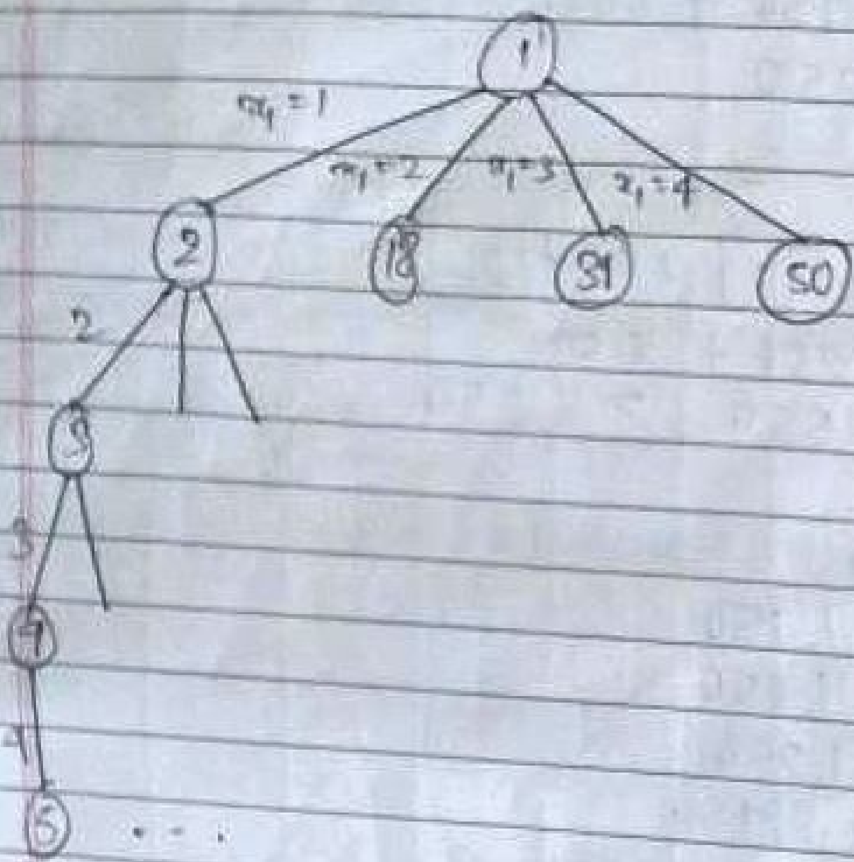
5-6 = 250
5-6-7 = 1,150
5-6-8 = 1,650
5-6-4 = 1,250
5-6-4-3 = 2,450
5-6-4-3-2 = 3,250
5-6-8-1 = 3,350

(Mon, 10 - Apr)

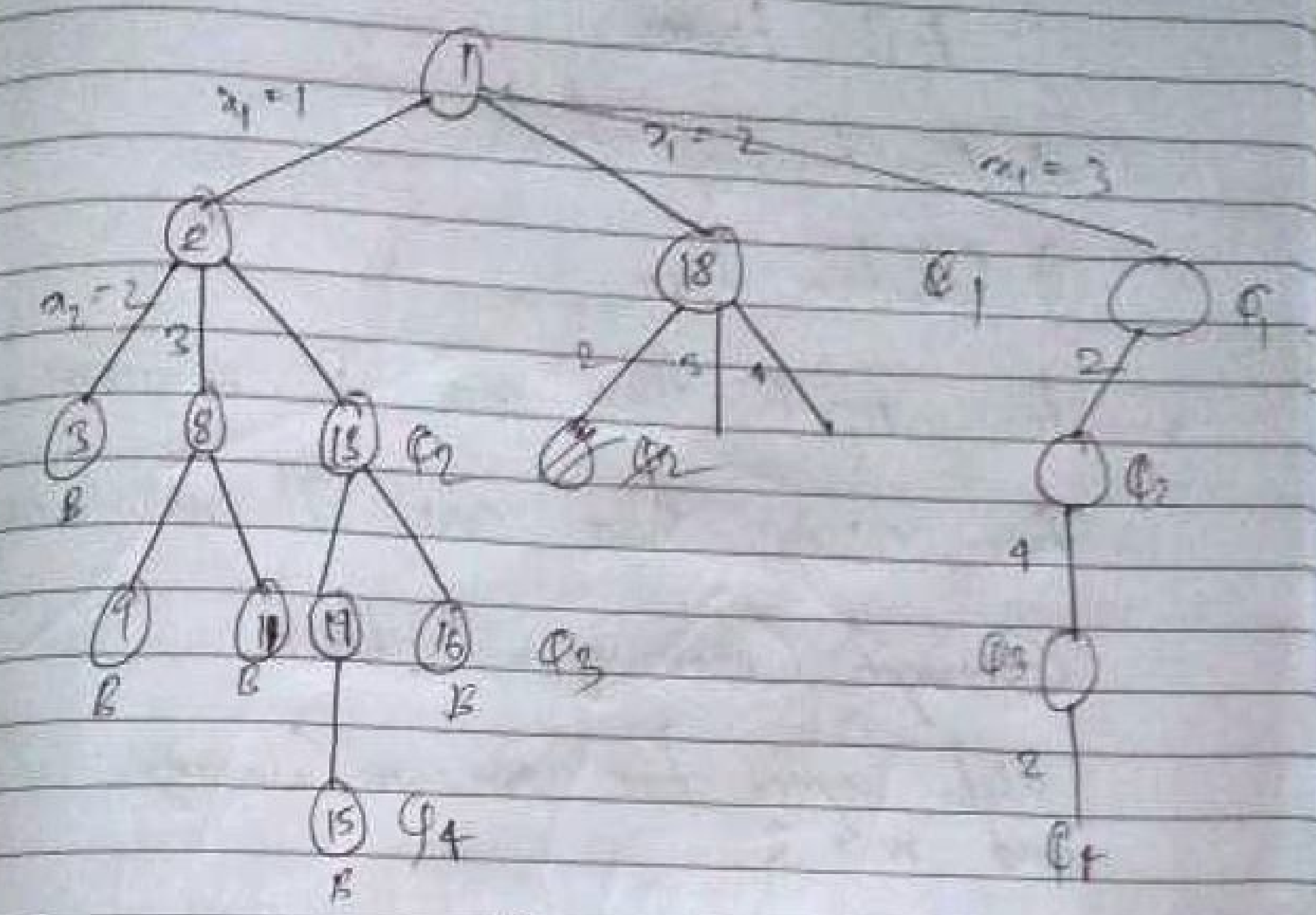
Backtracking

N Queen

Generic Problem



Partial Backtracking Tree



For $x_1 = 2$

	1	2	3	4
1		Q_1		
2	X	X	X	Q_2
3	Q_3	X	X	X
4			Q_4	

$\{2, 4, 1, 3\}$

For $x_1 = 3$

	1	2	3	4
1			Q_1	
2	Q_2	x	x	x
3	x	x	x	Q_3
4	x	Q_4	x	x

$\{3, 1, 4, 2\}$

Three Trees :

1) Full tree

2) Backtracking tree

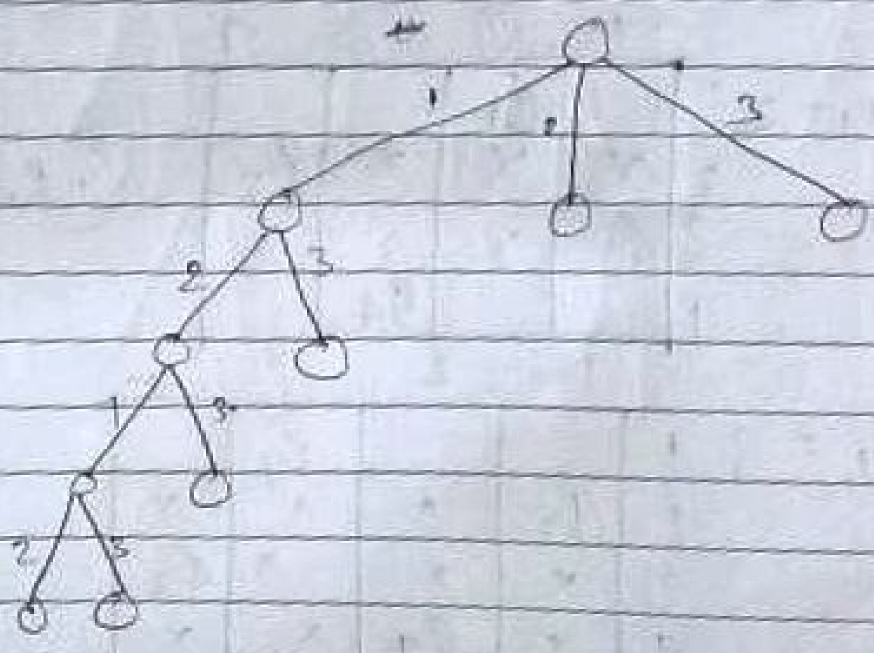
3) Solution tree

For n=4 there are 2 solutions

For n=8, there are 91 solutions

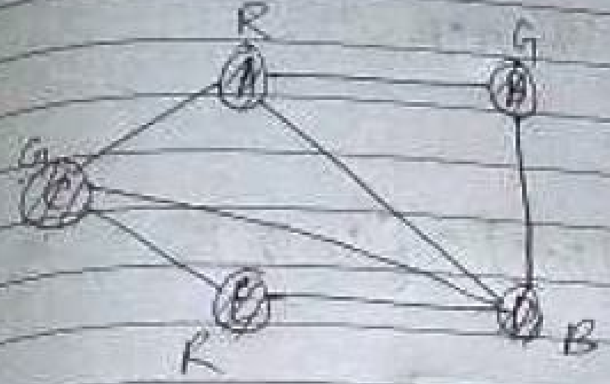
Graph Colouring Problem

State space tree for m Colouring when n=3 and m=3



Define the problem ⇒ First slide

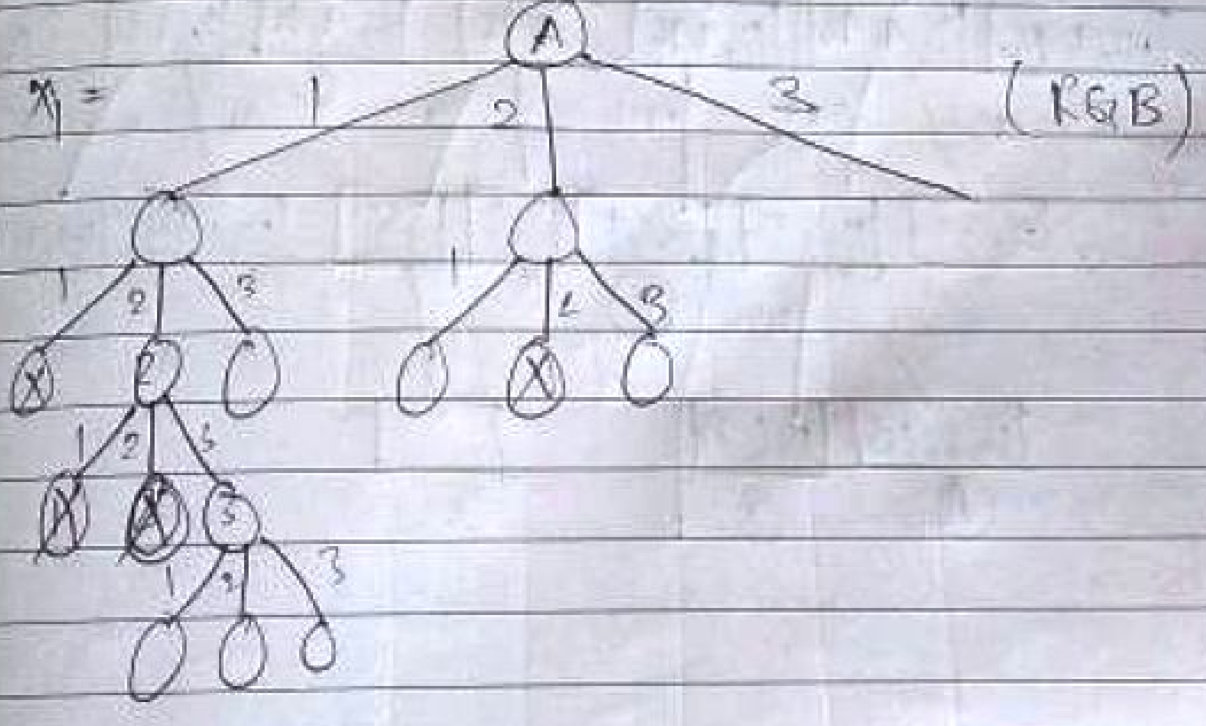
A.



m = 3

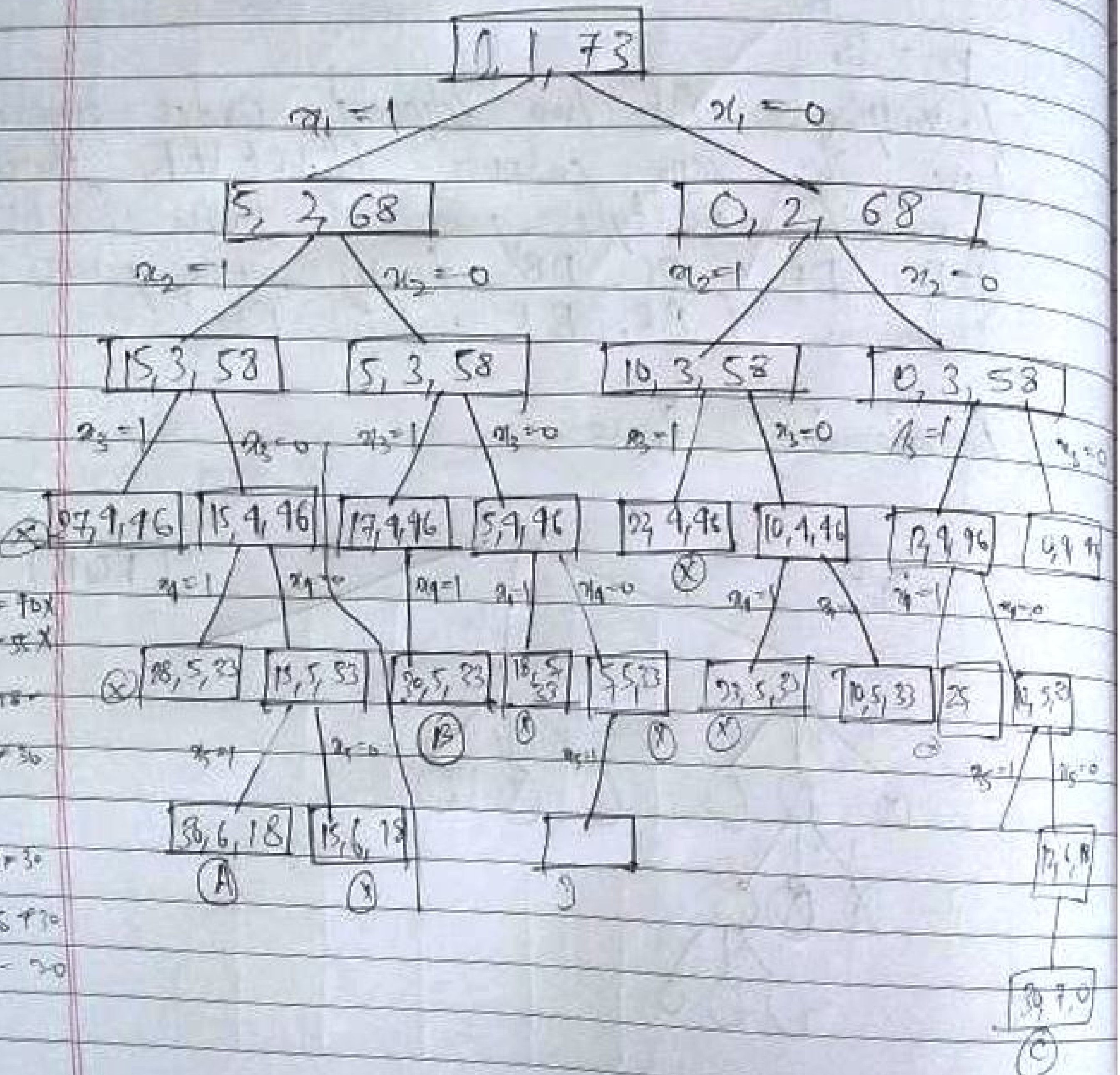
Principle : No two adjacent nodes should have the same colour. HOWEVER, there can be multiple pairs of nodes (like DA, DE ; DC, DB having same pairs of colours (BR, BR ; BG BG)

Backtracking Tree



(Thu, 13-04)

Q. $n = 6$, $m = 30$, $w[1:6] = \{5, 10, 12, 13, 15, 18\}$



27 + 13 = 40 X

40 + 15 = 55 X

15 + 13 = 28

28 + 15 ≠ 30

15 + 18 = 30

18 + 15 ≠ 30

5 + 15 = 20