TSP - branch & bound.

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

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

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

TSP for 5 cities by B & B method.

row reduction

∞	10	20	0	1
13	∞	14	2	0
[illegible]	3	∞	0	2
16	3	15	∞	0
12	0	3	12	∞

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,

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

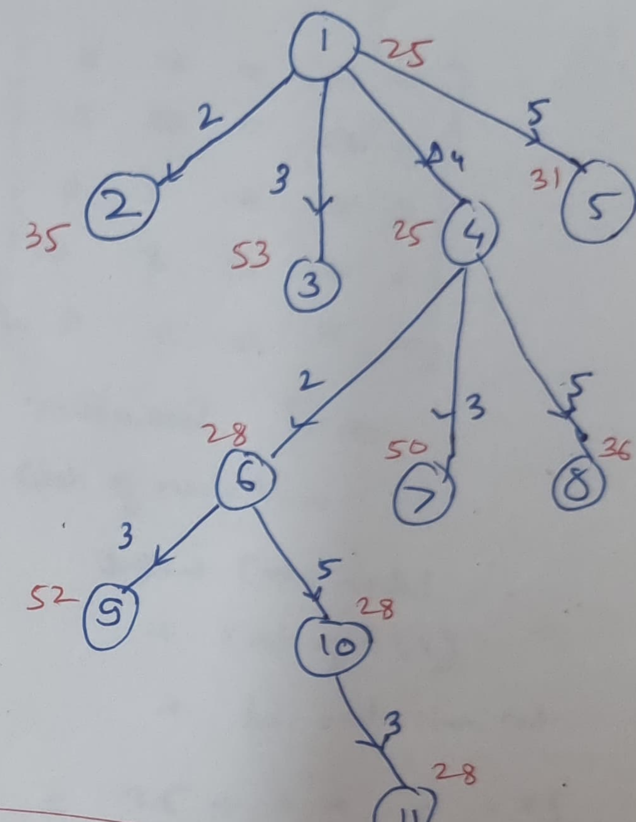
Total count by which the matrix is reduced is

$\underline{L = 25}$

node 2 (35)

as path is from 1 to 2, make row 1 to ∞. and edges incoming to 1 as ∞. , set (2,1) to ∞

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



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

Cost 10+6+2+7+3 = 28.

Cost node2 = ~~25~~ cost of node 1 + Cost of (1,2) edge + LC reduction cost)

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
$= 25 + 3 + 0 = 28$

---

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
$= 25 + 12 + 0 = 37$

---

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
$= 25 + 0 + 11 = 36$

---

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
$28 + 11 + 2 + 11$
$= 52$

node 10

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

$28 + 0 + 0$

$= 28$

node 11

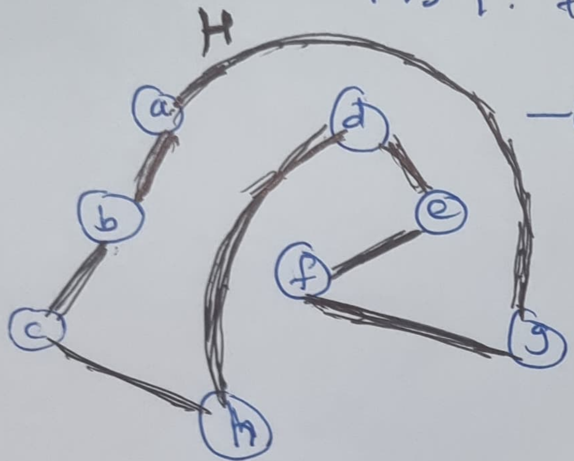
$$\begin{bmatrix} \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ 0 & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty \end{bmatrix}$$

$28 + 0$

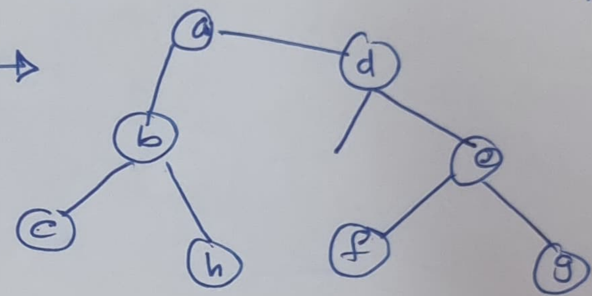
$= 28$

Approx. algo.

TSP- MST. technique

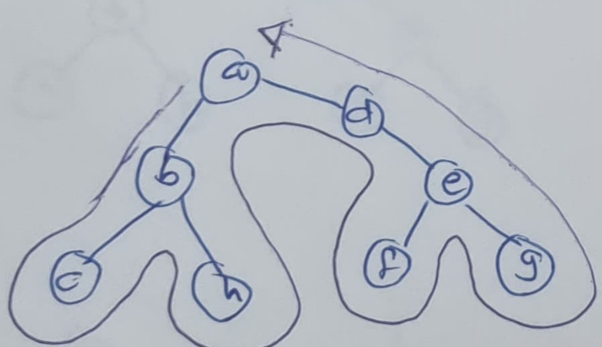


-(1) Say use MST (Heuristic)
T



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

H ← cycle that visits



Hamiltonian Cycle

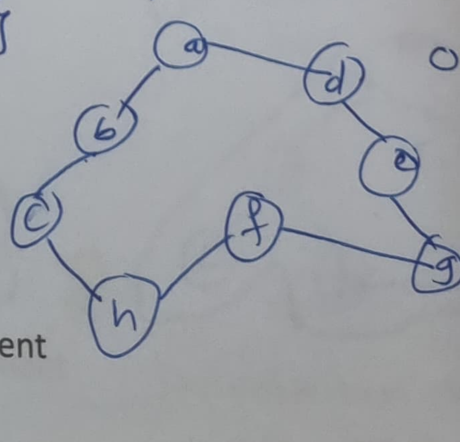
H: {a b c h d e f g a}

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

H cost = ~~14.715~~ 19.074

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

Optimal Tour by triangle inequality



Optimal Tour cost = 14.715

~~2. Table~~ Claim

~~Q.~~

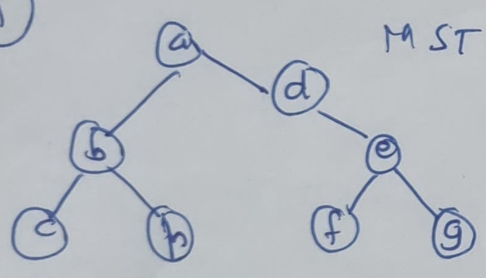
TSP soln by MST is 2-approx. algo.

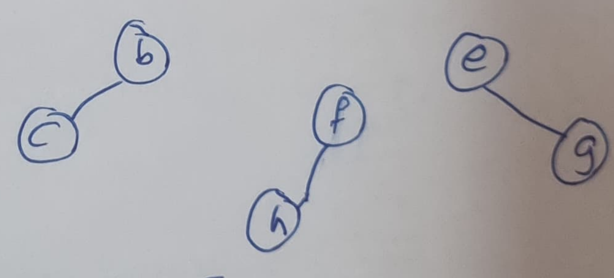
soln for TSP with approx. ratio = 2.

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

TSP- christofides Algo.

(1) MST-T. → Find (2) Minimum cost perfect-matching M, of odd degree nodes in T



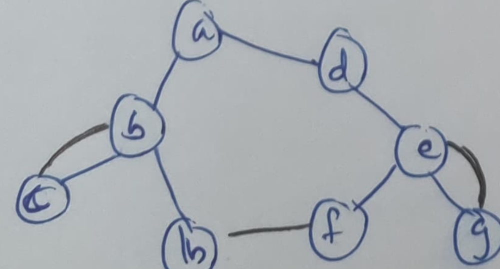


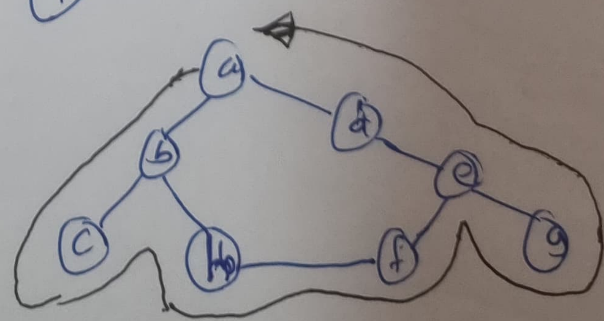
(3) Take G' union of spanning Tree T & matching edges M.

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

(4) Eulerian Tour on G'

G'





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

→ Union of MST & matching edges is eulerian.

→ every node has even degree.

This is 1.5 approx algo for TSP.

TSP-8

TSP Dynamic

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

→ 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)

$g(2, \{3,4\}) = \min\{c_{23} + g(3, \{4\}), c_{24} + g(4, \{3\})\}$
$= m\{9 + 20, 10 + 15\} = 25$

$g(3, \{2,4\}) = \min\{c_{32} + g(2, \{4\}), c_{34} + g(4, \{2\})\}$
$= m\{13 + 18, 12 + 13\} = 25$

$g(4, \{2,3\}) = \min\{c_{42}, g(2, \{3\}), c_{43}, g(3, \{2\})\}$
$= m\{8 + 15, 9 + 13\} = 23$

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

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

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

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

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

$= \underline{35}$

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

Complexity analysis of TSP.

TSP - brute force - $n!$

- n choices -

dynamic

n - choices

$-(n-1)$

$- n-2$

$- \ldots\ldots 1$ choices.

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

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

TSP-10

PLEDGE
Patni's Learning Edge

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

Tour - starts at 1 & end at 1

Source 1 vertex: $g(2, \phi) = C_{21} = 1$ — cost of travel from city 1 to {2, 3, 4}.

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

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

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

Set {2} : path thru 2

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

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

Set {3} : path thru 3

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

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

Set {4} : path thru 4

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

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

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

Set {2,3} : 4-2-3-1, 4-3-2-1

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

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

Set {2,4} : 3-2-4-1, 3-4-2-1

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

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

Set {3,4} : 2-3-4-1, 2-4-3-1

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

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

Optimal Tour.

TSP dynamic

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

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

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

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

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

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 $1 \rightarrow 3 \rightarrow 4 \rightarrow 2 \rightarrow 1$

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.

Bound.

Vertex	min
1	4
2	7
3	4
4	2
5	4
Bound	21

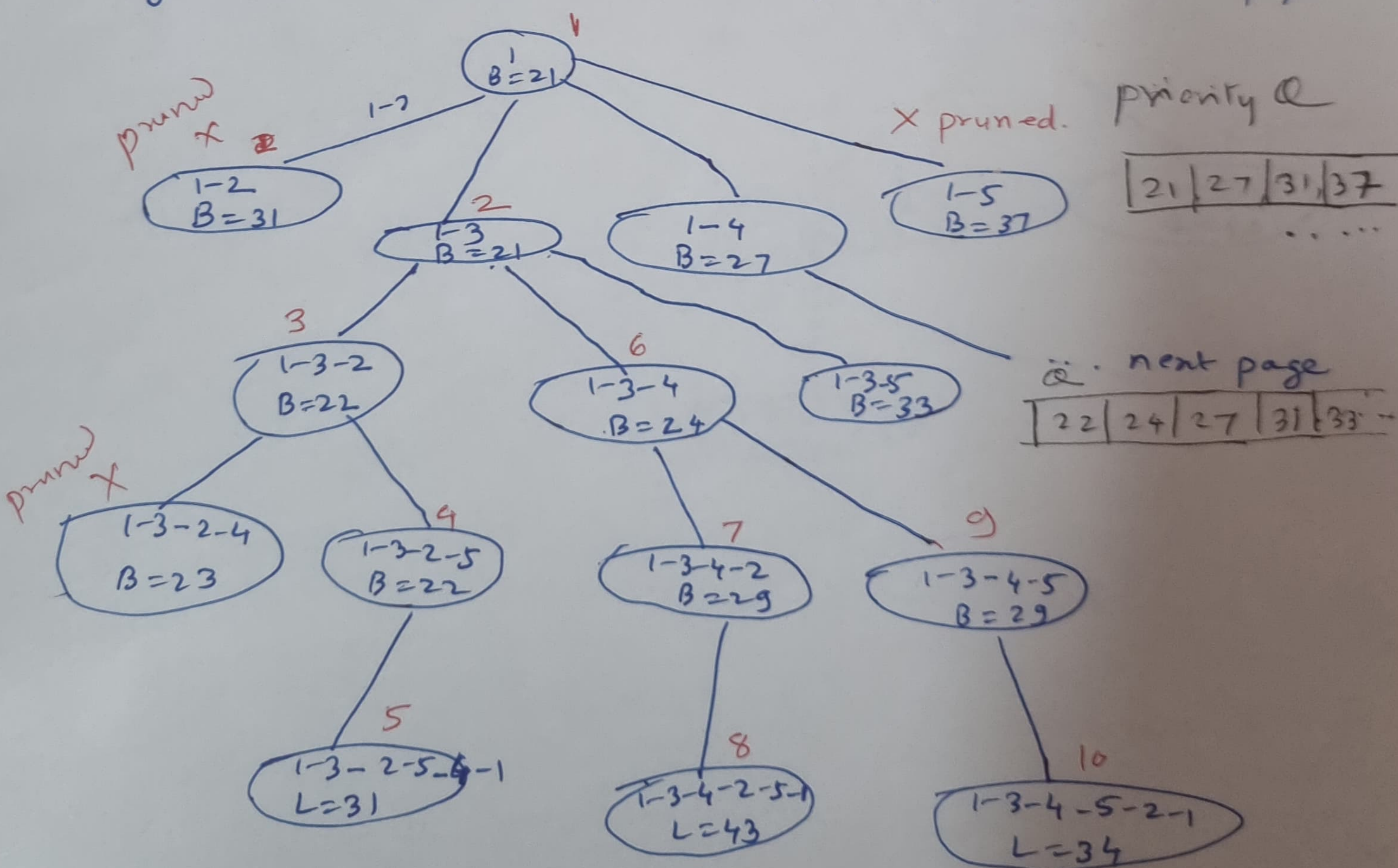
$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & \infty & 14 & 4 & 10 & 20 \\
2 & 14 & \infty & 7 & 8 & 7 \\
3 & 4 & 5 & \infty & 7 & 16 \\
4 & 11 & 7 & 9 & \infty & 2 \\
5 & 18 & 7 & 17 & 4 & \infty
\end{array}
$$

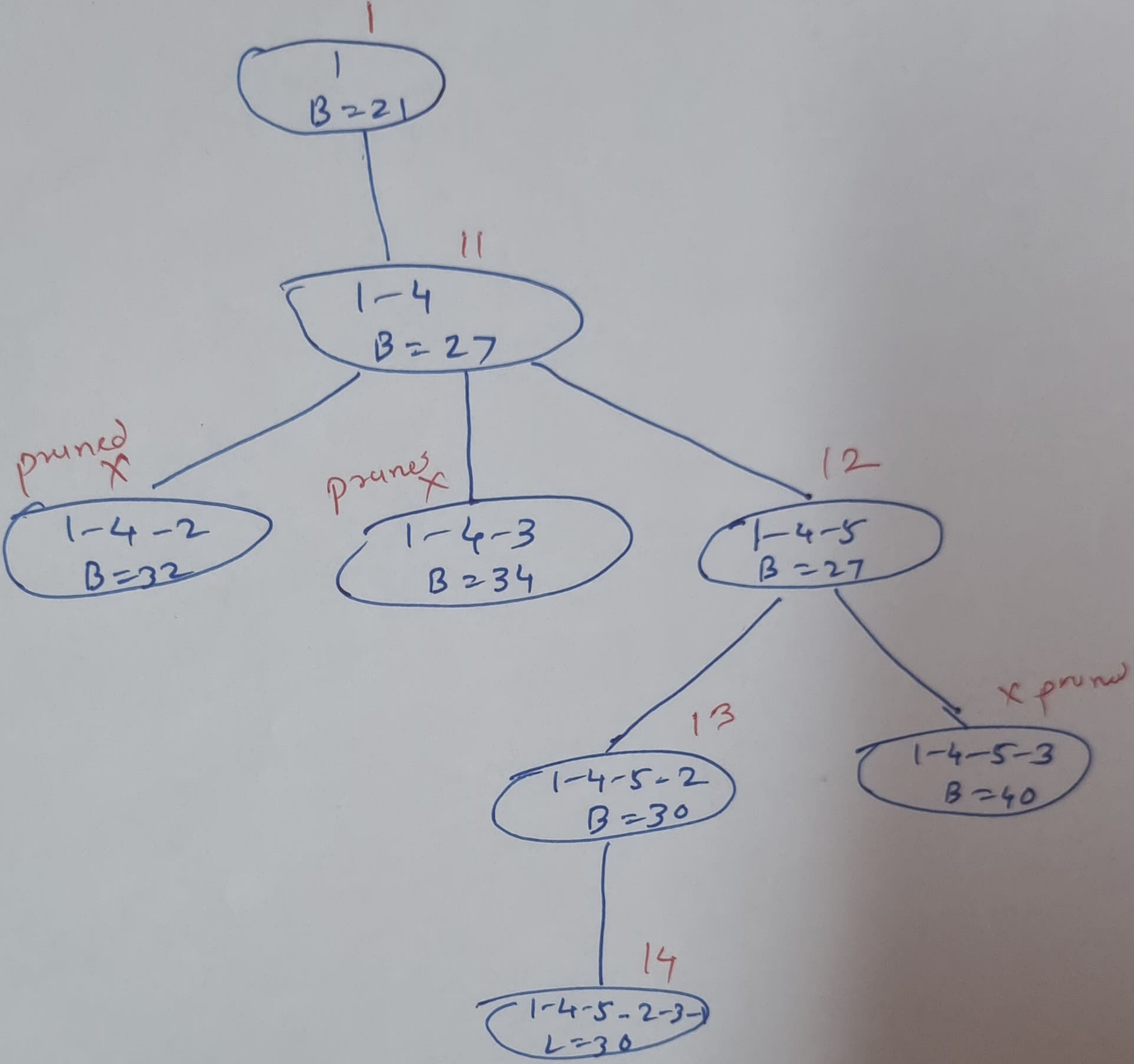
Source = 1

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

$= 14 + (7+4+2+4) = \underline{31}$

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




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