0/1 Knapsack problem using Branch & Bound

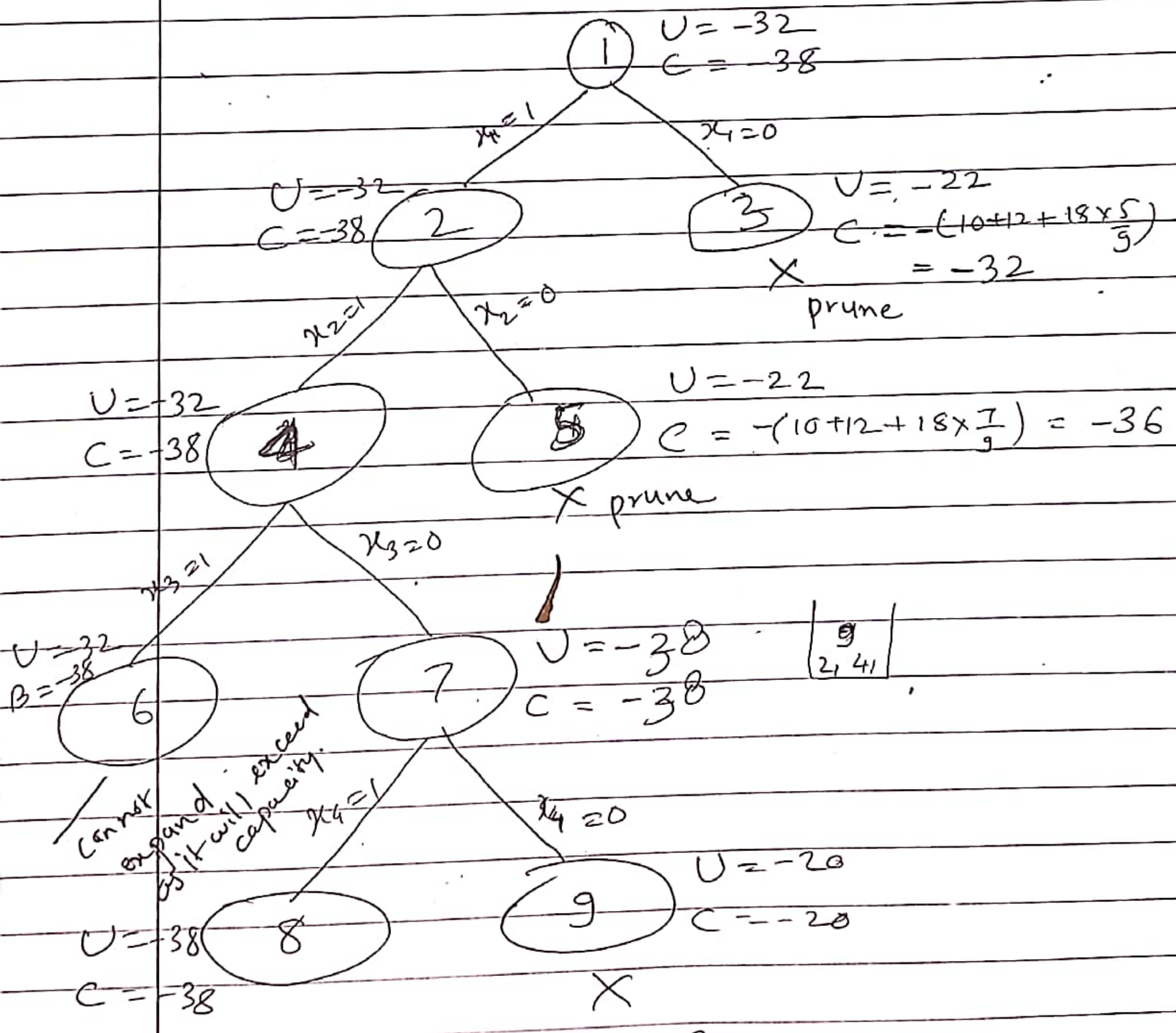
$n = 4$, Profit: $\{10, 10, 12, 18\}$
$M = 15$, Weight: $\{2, 4, 6, 9\}$

$\cancel{U = 32}$

Initial upper bound $= 32$ $\boxed{2,4,6}$
(by adding first 3 elements in bag)

$\boxed{2,4,6,3/9}$ Cost function:- $10 + 10 + 12 + 18 \times 3/9$
$C = 38$

— Negate the values.



Hence soln is $\{1, 1, 0, 1\}$

If entire tree is explored complexity is $\cancel{O(2^n)}$ $O(2^n)$.
best case possible ~~if~~ only one path explored. $O(n)$

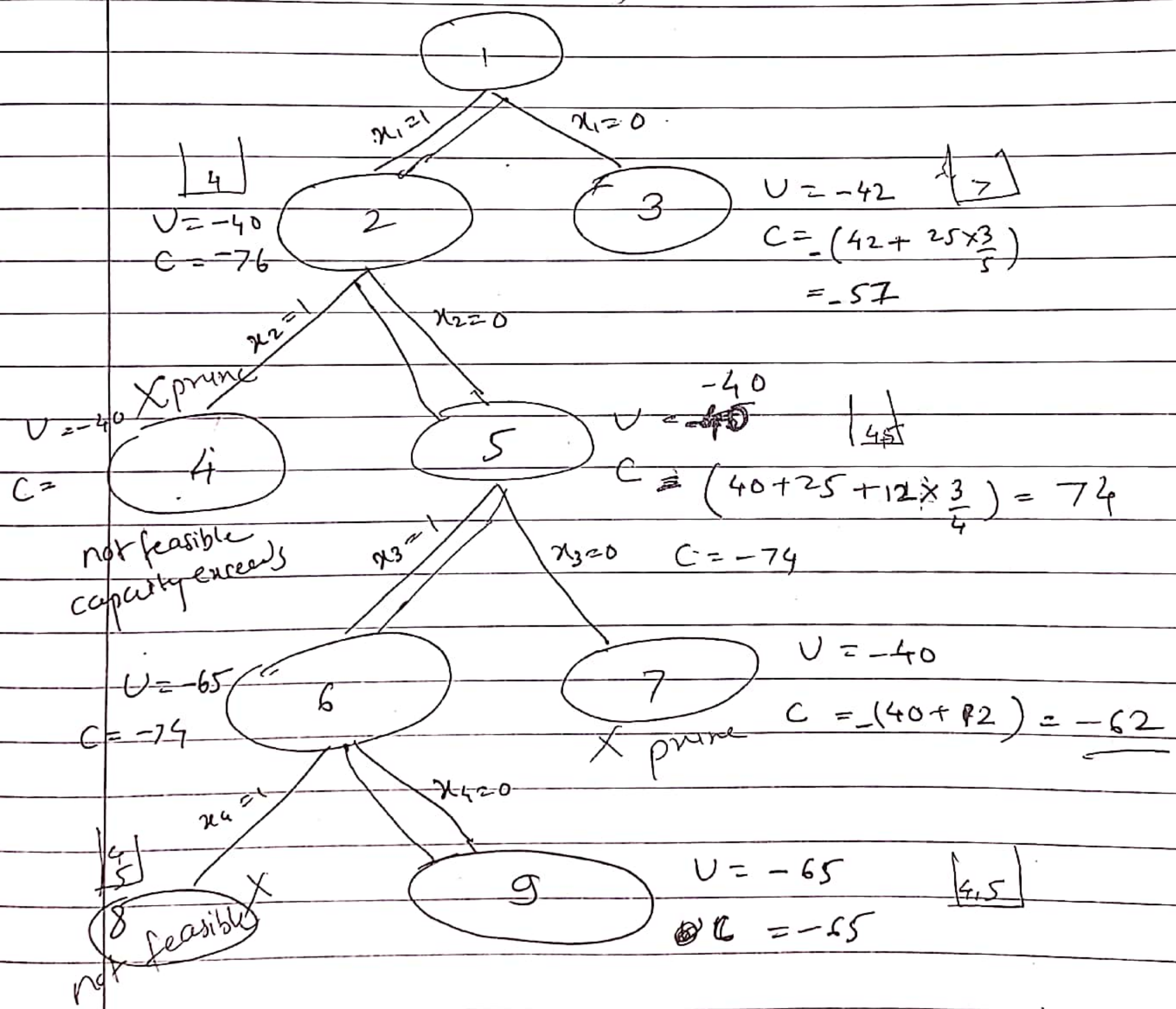
$n=4$ Profit $\{40, 42, 25, 12\}$

$M=10$ Weight $\{4, 7, 5, 3\}$

$\frac{P}{W}$: $\{10, 6, 5, 4\}$

Initial upper bound U : 40

Cost: $40 + 42 \times \frac{6}{7}$



1/18/35,000

K. J. SOMAIYA COLLEGE OF ENGINEERING

(Autonomous College Affiliated to University of Mumbai)

Candidate Roll No. ____________
(In figures)

Test Exam.

Name : ____________

Date : ____________ 20

Examination : ____________ Branch/Semester ____________

Subject : ____________

Junior Supervisor's full Signature with Date

Question No.	1	2	3	4	5	6	7	8	9	10	11	12	Total
Marks Obtained													

8- Queens Problem

1 → no 2 queens in a row
2 → no 2 queens in a column
3 → no 2 queens in left diagonal
4 → no 2 queens in right diagonal

→ Matrix is filled column by column.
Hence we need to satisfy remaining 3 conditions.

Apply Branch & Bound approach.

→ create 3 arrays to check conditions 1, 2 & 3 4.

→ Develope a numbering system to specify which queen is placed.

Preprocess

Create 2 NXN matrices,
one for 1 top-left to bottom right
2 Top right to bottom-left

Fill them in such manner that 2 queens sharing same TL-BR diagonal will have same value in matrix 1 and 2 queens sharing same top-right - bottom-left diagonal will have same value in matrix 2.

matrix 1

	0	1	2	3	4	5	6	7
0	7	6	5	4	3	2	1	0
1	8	7	6	5	4	3	2	1
2	9	8	7	6	5	4	3	2
3	10	9	8	7	6	5	4	3
4	11	10	9	8	7	6	5	4
5	12	11	10	9	8	7	6	5
6	13	12	11	10	9	8	7	6
7	14	13	12	11	10	9	8	7

matrix 2

0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	8
2	2	3	4	5	6	7	8	9
3	3	4	5	6	7	8	9	10
4	4	5	6	7	8	9	10	11
5	5	6	7	8	9	10	11	12
6	6	7	8	9	10	11	12	13
7	7	8	9	10	11	12	13	14

mat1[row][col] = row - col + (N-1)

mat2[row][col] = row + col.

To place queen check,

1. whether row 'j' is used or not
2. whether i+j is used or not
3. whether i-j+7 is used or not.

→ if answer is True, try another location for queen i on row j. ~~and di~~

→ mark row and diagonals

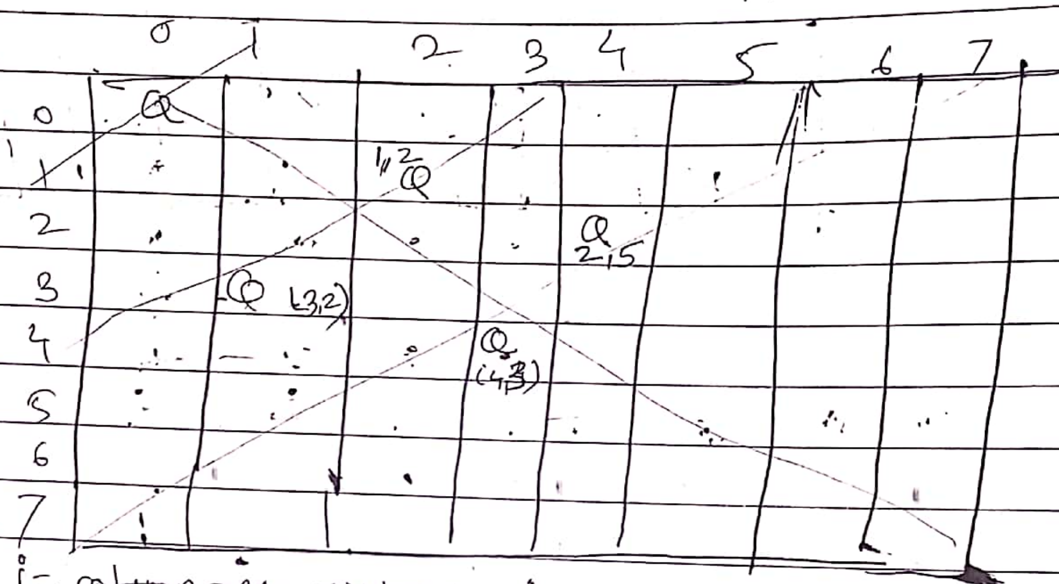
→ recur for queen i+1

row 0

(size n) row_bool_value [F F F F F F F F]

size 2(n-1) mat1_dia_bool [ ]

size 2*(n-1) mat2_dia_bool [ ]



i = ~~column~~ = 0 ; ~~row = 0~~ = j col = 0

i = 0

→ $\overset{col}{\cancel{row}}$_bool_val [T F F F F F F F]

mat1_dia_bool [i] [col] = T

0 0

mat 2_dia_bool [0] [col] = T

[0,0]

chk row = 1 (i) & col [0] ie ($\overset{1,0}{\cancel{0,1}}$) position.

row1 col_bool_val [T F T F F F F F]

row2 [T F T F T F F F]

row3 → [T T T F T F F F]

row4 → T T T T T F F F

0 1 2 3 4

bool array [F F F F ~~F~~]

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

	0	1	2	3
0	Q			
1			Q	
2				
3				

row [0] - (0,0)

col- bool-array [T F F F]

row [1] [1,0] X same column

[1,1] X - mat 1

[1,2]

col- bool array [T F T F]

0 1 2 3

row [2] , [2,0] - same colu.

[2,1] X mat 2

[2,2] X same colu

[2,3] X fun mat 1

Backtrack → col- bool- array [T F F F]

Q			

Changes in mat 1 & mat 2 as well.

row (1,3)

Mat1

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

Mat2

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

check row(1,3)

	0	1	2	3
0	Q			
1				Q
2				
3				

all are false
Hence place
at (1,3)

col-bool-array = [T F F T]

row (2,0) — sam col.
row (2,1) — possible to place
as false in all 3, bool- mat1, mat2
array

Q			
			Q
	Q		

Mat1

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

Mat2

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

→ row (3,0) — X same col
(3,1) X same col
(3,2) X not possible from Mat1
(3,3) X same col.

Hence backtrack
from row 3 — all posi X not possible

from row 2, (2,2) — not allowed from Mat1
(2,3) X same col.

no further for → row1 →

row 0 → earlier Queen at (0,0)
need to explore further.
reset array, mat1, mat2.
col_bool_array [F F F F]

mat 1

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

mat 2

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

Soln

	0	1	2	3
0		Q		
1				
2				
3				

row (0,1) – possible with all false at array, mat1, mat2

~~Top left~~

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

TR

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

BR BL

Bool array
[F T F F]

row (1,0) → X by mat2 (no)
row (1,1) → X by array
row (1,2) → X by mat 1
row (1,3) → possible
place Queen at (1,3)

mat1

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

mat2

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

array
[F T F F]
0 1 2 3

Soln

	Q		
			Q

row (2,0)

$\rightarrow$ OK by col_array, mat1, mat2

place Queen at (2,0).

$\Downarrow$

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

[T T F T]

Sol$_n$

		Q	
			Q
Q			

$\Downarrow$

row(3,0) – X by same col array.
row(3,1) – X by mat1
row(3,2) – OK by col array
– OK by mat1
– OK by mat2

3	2	1	0
4	3	2	1
5	4	3	2
6	5	4	3

0	1	2	3
1	2	3	4
2	3	4	5
3	4	5	6

[? T T T]

Solution

	Q		
			Q
Q			
		Q	

Scanned with CamScanner

Backtracking 0/1 Knapsack problem.

$p = [11, 21, 31, 33, 43, \cancel{53}\ 54, 55, 65]$ $m = 110$

$w = [1, 11, 21, 23, 33, 43, 45, 55]$ $n = 8$

p/w [11, 1.9, 1.47, 1.43, 1.3, 1.23, 1.22, 1.18]

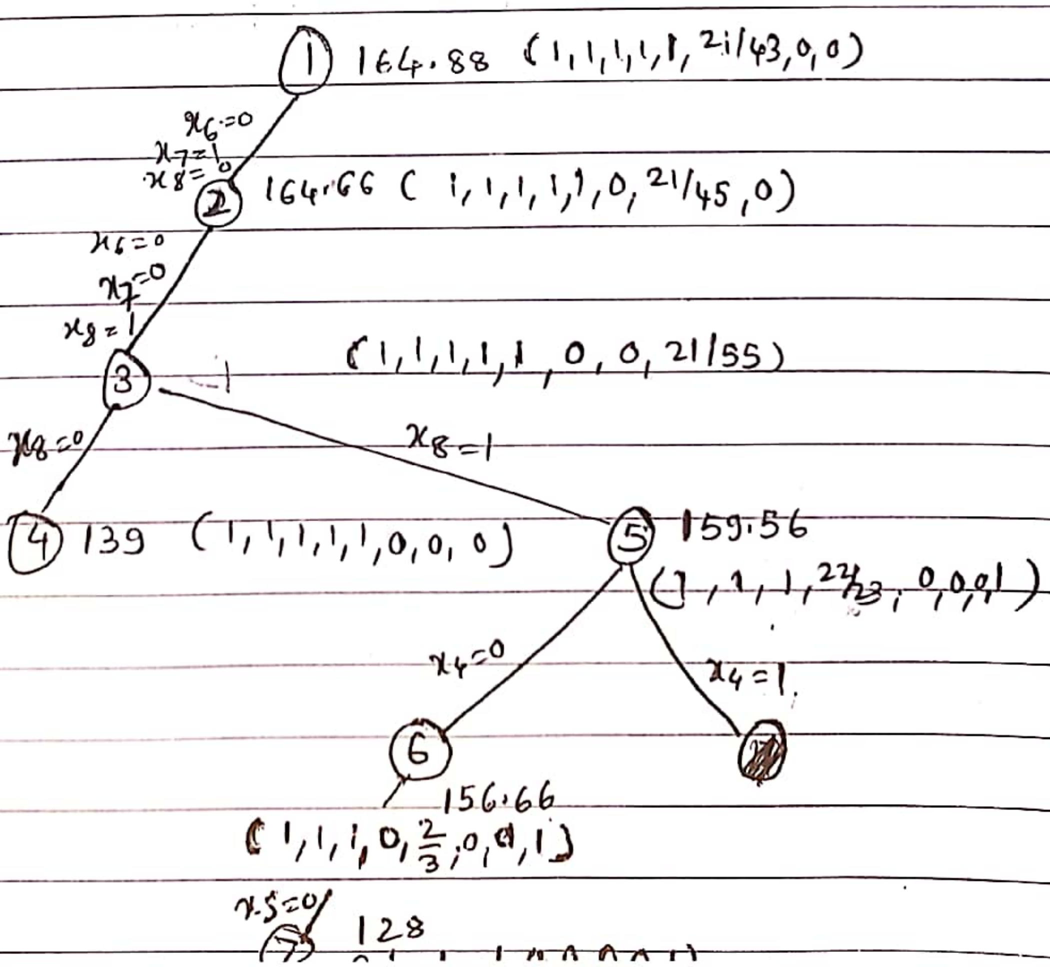
Greedy solution $- x = [1, 1, 1, 1, 1, 21/45, 0, 0]$

1+11+21 + 23+33 = 89 Rem. – 21

Profit $= 11 + 21 + 31 + 33 + 43 + 54 \times \frac{21}{43}$

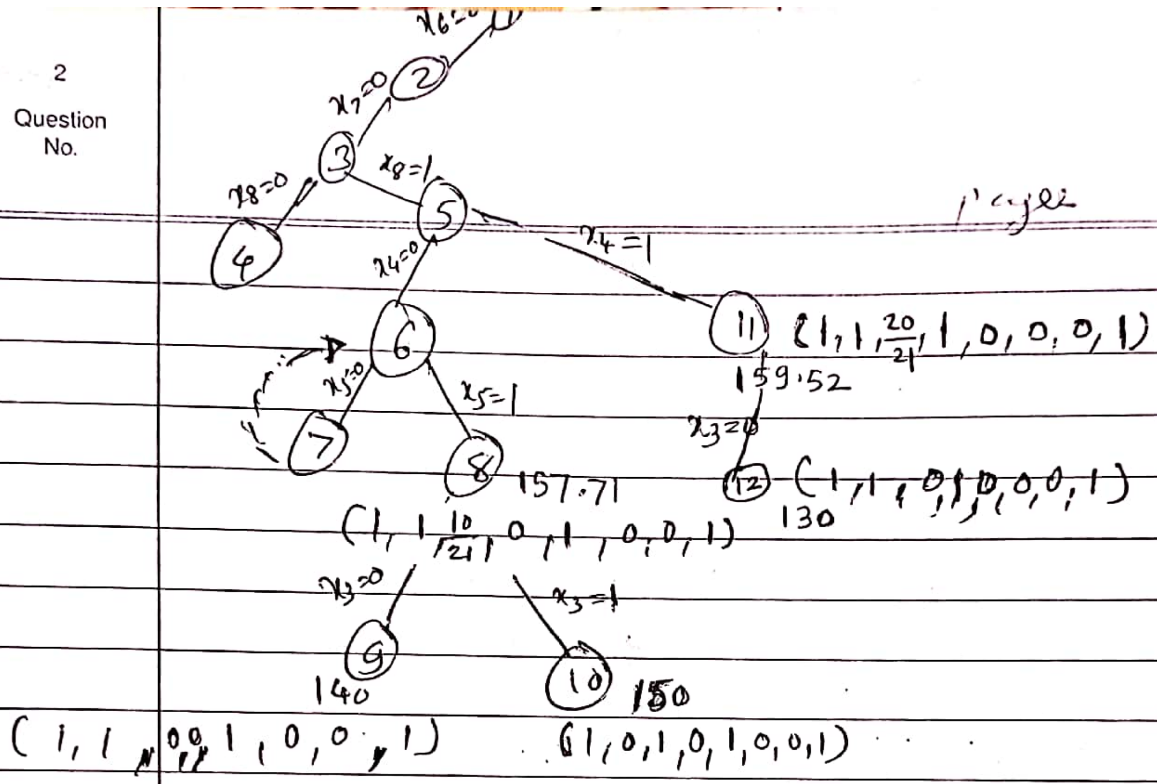
$= 142 + 14.71 = 164.88$

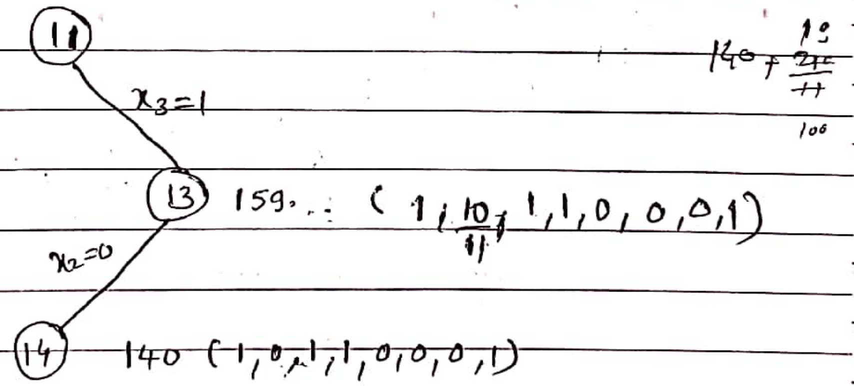
i.e. root node will be generated as,
first 5 objects being placed in knapsack, 6 be placed as 21/43 by greedy approach.



2

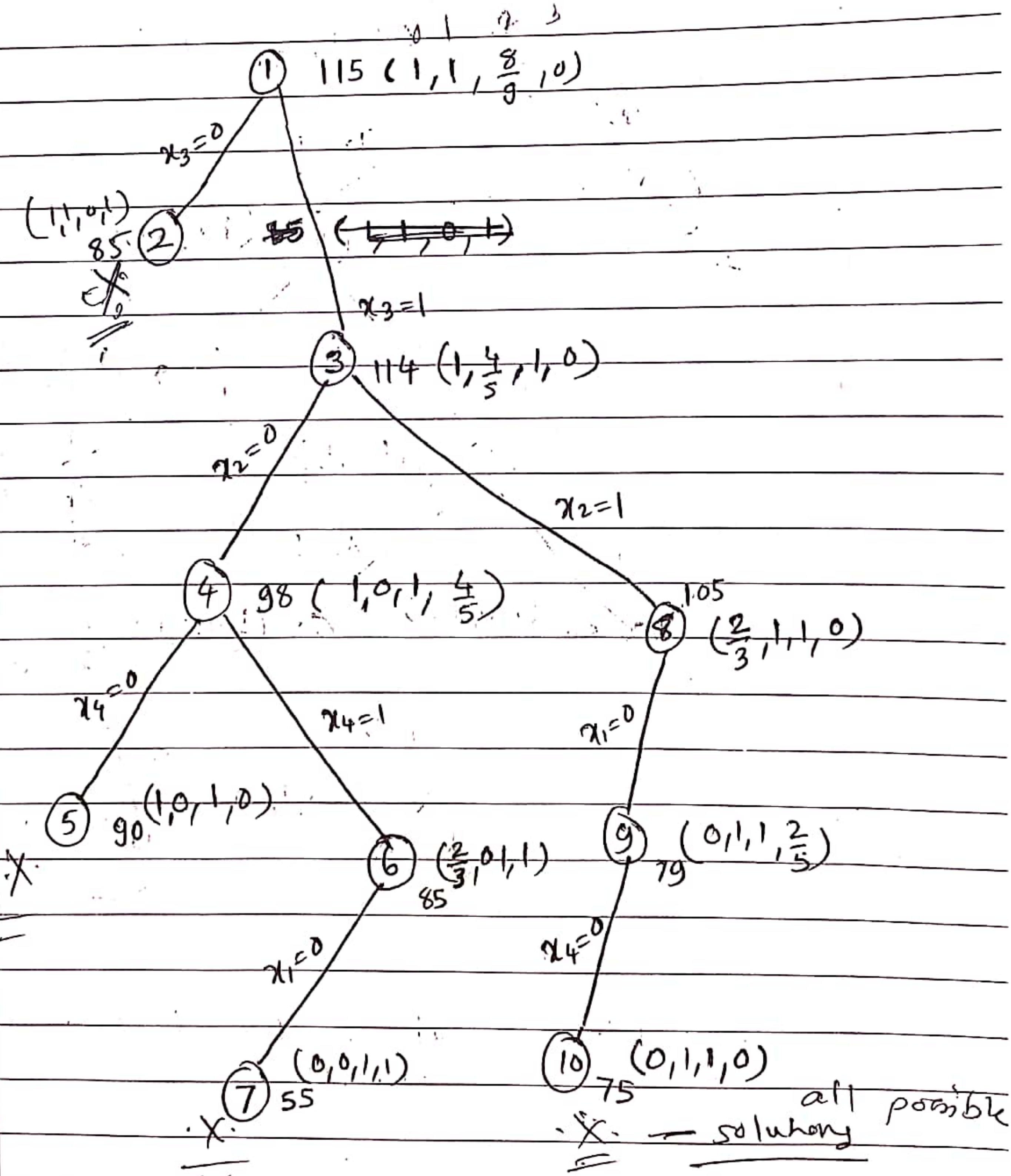
Question No.





$m = 16$, $n = 4$, $P = [45, 30, 45, 10]$ $\frac{P}{w} = [15, 6, 5, 2]$

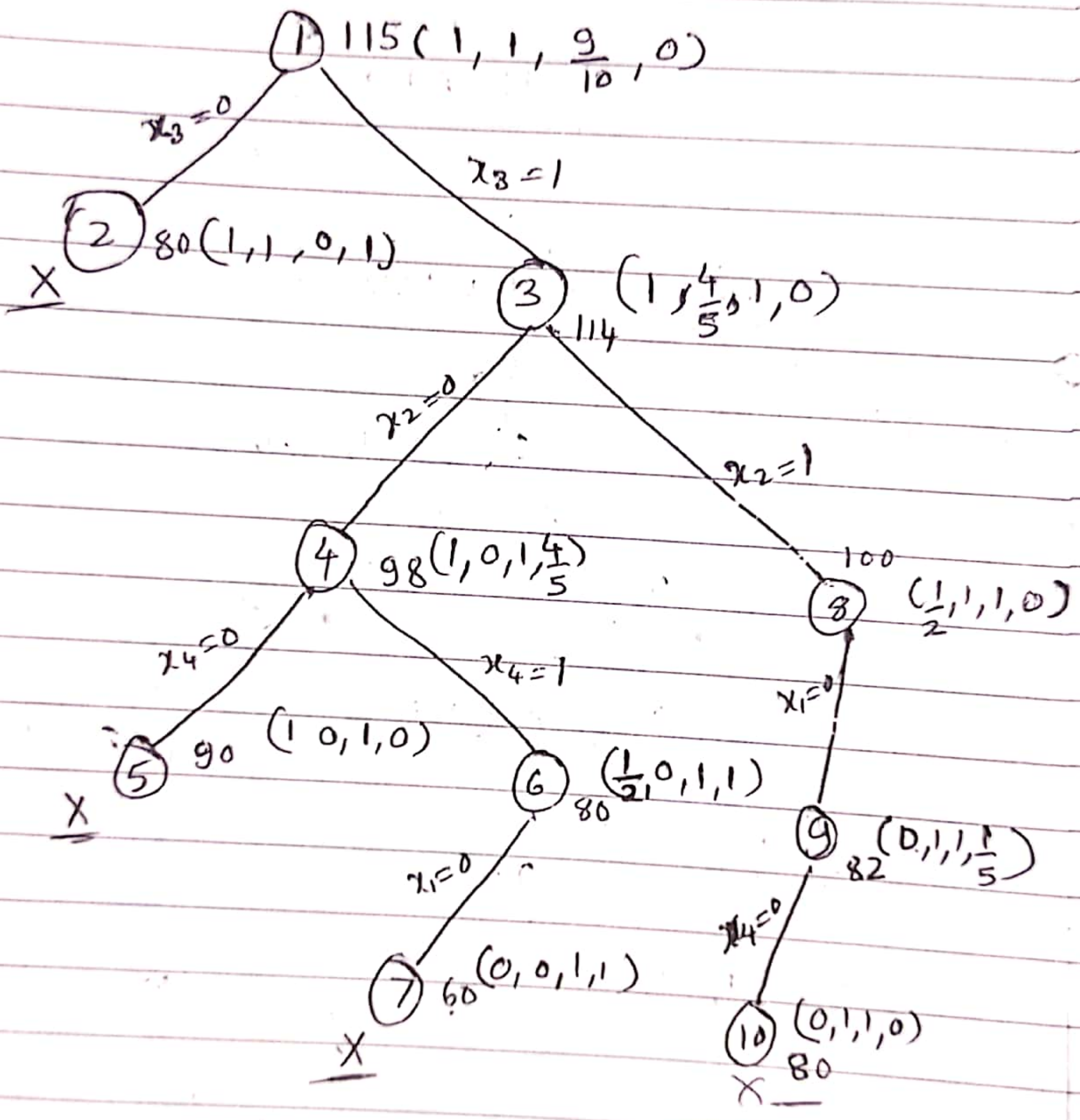
$w = [3, 5, 9, 5]$



page 4

$n=4,\ m=16 \quad P=[40,30,50,10] \quad \frac{p}{w}=[20,6,5,2]$

$w=[2,5,10,5]$



possible solutions by backtracking method.

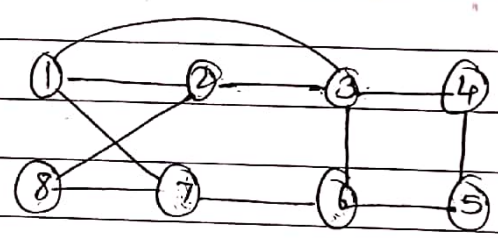
ge4 m
16, 5, 2] ple:

0/1 knapsack backtracking.

p: [25 45 12 7 6 10 5] n=6
w = [5 11 3 2 2 7 4] m=15

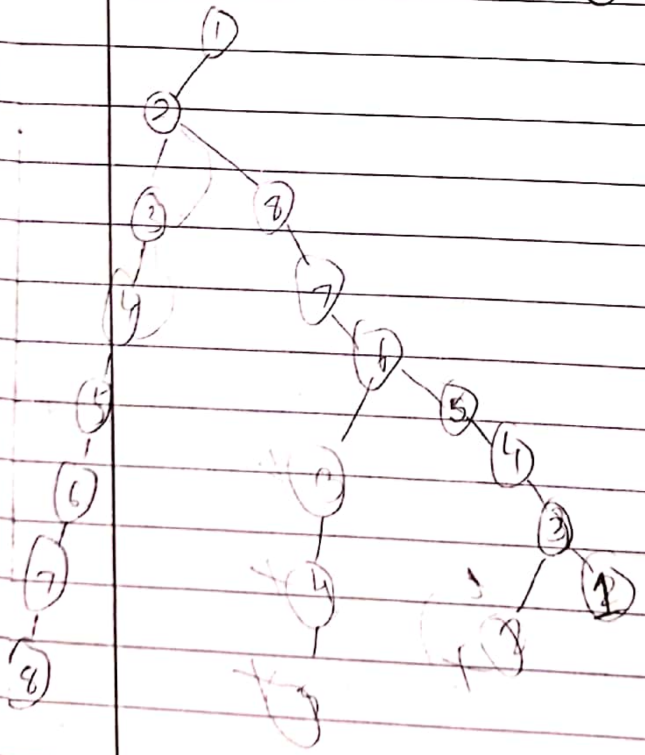
Hamiltonian Cycles – a round trip path along 'n' edges of graph G that visits every vertex once and returns to starting position.
– no easy way to determine whether a graph contains hamiltonian cycle

(1/2,1,1,0)



hamiltonian cycle
1, 2, 8, 7, 6, 5, 4, 3, 1
~~1 7 6 5 4 3 2~~
1 3 4 5 6 7 8 2 1

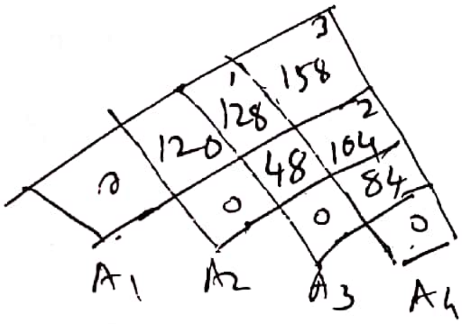
(1,1,1/5)



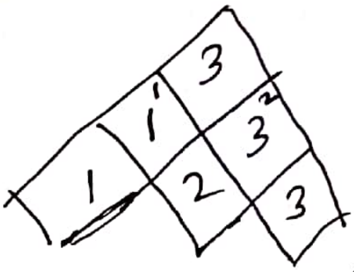
MCM - soln-order

initial set of dimensions ⟨5 4 6 2 7⟩

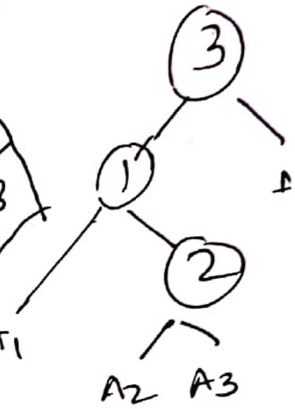
A_1	A_2	A_3	A_4	~~A_5~~
5x4	4x6	6x2	2x7	



m[i,j]



S[i,j]



order $(A_1(A_2A_3))A_4$

① $\overset{5\times4}{(A_1)}\overset{4\times2}{(A_2A_3)} = \underline{40+48}$

✓ $\underset{5\times6}{(A_1A_2)}\underset{6\times2}{(A_3)} = \underline{60+120}$

② $\overset{4\times6}{A_2}\overset{6\times7}{(A_3A_4)} = 168+84$

✓ $\underset{4\times2}{(A_2A_3)}\underset{2\times7}{A_4} = 56+48$

③ $\overset{5\times4}{A_1}\overset{4\times7}{(A_2A_3A_4)} = 140+104 = 244$

$\overset{5\times6}{(A_1A_2)}\overset{6\times7}{(A_3A_4)} = 120+84+210$

~~$\overset{5\times4}{A_1}\overset{4\times2}{(A_2A_3)}\overset{2\times7}{A_4}$~~ $= 48+40+70 = 158$

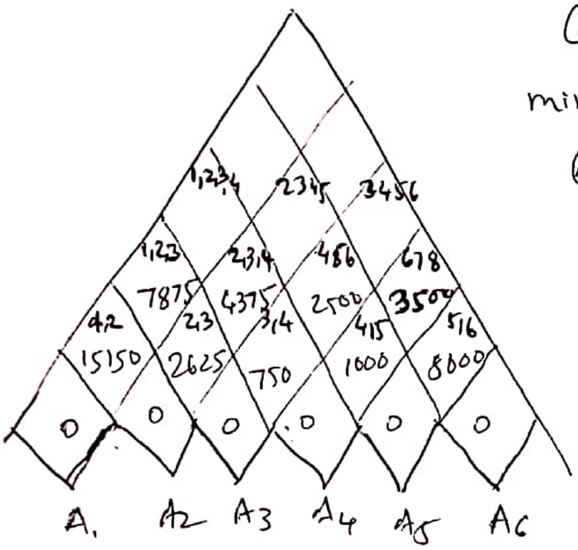
✓ $\underset{5\times2}{(A_1A_2A_3)}\underset{2\times7}{A_4} = 70+128$

$n \leftarrow length[p]-1$
for $i=1$ to n
do $m[i,j]=0$
for $L=2$ to n do
(L is length of subch
for $i=1$ to $n-L+$
do $j=i+L-1$
$m[i,j] = \infty$
for $k=i$ to $j-1$
do
$q \leftarrow m[i,k] + m[k+1,j] + P_{i-1}P_kP_j$
if $q < m[i,j]$
then
$m[i,j] \leftarrow q$
$S[i,j] \leftarrow k$
return m, s

mcm.

matrix chain multiplication

A_1	A_2	A_3	A_4	A_5	A_6
30X35	35X15	15X5	5X10	10X20	20X25



① $(A_1 A_2) A_3$ = ~~7875~~

min

$A_1 (A_2 A_3)$ = ~~4375~~ 7875
$(30 \times 35 \times 5) +$ = 5250 + 2625
2625
= 7875

② $(A_2 A_3) A_4$ = 4375
min
$A_2 (A_3 A_4)$ = 6000

③ $(A_3 A_4) A_5$ = 3750
$A_3 (A_4 A_5)$ = 2500

④ $(A_4 A_5) A_6$ = 3500
$A_4 (A_5 A_6)$ = 5750

35
x 150
5250

applying DP to NP-class →
↓
formulations may not be efficient as no. of subprobs is large (should be poly. number

Dyna. -
substructure
table structure
bottom up comput.

D & C

	Dyna	D & C	Greedy
divide subproblems	Y	Y	n
independent subprb. (subprb share sub-sub prb)	~~n~~	Y	n
guarantee optimal soln	Y	Y	n.
Choose best at ~~each~~ current step			Y

$$\begin{bmatrix} 0 & 10 & 15 & 20 \\ 5 & 0 & 9 & 10 \\ 6 & 13 & 0 & 12 \\ 8 & 8 & 9 & 0 \end{bmatrix}$$

TSP Dynamic

Start & end 1.

$g(2, \phi) = 5 \quad g(3, \phi) = 6, \quad g(4, \phi) = 8$

② $g(2, \{3\}) = C_{23} + g(3, \phi) = 15 \quad (9+6)$

$g(2, \{4\}) = C_{24} + g(4, \phi) = 18 \quad (10+8)$

$g(3, \{2\}) = C_{32} + g(2, \phi) = 13+5 = 18$

$g(3, \{4\}) = C_{34} + g(4, \phi) = 12+8 = 20$

$g(4, \{2\}) = C_{42} + g(2, \phi) = 8+5 = 13$

$g(4, \{3\}) = C_{43} + g(3, \phi) = 9+6 = 15$

③ $g(2, \{3,4\}) = \min\{C_{23} + g(3, \{4\}), \; C_{24} + g(4, \{3\})\} = 25$

$g(3, \{2,4\}) = \min\{C_{32} + g(2, \{4\}), \; C_{34} + g(4, \{2\})\} = 25$

$g(4, \{2,3\}) = \min\{C_{42} + g(2, \{3\}), \; C_{43} + g(3, \{2\})\} = 23$

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

$= \min\{35, 40, 43\}$

$= 35$

PLEDGE
Patni's Learning Edge

TSP - branch & bound.

Rule:- A row is said to be reduced iff it contains at least one zero & all remaining entries are non negative.

A matrix is said to be reduced if every row & column is reduced.

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & \infty & 20 & 30 & 10 & 11 \\ 2 & 15 & \infty & 16 & 4 & 2 \\ 3 & 3 & 5 & \infty & 2 & 4 \\ 4 & 19 & 6 & 18 & \infty & 3 \\ 5 & 16 & 4 & 7 & 16 & \infty \end{array}$$

TSP for 5 cities by B & B method.

row reduction

$$\begin{matrix} \infty & 10 & 20 & 0 & 1 \\ 13 & \infty & 14 & 2 & 0 \\ 1 & 3 & \infty & 0 & 2 \\ 16 & 3 & 15 & \infty & 0 \\ 12 & 0 & 3 & 12 & \infty \end{matrix}$$

applying row & column reduction...

- subtract 10, 2, 2, 3, 4 respectively from row 1 to 5 and then subtract 1, 3 from columns 1 & 3.

Reduced cost matrix is,

$$= \begin{bmatrix} \infty & 10 & 17 & 0 & 1 \\ 12 & \infty & 11 & 2 & 0 \\ 0 & 3 & \infty & 0 & 2 \\ 15 & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & 12 & \infty \end{bmatrix}$$

Total count by which the matrix is reduced is

$L = 25$

node 2 (35)

as path is from 1 to 2,

make row 1 to ∞.

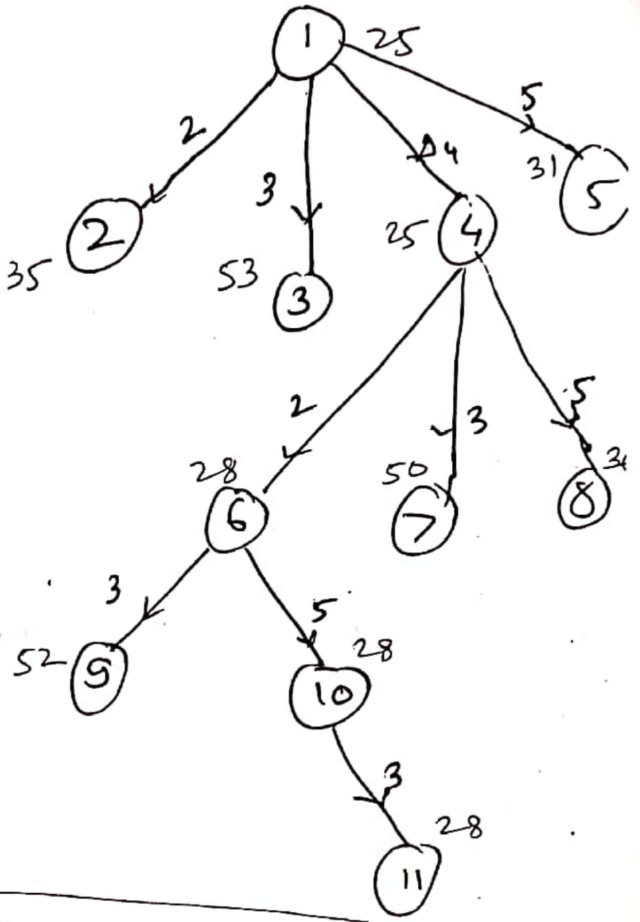
and edges incoming to 1

as ∞. , set (2,1) to ∞

$$
\begin{array}{c|c}
 & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} &
\begin{bmatrix}
\infty & \infty & \infty & \infty & \infty \\
\cancel{12}\ \infty & \infty & 11 & 2 & 0 \\
0 & \infty & \infty & 0 & 2 \\
15 & \infty & 12 & \infty & 0 \\
11 & \infty & 0 & 12 & \infty
\end{bmatrix}
\end{array}
$$

Cost = ~~25~~ Cost of node 1
node 2
+ Cost of (1,2) edge

+ LC reduction cost)



Path 1-4-2-5-3-1
Cost 10+6+2+7+3 = 28.

Cost of node 2 = 25+10+0

= 35

---

node 3 Computation (53)

as path from 1 to 3,

make row 1 to ∞

& edges to 3 to ∞

and reduce matrix.

set (3,1) to ∞

$$
\begin{bmatrix}
\infty & \infty & \infty & \infty & \infty \\
12 & \infty & \infty & 2 & 0 \\
\infty & 3 & \infty & 0 & 2 \\
15 & 3 & \infty & \infty & 0 \\
11 & 0 & \infty & 12 & \infty
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\infty & \infty & \infty & \infty & \infty \\
1 & \infty & \infty & 2 & 0 \\
\infty & 3 & \infty & 0 & 2 \\
4 & 3 & \infty & \infty & 0 \\
0 & 0 & \infty & 12 & \infty
\end{bmatrix}
$$

reduce col 1 by subtracting 11 from it

Cost of reduction = 11

Cost of node 3 = 25 + 17 + 11 = 53

node 4

computation. (25)

as path is from 1 to 4,

row 1 → ∞

incoming to 4 as ∞

and (4,1) → ∞.

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & 11 & \infty & 0 \\ 0 & 3 & \infty & \infty & 2 \\ \infty & 3 & 12 & \infty & 0 \\ 11 & 0 & 0 & \infty & \infty \end{bmatrix}$$

↳ reduced form.

∴ cost of node 4 =

~~25~~ + cost of node 1

+ cost of (1,4)

+ L reduction cost

= 25 + 0 + 0 = 25

node 5 computation. (31)

as path is from 1 to 5

row 1 → ∞

incoming to 5 → ∞

(5,1) → ∞

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \cancel{12}\ 10 & \infty & \cancel{11}\ 9 & \cancel{2}\ 0 & \infty \\ 0 & 3 & \infty & 0 & \infty \\ \cancel{15}\ 12 & \cancel{3}\ 0 & \cancel{12}\ 9 & \infty & \infty \\ \infty & 0 & 0 & 12 & \infty \end{bmatrix}$$

subtract 2 from row 2 & 3 from row 4

cost of node 5 = cost of node 1 + cost (1,5) + L reduction cost

= 25 + 1 + 5

= 31

node 6 computation

path from 4 to 2, $(2,4) \to \infty$

row $4 \to \infty$, column $2 \to \infty$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & 11 & \infty & 0 \\ 0 & \infty & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ 11 & \infty & 0 & \infty & \infty \end{bmatrix}$$

cost of node 6 = cost of node 4 + cost (4,2) + reduction cost $= 25 + 3 + 0 = \underline{28}$

---

node 7: path from 4 to 3, so $(3,4) \to \infty$

row $4 \to \infty$ column $3 \to \infty$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & \infty & \infty & 0 \\ 0 & 3 & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ 11 & 0 & \infty & \infty & \infty \end{bmatrix}$$

cost of node 7 = cost of node 4 + cost (4,3) + reduction cost $= 25 + 12 + 0 = \underline{37}$

---

node 8: path from 4 to 5, so $(5,4) \to \infty$

row $5 \to \infty$, column $4 \to \infty$

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ 12 & \infty & 11 & \infty & \infty \\ 0 & 3 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 11 & 0 & 0 & \infty & \infty \end{bmatrix}$$

cost of node 8 = cost of node 4 + cost (5,4) + reduction $= 25 + 0 + 11 = \underline{36}$

---

node 9

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty & \infty \\ 11 & \infty & \infty & \infty & \infty \end{bmatrix}$$

cost of node 9 = cost of node 6 + cost (2,3) + reduction

$28 + 11 + 2 + 11$

$= \underline{52}$

node 10

$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 0 & \infty & \infty \end{bmatrix}$$

$28 + 0 + 0$

$= \underline{28}$

node 11

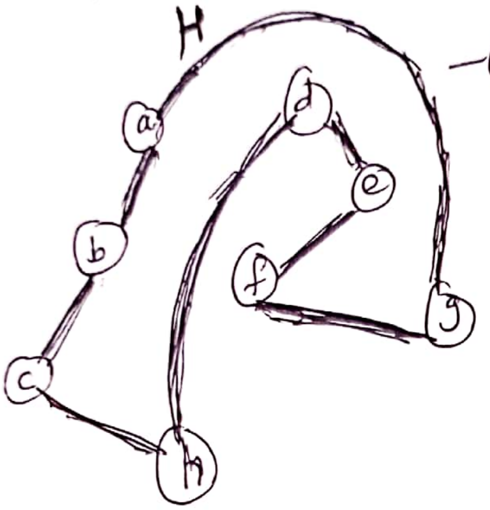
$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

$28 + 0$

$= \underline{28}$

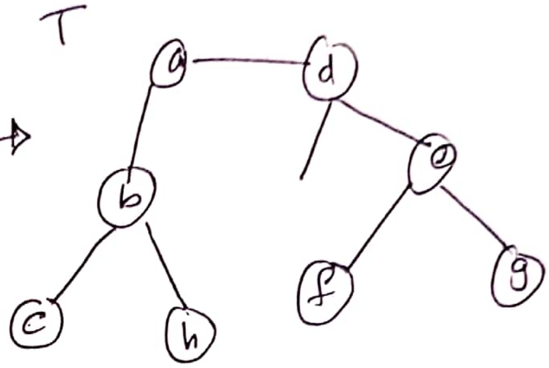
Approx. algo.

TSP- MST. technique



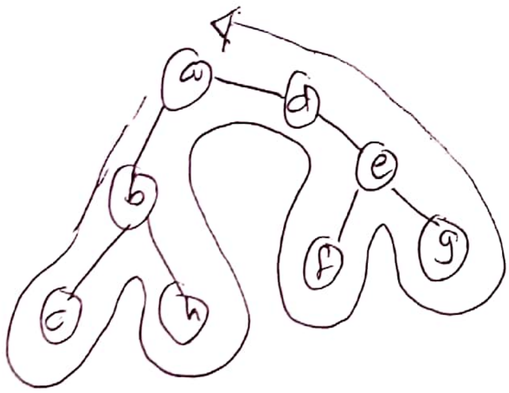
(1) Say use MST (Heuristic)

T



(2) Find ordered list of
W ← vertices in preorder walk of T.

H ← cycle that visits



Hamiltonian Cycle

H: {a b c h d e f g a}

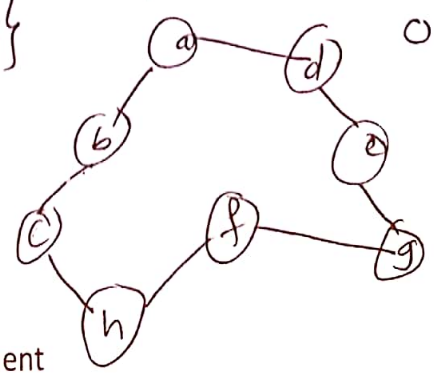
$d = \sqrt{(x_1-x_2)^2+(y_1-y_2)}$

by euclidian distances,

H cost = ~~14.715~~ 19.074

W: ~~{abcbhbadef~~
{a b c b h a d e f e g e d a}

optimal Tour by triangle inequality



Optimal Tour cost = 14.715

~~Q. Table~~ Claim

~~Q.~~

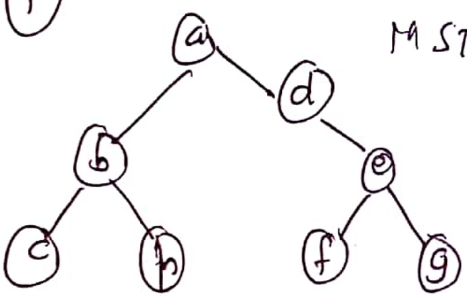
TSP soln by MST is 2-approx. algo.

soln for TSP with approx. ratio = 2.

$c(W) = 2 \cdot c(T)$. each edge visited twice.

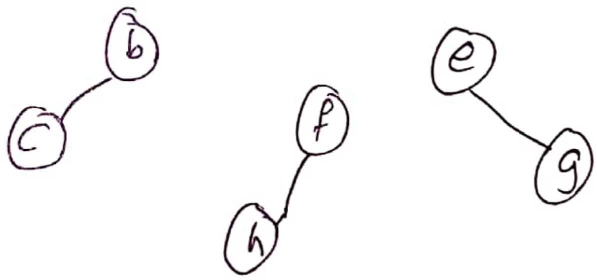
TSP - Christofides Algo.

(1) MST - T.



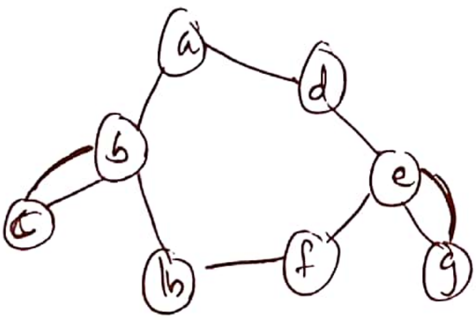
→ Find | minimum cost perfect matching M, of odd degree nodes in T

(2)



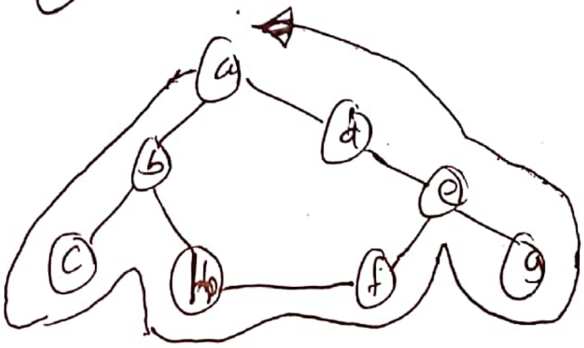
(3) Take G' union of spanning Tree T & matching edges M.

G'



travels thru every edge once & start & end at same vertex

(4) Eulerian Tour on G'



Hamiltonian visits every vertex only once & start, end at same

TSP Dynamic

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

$\rightarrow$ Tour starts & ends at vertex 1

ϕ - Vertex 1

①

C_{21} $g(2, \phi) = 5$

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

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

②

$g(2, \{3\}) = C_{23} + g(3, \phi) = 9 + 6 = 15$

$g(2, \{4\}) = C_{24} + g(4, \phi) = 10 + 8 = 18$

$g(3, \{2\}) = C_{32} + g(2, \phi) = 13 + 5 = 18$

$g(3, \{4\}) = C_{34} + g(4, \phi) = 12 + 8 = 20$

$g(4, \{2\}) = C_{42} + g(2, \phi) = 8 + 5 = 13$

$g(4, \{3\}) = C_{43} + g(3, \phi) = 9 + 6 = 15$

③ $g(2, \{3,4\}) = \min\{C_{23} + g(3, \{4\}), C_{24} + g(4, \{3\})\}$

$= \min\{9 + 20, 10 + 15\} = 25$

$g(3, \{2,4\}) = \min\{C_{32} + g(2, \{4\}), C_{34} + g(4, \{2\})\}$

$= \min\{13 + 18, 12 + 13\} = 25$

$g(4, \{2,3\}) = \min\{C_{42}, g(2, \{3\}), C_{43}, g(3, \{2\})\}$

$= \min\{8 + 15, 9 + 13\} = 23$

$$g(1, \{2,3,4\}) = \min \{ c_{12} + g(2, \{3,4\}), \; c_{13} + g(3, \{2,4\}), \; c_{14} + g(4, \{2,3\}) \}$$

$$= m\{10+25, 15+25, 20+23\}$$

$$= m\{35, 40, 43\}$$

$$= \underline{35}$$

Path
$1 \to 2 \to 4 \to 3 \to 1$
Cost 35

Complexity analysis of TSP.

TSP- brute force - $n!$

- n choices -

dynamic

n - choices

-(n-1)

- n-2

- ..... 1 choices.

$\therefore (n-1)(n-2)(n-3) \ldots 3 \times 2 \times 1 = (n-1)!$

$\therefore$ dynamic $O(n^3 \cdot 2^n)$

$$C\begin{bmatrix} 0 & 2 & 9 & 10 \\ 1 & 0 & 6 & 4 \\ 15 & 7 & 0 & 8 \\ 6 & 3 & 12 & 0 \end{bmatrix}$$

Tour - starts at 1 & end at 1

cost of travel from city 1 to (2, 3, 4).

Source 1 vertex:

$g(2, \phi) = C_{21} = 1$

$g(3, \phi) = C_{31} = 15$

$g(4, \phi) = C_{41} = 6$

K=1, consider sets of 1 element ← cost of min. distance by visiting 1 of city as intermedia

Set {2} : $g(3, \{2\}) = C_{32} + g(2, \phi) = 7+1 = 8$; $p(3,\{2\}) = 2$

path thru 2

$g(4, \{2\}) = C_{42} + g(2, \phi) = 3+1 = 4$; $p(4, \{2\}) = 2$

Set {3} : $g(2, \{3\}) = C_{23} + g(3, \phi) = 6+15 = 21$; $p(2, \{3\}) = 3$

path thru 3

$g(4, \{3\}) = C_{43} + g(3, \phi) = 12+15 = 27$, $p(4, \{3\}) = 3$

Set {4} : $g(2, \{4\}) = C_{24} + g(4, \phi) = 4+6 = 10$; $p(2, \{4\}) = 4$

path thru 4

$g(3, \{4\}) = C_{34} + g(4, \phi) = 8+6 = 14$; $p(3, \{4\}) = 4$

K = 2, consider sets of 2 elements. (2 cities inbetween)

Set {2,3} : $g(4, \{2,3\}) = \min\{C_{42} + g(2, \{3\}), C_{43} + g(3, \{2\})\}$

4 - 2-3-1, 3-2

$= \min\{3+21, 12+8\} = 20$; $p(4, \{2,3\}) = 3$

Set {2,4} : $g(3, \{2,4\}) = \min\{C_{32} + g(2, \{4\}), C_{34} + g(4, \{2\})\}$

3 - 2-4-1, 4-2-1

$= \min(7+10, 8+4) = 12$; $p(3, \{2,4\}) = 4$

Set {3,4} :

$g(2, \{3,4\}) = \min\{C_{23} + g(3, \{4\}), C_{24} + g(4, \{3\})\}$

2-3-4-1
2-4-3-1

$= \min\{6+14, 4+27\} = 20$; $p(2, \{3,4\}) = 3$

Optimal Tour. TSP dynamic

$g(1, \{2,3,4\})$ 3 cities inbetween.

$$= \min \{ c_{12} + \overset{1-2-3-4-1}{g(2, \{3,4\})},$$

$$c_{13} + \overset{1-3-4-2-1}{g(3, \{2,4\})},$$

$$c_{14} + \overset{1-4-3-2-1}{g(4, \{2,3\})} \}$$

$$= \min (2+20,\ 9+12,\ 10+20\}$$

Cost of optimal Tour. $= 21$

Successor of node 1 : $p(1, \{2,3,4\}) = 3$

Successor of node 3 : $p(3, \{2,4\}) = 4$

successor of node 4 : $p(4, \{2\}) = 2$

Tour $1 \to 3 \to 4 \to 2 \to 1$

Dynamic Prg. breaks prb. into $2^n \cdot n$ subprb.s. Each sub. prb. takes n computation

↳ time complexity $O(2^n \cdot n^2)$.

TSP- Branch & Bound.

Different bounding function example.

ound.

vertex	min
1	4
2	7
3	4
4	2
5	4
und	21

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & \infty & 14 & 4 & 10 & 20 \\ 2 & 14 & \infty & 7 & 8 & 7 \\ 3 & 4 & 5 & \infty & 7 & 16 \\ 4 & 11 & 7 & 9 & \infty & 2 \\ 5 & 18 & 7 & 17 & 4 & \infty \end{array}$$

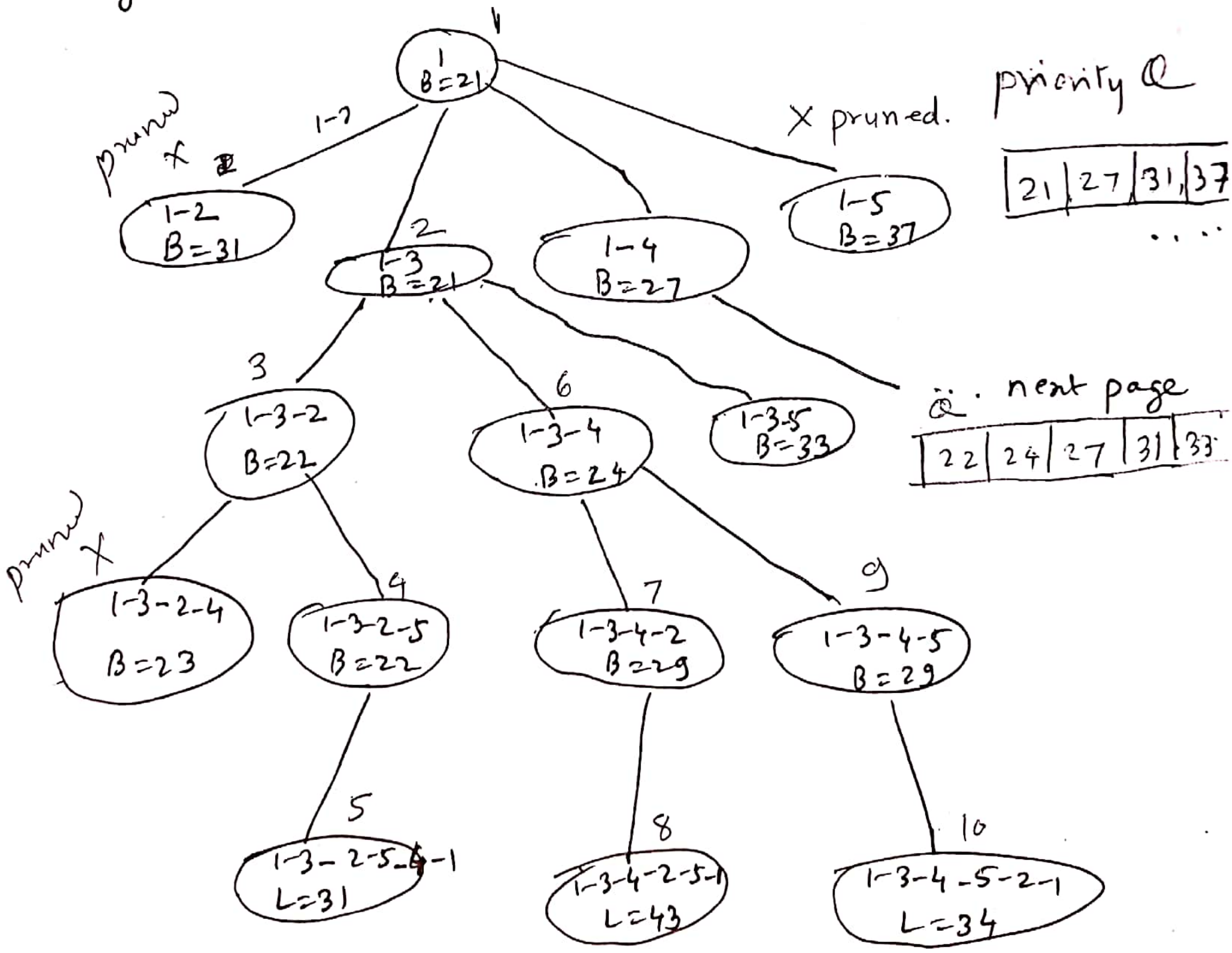
Source = 1

Define Bound as.

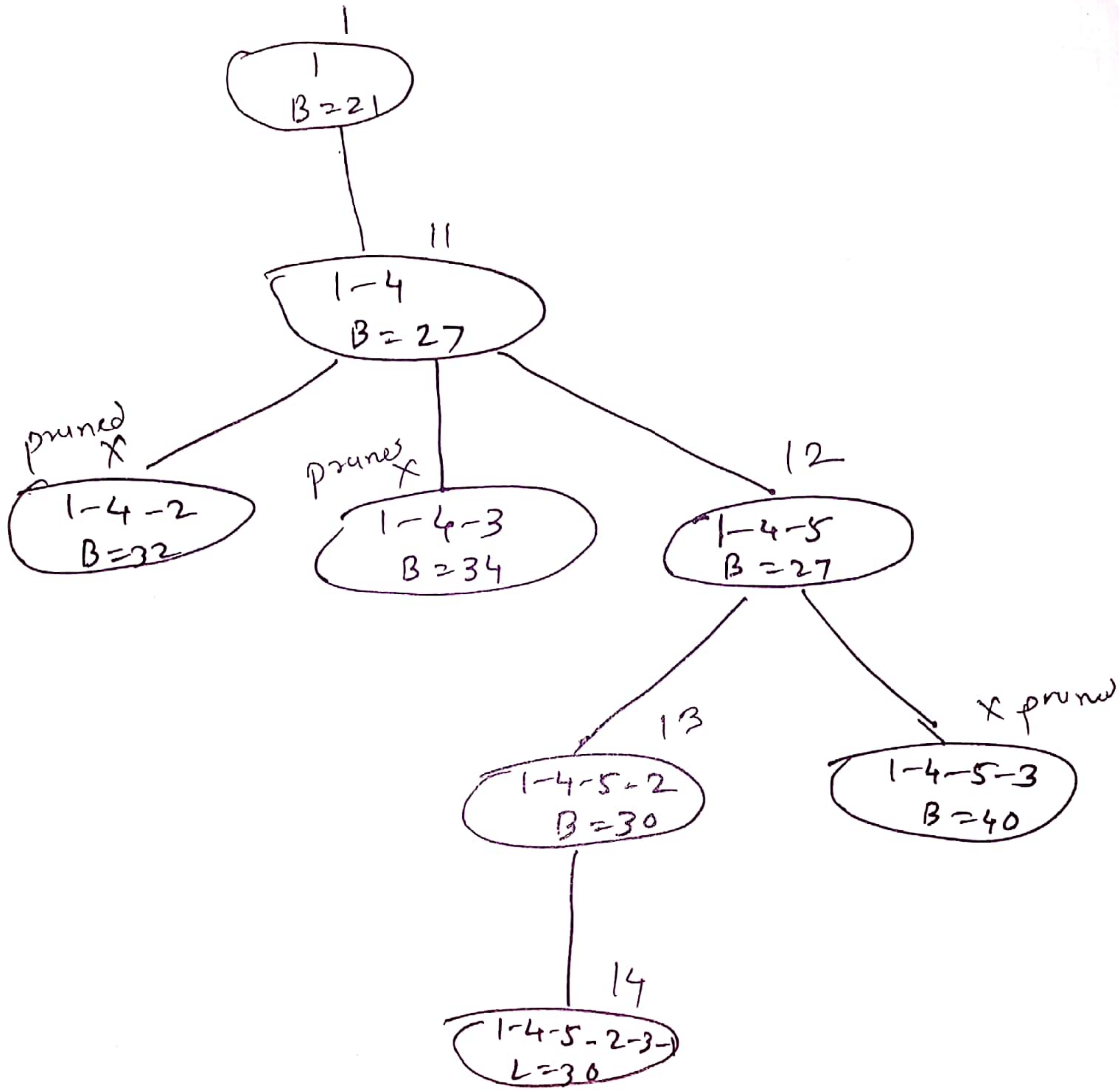
length from 1 to 2 + sum of min. outgoing edges for vertices 2 to 5

$= 14 + (7+4+2+4) = 31$

Each node gets added into priority queue, node with best bound is removed and processed. Algorithm terminates when priority queue is empty.

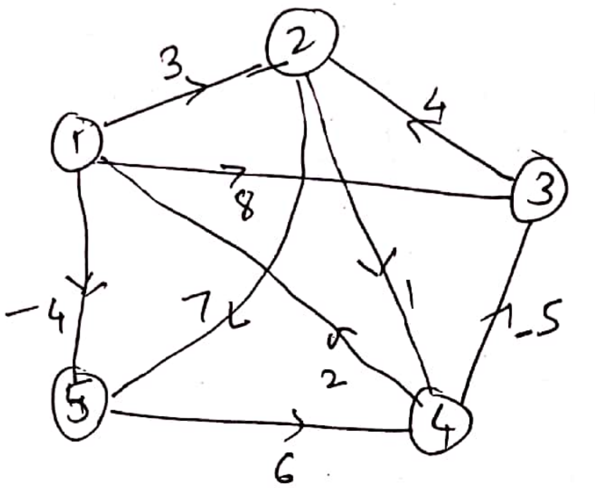


TSP-13


1
1
B=21
11
1-4
B=27
pruned X
1-4-2
B=32
prunes X
1-4-3
B=34
12
1-4-5
B=27
13
1-4-5-2
B=30
X prune
1-4-5-3
B=40
14
1-4-5-2-3-1
L=30

X

Floyd Warshall Algo.



$A[i,j] =$

$\min(A[i,j],\ A[i,k] + A[k,j])$

Initial

$$A_0 = \begin{bmatrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \infty & -5 & 0 & \infty \\ \infty & \infty & \infty & 6 & 0 \end{bmatrix} \qquad P_0 = \begin{bmatrix} - & 1 & 1 & - & 1 \\ - & - & - & 2 & 2 \\ - & 3 & - & - & - \\ 4 & - & 4 & - & - \\ - & - & - & 5 & - \end{bmatrix}$$

K. Vertex 1 inbetween.

Pass 1

$$\begin{bmatrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \underline{5} & -5 & 0 & \underline{-2} \\ \infty & \infty & \infty & 6 & 0 \end{bmatrix} \qquad P_1 = \begin{bmatrix} - & 1 & 1 & - & 1 \\ - & - & - & 2 & 2 \\ - & 3 & - & - & - \\ 4 & \underline{1} & 4 & - & \underline{1} \\ - & - & - & 5 & - \end{bmatrix}$$

Pass 2 Vertex 2 inbetween.

$$A_2 \begin{bmatrix} 0 & 3 & 8 & \underline{4} & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \underline{5} & \underline{11} \\ 2 & 5 & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{bmatrix} \qquad P_2 \begin{bmatrix} - & 1 & 1 & \underline{2} & 1 \\ - & - & - & 2 & 2 \\ - & 3 & - & 2 & 2 \\ 4 & 1 & 4 & - & 1 \\ - & - & - & 5 & - \end{bmatrix}$$

Pass 3 Vertex 3 inbetween.

$$A_3 \begin{bmatrix} 0 & 3 & 8 & 4 & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & 5 & 11 \\ 2 & \underline{-1} & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{bmatrix} \qquad P_3 \begin{bmatrix} - & 1 & 1 & 2 & 1 \\ - & - & - & 2 & 2 \\ - & 3 & - & 2 & 2 \\ 4 & \underline{3} & 4 & - & 1 \\ - & - & - & 5 & - \end{bmatrix} \quad 4332$$

Pass 4 Vertex 4 inbetween.

5-1
5-4-1
5-2
5-4-2
5-3

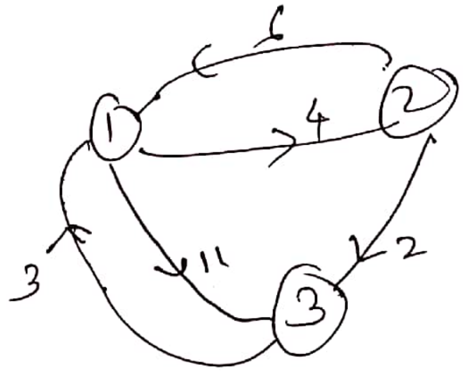
$$A_4 \begin{bmatrix} 0 & 3 & \underline{-1} & 4 & -4 \\ \underline{3} & 0 & \underline{-4} & 1 & \underline{\underline{-1}} \\ \underline{7} & 4 & 0 & 5 & \underline{\underline{3}} \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{bmatrix}$$

1-4-4-3
2-4-4-1
2-4-4-3
2-5-2-4-1-5
3-4-4-1 → 3-2-4-1
~~3-4-1-5~~
3-5 → 3-~~2-1-5~~
~~2-1-5~~
3-4-4-5

$$P_4 \begin{bmatrix} - & 1 & \underline{4} & 2 & 1 \\ \underline{4} & - & \underline{4} & 2 & \underline{\underline{1}} \\ \underline{4} & 3 & 4 & - & \underline{\underline{1}} \\ 4 & 3 & 4 & - & 1 \\ 4 & 3 & 4 & 5 & - \end{bmatrix}$$

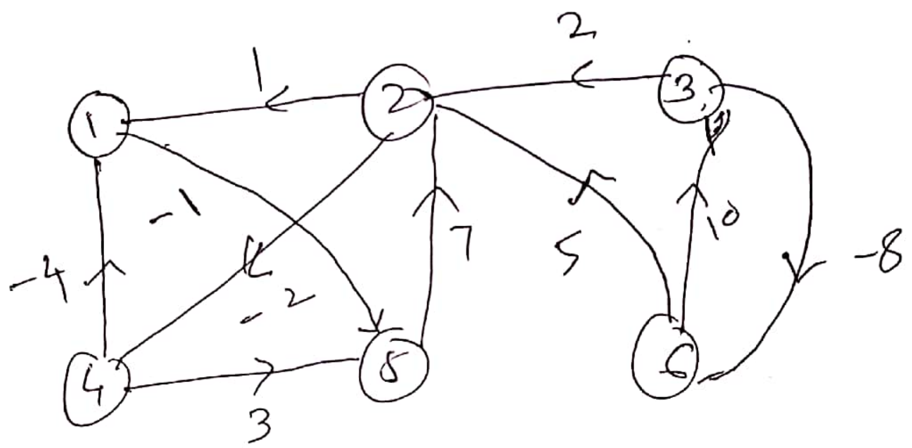
Solve by Floyd Warshall's Algo.

①



$$A_3 \begin{bmatrix} 0 & 4 & 6 \\ 5 & 0 & 2 \\ 3 & 7 & 0 \end{bmatrix} \qquad P_3 \begin{bmatrix} - & 1 & 2 \\ 3 & - & 2 \\ 3 & 1 & - \end{bmatrix}$$

②



$$A_6 = \begin{bmatrix} 0 & 6 & \infty & 4 & -1 & \infty \\ -6 & 0 & \infty & -2 & 0 & \infty \\ -4 & -3 & 0 & -5 & -3 & -8 \\ -4 & 10 & \infty & 0 & -5 & \infty \\ 1 & 7 & \infty & 5 & 0 & \infty \\ -1 & 5 & 10 & 3 & 5 & 0 \end{bmatrix} \quad P_6 = \begin{bmatrix} - & 5 & - & 2 & 1 & - \\ 4 & - & - & 2 & 1 & - \\ 2 & 6 & - & 2 & 1 & 3 \\ 4 & 5 & - & - & 1 & - \\ 4 & 5 & - & 2 & - & - \\ 4 & 6 & 6 & 2 & 1 & - \end{bmatrix}$$

3

Pass 5 Vertex 5 inbetween

$$A_5 = \begin{bmatrix} 0 & 1 & -3 & 2 & -4 \\ 3 & 0 & -4 & 1 & -1 \\ 7 & 4 & 0 & 5 & 3 \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{bmatrix} \quad P_5 = \begin{bmatrix} - & 3 & 4 & 5 & 1 \\ 4 & - & 4 & 2 & 1 \\ 4 & 3 & - & 2 & 1 \\ 4 & 3 & 4 & - & 1 \\ 4 & 3 & 4 & 5 & - \end{bmatrix}$$

① 1-5-5-2 — 1-5-5-3 1-5-5-4

A 1 5 3- 1-5-4-3 1-5-4

1-5 3 2

Path 1 - 2

1 - 3 - 2

1 - 4 - 3 - 2

1 - 5 - 4 - 3 - 2

A

1
1-1
2
4-2
3