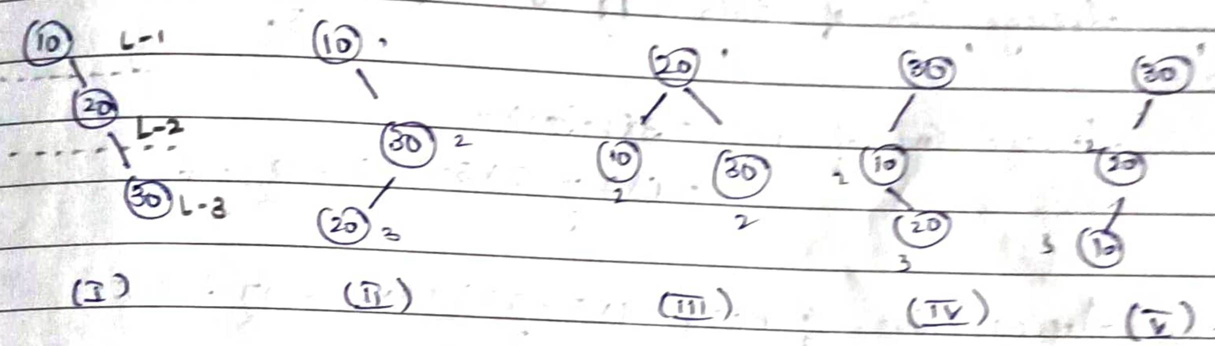
<u>Optimal Binary Search Tree</u>

From 'n' nodes (or) 'n' keys we can create $\frac{{}^{2n}C_n}{n+1}$ BST Binary search tree.

Consider 3 keys 10, 20, 30.

$n = 3$

$$\frac{{}^{2(3)}C_3}{4} = \frac{{}^{6}C_3}{4} = \frac{\frac{6!}{(3)!\,(3)!}}{4} = \frac{\frac{\cancel{6}\times5\times4\times\cancel{3!}}{\cancel{3!}\times\cancel{3!}}}{4} = \frac{20}{4} = 5$$



⇒ Cost of a binary search tree is no of Comparision required to search any element in the tree.

⇒ The elem No of search required for the element at level 1 is 1, level 2 is 2 and so on - - - - - -

⇒ Average Cost of BST(1) $= \frac{1+2+3}{3} = \frac{6}{3} = 2$

Average Cost of BST (2) $= \frac{1+2+3}{3} = \frac{6}{3} = 2$

Average Cost of BST (3) $= \frac{1+2+2}{3} = \frac{5}{3} = 1.6$

Average Cost of BTS(4) $= \frac{1+2+3}{3} = 2$

Average Cost of BST(5) $= \frac{1+2+3}{3} = 2$

→ 3rd tree is having min cost compared to other BST.

→ Along with these keys if frequency of each element is also given then cost calculation is as follows.

For each node cost will be $\sum_{i=1}^{n}(f_i \times l_i)$ i.e., (frequency × level).

ex: 10, 20, 30

freq 3, 2, 5.

Average cost of BST 1

$$\frac{3\times1+2\times2+5\times3}{3} = \frac{22}{3} = 7.3$$

Average cost of $BST_2 = \frac{19}{3} = 6.3$ $\left(\frac{1\times3+2\times5+3\times2}{3}\right)$

Average cost of $BST_3 = \cancel{3\times1+5\times2+}$

$$= \frac{1\times2+2\times3+2\times5}{3} = \frac{18}{3} = 6$$

Average cost of $BST_4 = \frac{1\times5+2\times3+3\times2}{3} = \frac{17}{3} = 5.6$

Average cost of $BST_5 = \frac{1\times5+2\times2+3\times3}{3} = \frac{18}{3} = 6$

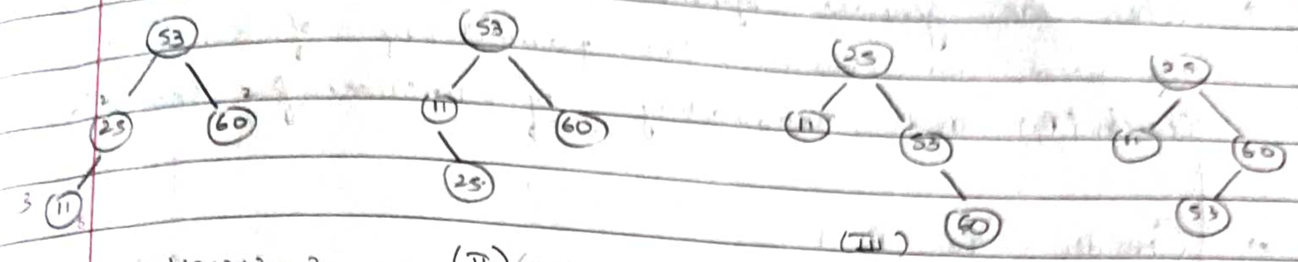
∴ 4th tree is having min cost compared to other BST

ex2: 53, 25, 60, 11. Identify all BST can be created by taking n=4.

$$\frac{{}^{2n}C_n}{n+1} = \frac{{}^{8}C_4}{5} = \frac{\frac{8\times7\times6\times5\times4!}{(4)!\,(4)!}}{5} = \frac{\frac{8\times7\times6\times5}{4\times3\times2}}{5}$$

$$= \frac{70}{5} = 14.$$

Camlin Page
Date / /

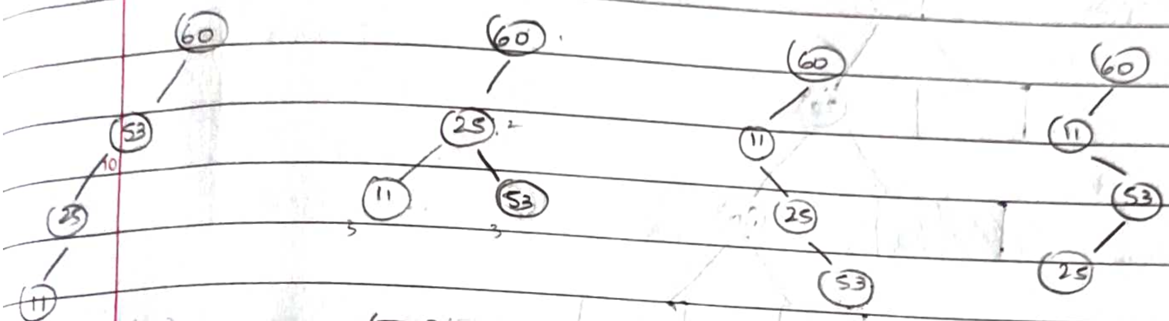


(I) $AC = \frac{1+2+2+3}{4} = 2$

(II) $AC = 2$

(III) $AC = 2$

(IV) $AC = 2$

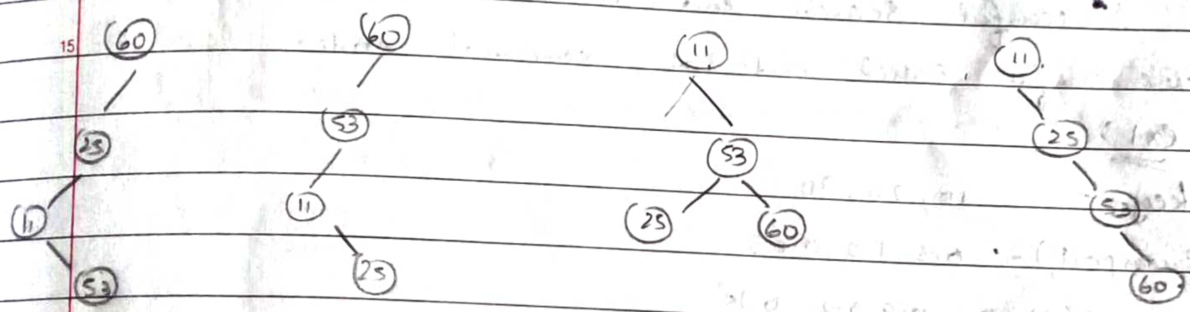


(V) $AC = \frac{10}{4} = 2.5$

(VI) $AC = \frac{9}{4} = 2.2$

(VII) $AC = 2.5$

(VIII) $AC = 2.5$

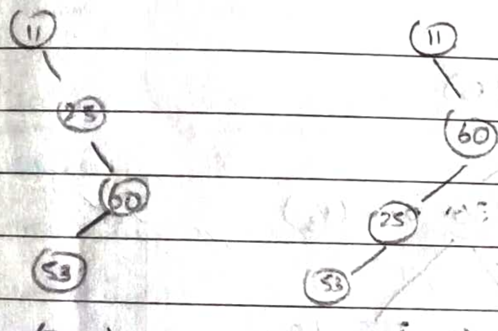


(IX) $AC = 2.5$

(X) $AC = 2.5$

(XI) $AC = 2.2$

(XII) $AC = 2.5$

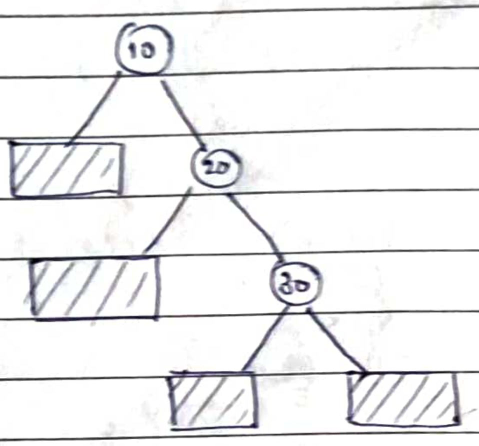


(XIII) $AC = 2.5$

(XIV) $AC = 2.5$

⇒ When we generate optimal binary search tree we need to Consider, the probability of succesful Search (P_i) and probability of unsuccessful search (q_i

⇒ Consider the BST.



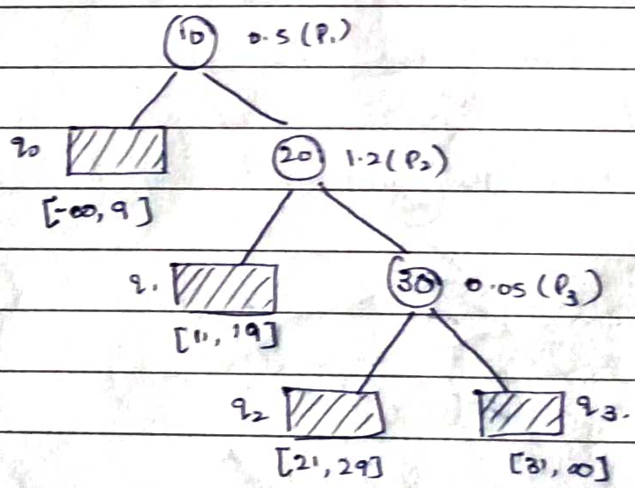
Succesful search end at internal node and unsuccessful search end at external node.

<u>ex1:</u>

key - 10, 20, 30

Succes prob(P_i) - 0.5, 1.2, 0.05

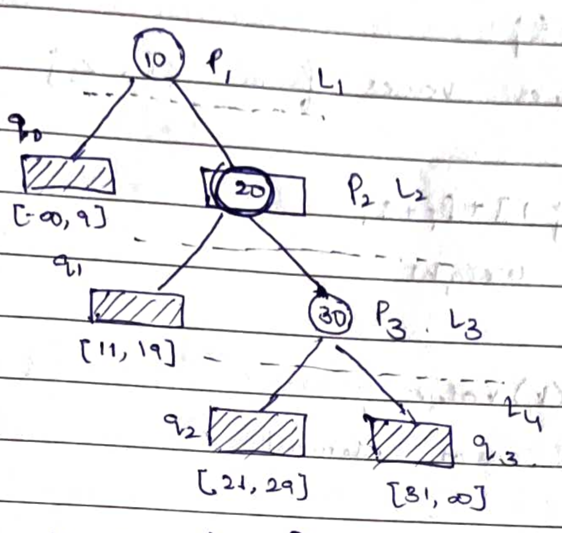
unsuc prob(q_i) - 0.5, 0.3, 0.2, 0.15.



Ex2: k - 10 20 30

p_i - 0.1 0.2 0.3

q_i - 0.1 0.05 0.15 0.05



$$Cost[0,n] = \sum_{i=1}^{n} P_i \times (level(a_i)) + \sum_{i=0}^{n} q_i (level(E_i) - 1)$$

$$Cost[0,3] = 0.1\times1 + 0.2\times2 + 0.3\times3 + 0.1\times1 + 0.05\times2 + 0.15\times3 + 0.05\times3$$
$$= 0.1 + 0.4 + 0.9 + 0.1 + 0.1 + 0.45 + 0.15$$
$$= 1.4 + 0.8$$
$$= \underline{2.2}$$

0.45
0.15
0.60
0.2

Creation/Generation of Optimal Binary Search Tree using Dynamic Programming:

Problem Statement: We need to generate a binary search tree which gives minimum cost by considering both successful and unsuccessful search probabilities.

⇒ We use tabular method for solving this problem.

Equation:

Equation:

* $Cost[i,j] = \min\limits_{i<k\leq j} \{Cost[i,k-1] + Cost[k,j]\} + w[i,j]$

where k values varies from $<i$, $\leq j$

* $w[i,j] = w[i,j-1] + p_j + q_j$

w represents weight

* $r[i,j] = \min(k)$ value

r represents root element.

ex3: k: 10, 20, 30, 40

p_i: 3, 3, 1, 1

q_i: 2, 3, 1, 1, 1

10

	0	1	2	3	4
0,0; 1,1; 2,2; 3,3; 4,4 → $j-i=0$	$w_{00}=2$ $c_{00}=0$ $r_{00}=0$	$w_{11}=3$ $c_{11}=0$ $r_{11}=0$	$w_{22}=1$ $c_{22}=0$ $r_{22}=0$	$w_{33}=1$ $c_{33}=0$ $r_{33}=0$	$w_{44}=$ $c_{44}=$ $r_{44}=$
i,j (j,i): (0,1) (1,0); (1,2) (2,1); (2,3) (3,2); (3,4) (4,3) → $j-i=1$	$w_{01}=8$ $c_{01}=8$ $r_{01}=1$	$w_{12}=7$ $c_{12}=7$ $r_{12}=2$	$w_{23}=3$ $c_{23}=3$ $r_{23}=3$	$w_{34}=3$ $c_{34}=3$ $r_{34}=4$	
(i,j): (0,2), (1,3), (2,4) → $j-i=2$	$w_{02}=12$ $c_{02}=19$ $r_{02}=1$	$w_{13}=9$ $c_{13}=12$ $r_{13}=2$	$w_{24}=5$ $c_{24}=8$ $r_{24}=3$		
(i,j): (0,3), (1,4) → $j-i=3$	$w_{03}=14$ $c_{03}=25$ $r_{03}=2$	$w_{14}=11$ $c_{14}=19$ $r_{14}=2$			
(i,j): (0,4) → $j-i=4$	$w_{04}=16$ $c_{04}=32$ $r_{04}=2$				

Camlin Page
Date / /

$c[i,i] = 0$
$w[i,i] = q_0$
$r[i,i] = 0$

1st row represent unsuccessful row.

$w[i,j] = w[i, j-1] + p_j + q_j$

→ $w[0,1] = w[0,0] + p_1 + q_1$
$= 2+3+3 = 8$

$i=0$

$j=0$

$c[0,1] = \min(c[0,0] + c[1,1]) + w[0,1]$ $k = 1$
$= 0+0+8 = 8$ $1 < k \leq j$
$r[0,1] = \min(k)$ value $0 \leq k \leq r$
$= 1$

→ $w[1,2] = w[1,1] + p_2 + q_2$ $i=1$
$= 3+3+1 = 7$ $j=2$

$c[1,2] = \min[(c[1,1] + c[2,2]) + w[1,2]$ $k=2$
$= 0+0+7 = 7$

$r[1,2] = 2.$

→ $w[2,3] = w[2,2] + p_3 + q_3$
$= 1+1+1 = 3$ $i=2$ $k=3$ $j=3$

$c[2,3] = \min[c[2,2] + c[3,3]) + w[2,3]$
$= 0+0+3$
$= 3$

$r[2,3] = 3$

→ $w[3,4] = w[3,3] + p_4 + q_4$
$= 1+1+1$
$= 3$

$c[3,4] = \min(c[3,3] + c[4,4]) + w[3,4]$
$= 0+0+3 = 3$

$r[3,4] = 4$

→ $w[0,2] = w[0,1] + P_2 + q_2$

$= 8 + 3 + 1$

$= 12$

If $k=2$: $c[0,2] = \min(c[0,1] + c[2,2]) + w[0,2]$

$= 0 + 7 + 12$

$= 19$

min is 19, so we consider $k=1$

$r[0,2] = 1$

→ $w[1,3] = w[1,2] + P_3 + q_3$

$= 7 + 1 + 1$

$= 9$

$$c[1,3] = \begin{array}{l} k=2 \\ k=3 \end{array} \min \begin{bmatrix} c[1,1] + c[2,3] \\ c[1,2] + c[3,3] \end{bmatrix} + w[1,3]$$

$= 3 + 9$

$= 12$

$r[1,3] = 2$

→ $w[2,4] = w[2,3] + P_4 + q_4$

$= 3 + 1 + 1 = 5$

$$c[2,4] = \min \begin{array}{l} k=3 \\ k=4 \end{array} \begin{bmatrix} c[2,2] + c[3,4] \\ c[2,3] + c[4,4] \end{bmatrix} + w[2,4]$$

$= 3 + 5 = 8$

$r[2,4] = 3$

→ $w[0,3] = w[0,2] + P_3 + q_3$

$= 12 + 1 + 1$

$= 14$

$$c[0,3] = \min \begin{matrix} k=1 \\ k=2 \\ k=3 \end{matrix} \begin{bmatrix} c[0,0] + c[1,3] \\ c[0,1] + c[2,3] \\ c[0,2] + c[3,3] \end{bmatrix} + w[0,3]$$

$$= 11 + 14 = 25$$

$r[0,3] = \min(k) = 2$

$\rightarrow$ $w[1,4] = w[1,3] + p_4 + q_4$

$$= 9 + 1 + 1 = 11$$

$$c[1,4] = \min \begin{matrix} k=2 \\ k=3 \\ k=4 \end{matrix} \begin{bmatrix} c[1,1] + c[2,4] \\ c[1,2] + c[3,4] \\ c[1,3] + c[4,4] \end{bmatrix} + w[1,4]$$

$$= 8 + 11 = 19$$

$r[1,4] = 2$

$\rightarrow$ $w[0,4] = w[0,3] + p_4 + q_4$

$$= 14 + 1 + 1 = 16$$

$$c[0,4] = \min \begin{matrix} k=1 \\ k=2 \\ k=3 \\ k=4 \end{matrix} \begin{bmatrix} c[0,0] + c[1,4] \\ c[0,1] + c[2,4] \\ c[0,2] + c[3,4] \\ c[0,3] + c[4,4] \end{bmatrix} + w[0,4]$$

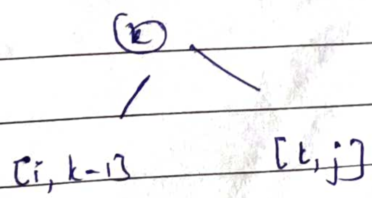
$$= 16 + 16 = 32$$

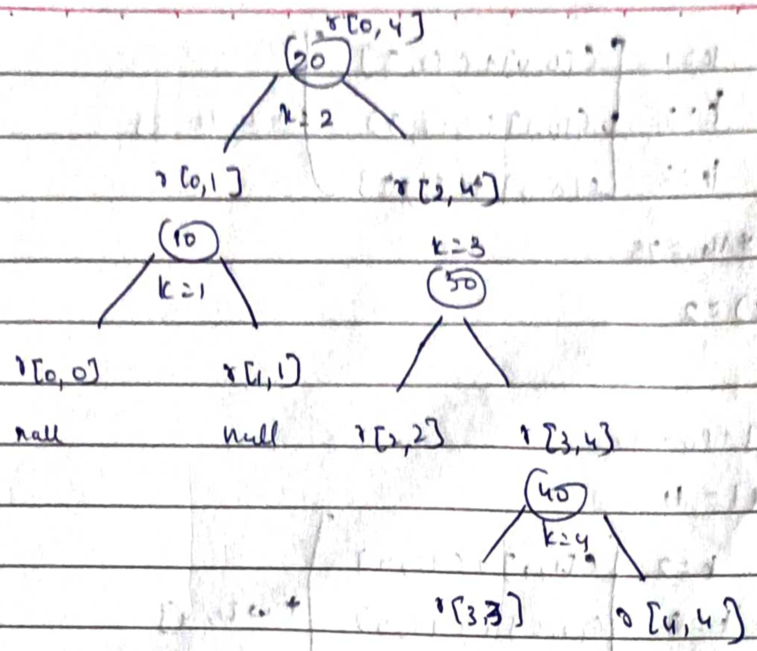
$r[0,4] = 2$

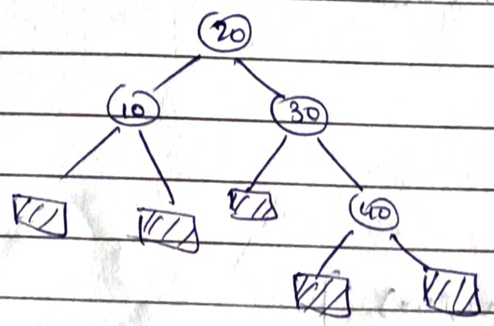
<u>Optimal</u> cost of tree $= \dfrac{32}{\Sigma(p_i + q_i)} = \dfrac{32}{8+8} = \dfrac{32}{16}$

$$= \underline{\underline{2}}$$

Optimal Bstree will be







while constructing tree wrt 'k' value [from table] we find left subtree as [i, k-1] and right subtree as [k, j]

→ In above example,
for r[0,4], we found k=2 from table
So left subtree of this as (0,1) and right subtree as (2,4)

Construction of Optimal Binary Search tree using dynamic pgming when only frequency is given.

key 10 12 16 21
4 2 6 3

C.L.	0	1	2	3	4
0	0	4^1	8^1	20^3	26^3
1		0	2^2	10^3	16^2
2			0	6^3	12^3
3				0	3^4
4					0

$C[i,j] = \min[\text{cost}[i, k-1] + \text{cost}[k, j]] + w[i,j]$

$C[0,2] = \min \begin{bmatrix} \overset{0}{C[0,0]} + \overset{2}{C[1,2]} \\ \underset{4}{C[0,1]} + \underset{0}{C[2,2]} \end{bmatrix} + 6$ $\quad j-i=2 \quad (0,2)$

$i=0 \quad (1,3)$

$j=2 \quad (2,4)$

$k = 0 < k \le 2 \qquad = 2+6$

$= 1, 2 \qquad = 8$

$C[1,3] = \min \begin{bmatrix} C[1,1] + \overset{0\ +\ 6}{C[2,3]} \\ \underset{2\ +\ 0}{C[1,2] + C[3,3]} \end{bmatrix} + 8$

$i=1$

$j=3$

$k = 1 < k \le 3 \qquad = \underline{10}$

$= 2, 3$

$C[2,4] = \min \begin{bmatrix} \overset{0\ +\ 3}{C[2,2] + C[3,4]} \\ \underset{6\ +\ 0}{C[2,3] + C[4,4]} \end{bmatrix} + 9$ $\quad w = \text{freq}[3,4]$

$i=2 \quad = 6+3 = 9$

$j=4 \quad 12^3$ since k value $= 3$.

$k = 2 < k \le 4 \qquad = \underline{12}$

$= 3, 4$

$$c[0,3] = \min \begin{bmatrix} c[0,0] + c[1,3] \\ c[0,1] + c[2,3] \\ c[0,2] + c[3,3] \end{bmatrix} + 12$$

$i = 0$

$j = 3$

$k = 1, 2, 3$

$$= \min[8+0] + 12$$

$$= \underline{20}$$

$$c[1,4] = \min \begin{bmatrix} c[1,1] + c[2,4] \\ c[1,2] + c[3,4] \\ c[1,3] + c[4,4] \end{bmatrix} + 11$$

$i = 1$

$j = 4$

$k = 2, 3, 4.$

$$= \min[2+3] + 11$$

$$= 5 + 11$$

$$= \underline{16}$$

$$c[0,4] = \min \begin{bmatrix} c[0,0] + c[1,4] \\ c[0,1] + c[2,4] \\ c[0,2] + c[3,4] \\ c[0,3] + c[4,4] \\ c[0,4] \end{bmatrix} + 15.$$

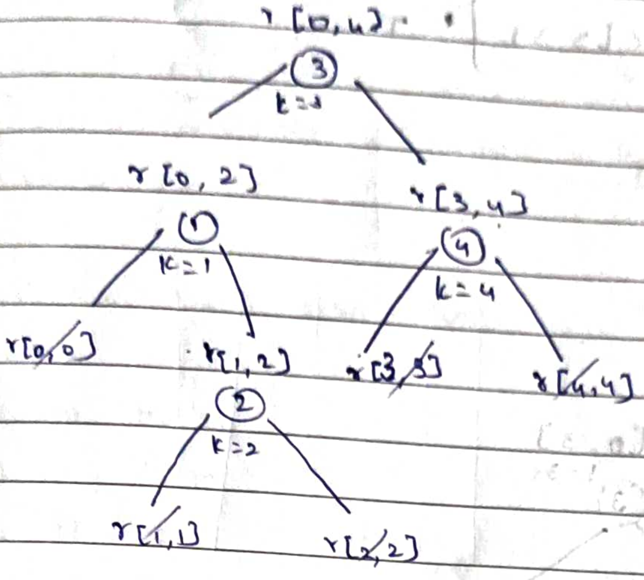
$i = 0$

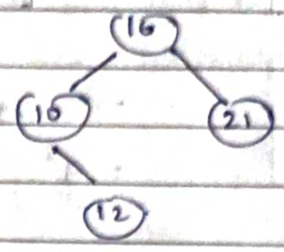
$j = 4$

$k = 1, 2, 3, 4.$

$$= \min$$

$$= \underline{26}$$





2) key 10 20 30.

3 2 5.

	0	1	2	3
0	0	3^1	7^1	17^3
1		0	2^2	9^3
2			0	5^3
3				0

$c[i,j] = \min[\text{Cost}[i,k-1] + \text{Cost}[k,j]] + w[i,j]$

$c[0,2] = \min \begin{bmatrix} \overset{0}{c[0,0]} + \overset{2}{c[1,2]} \\ \overset{3}{c[0,1]} + \overset{0}{c[2,2]} \end{bmatrix} + 5$

$i=0$

$j=2$

$k=1,2. \quad = \min[0+2]+5$

$= 7$

$c[0,3] = \min \begin{bmatrix} \overset{0}{c[0,0]} + \overset{9}{c[1,3]} \\ \overset{3}{c[0,1]} + \overset{5}{c[2,3]} \\ \underset{7}{c[0,2]} + \underset{0}{c[3,3]} \end{bmatrix} + 10.$

$i=0$

$j=3$

$k=1,2,3. = \min[7+0]+10$

$= 17$

$c[i, k-1] + c[k, j]$

$$c[1,3] = \min \begin{bmatrix} c[1,1] + c[2,3] \\ c[1,2] + c[3,3] \end{bmatrix} + 7$$

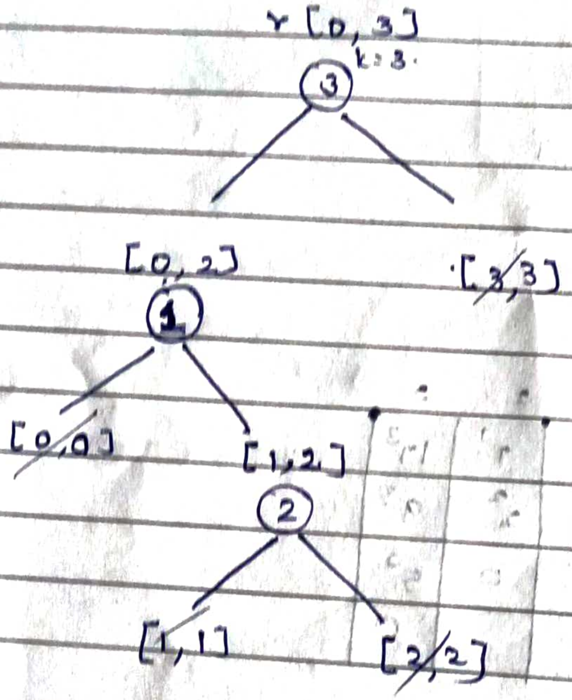
$i=1$

$j=3$

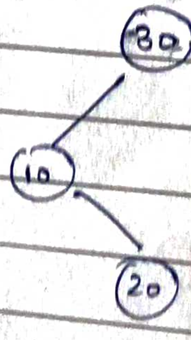
$k=2,3$

$= 2 + 7$

$= \underline{9}$



1	2	3
10	20	30

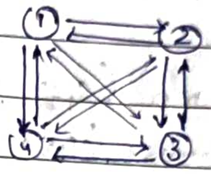


Travelling Sales person problem:

Problem Statement: We start from 1 vertex and find out the path which cover all the vertices (exactly once) and returning back to starting vertex

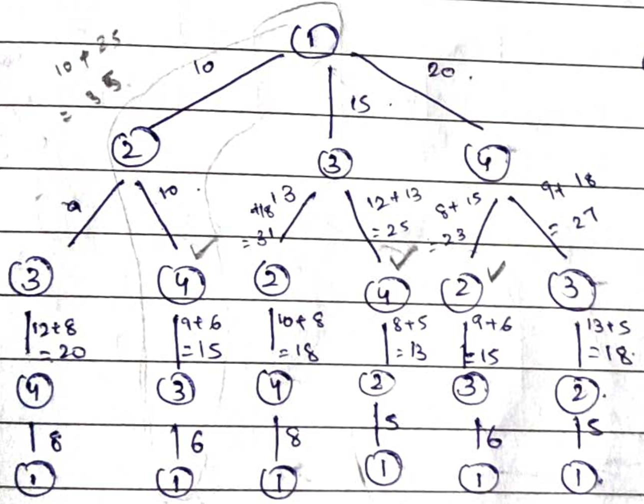
Objective: We should achieve this path with min. cost

	1	2	3	4
1	0	10	15	20
2.	5	0	9	10
3	6	13	0	12
4	8	8	9	0



The different possibilities of covering all other vertices & reaching back to 1 is as follows.

Brute force method:
Calculating all possibilities



Cost of moving from 3 to 4 is 12.
Travelling from 3 to 1 via 4 is 12+8 = 20.

Cost of 2 to 1 (path 1) min (9+20) = 29
(path 2) min (10+15) = 25.

∴ Cost of 2 to 1 = 25.

we can take any vertex as starting vertex.

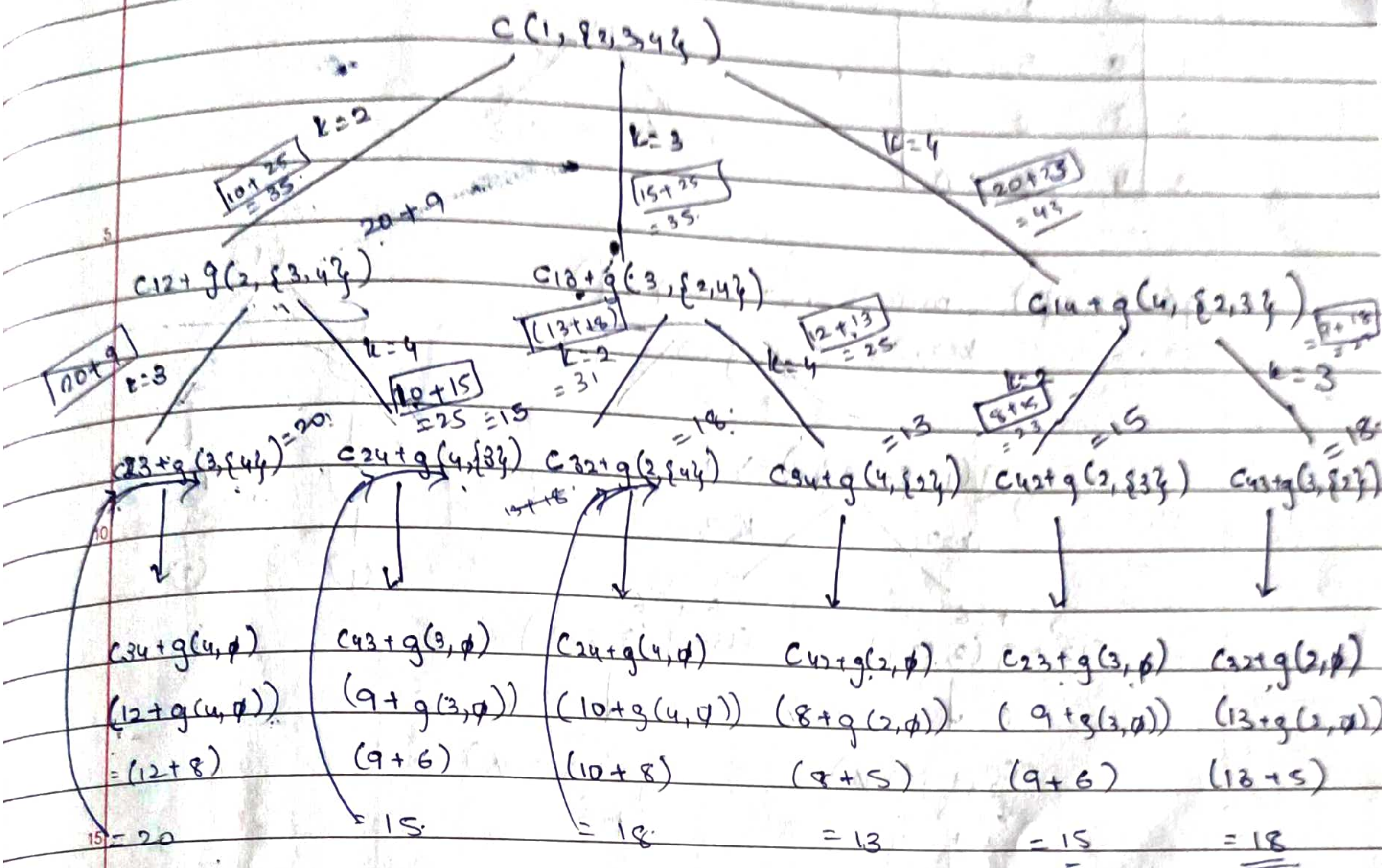
→ Solving above pblm using dynamic prog. approach.

$$\boxed{g(i, \{S\}) = \min_{\substack{k = 2 \text{ to } \le n \\ \text{(or)} \\ k \in S}} \left(c_{ik} + g(k, \{S\} - k)\right)}$$ cost to 1

In the above ex. if we consider starting vertex = 1, $i = 1$, $S = \{2, 3, 4\}$ (the remaining vertices to be visited)

$$g(1, \{2,3,4\}) = \min_{\substack{\cancel{k=2 \text{ to}} \\ k = \{2,3,4\}}} \begin{cases} (c_{12} + g(2, \{3,4\}), \\ (c_{13} + g(3, \{2,4\}), \\ (c_{14} + g(4, \{2,3\}). \end{cases}$$

Using the above eq^n, the recursive tree to trace the path is as shown below.



$g(2,\phi)$ - 5

$g(3,\phi)$ - 6

$g(4,\phi)$ - 8

$g(2,\{3,4\})$ - ~~29~~ minvalue = 25

$g(3,\{2,4\})$ - ~~31~~ 25

$g(4,\{2,3\})$ - ~~33~~ 23

$g(2,\{3\})$ - 15

$g(2,\{4\})$ - 18

$g(3,\{2\})$ - 18

$g(3,\{4\})$ - 20

$g(4,\{2\})$ - 13

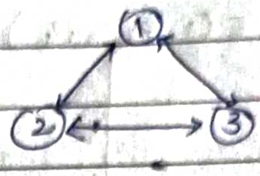
$g(4,\{3\})$ - 15

$g(1,\{2,3,4\}) \rightarrow 35$

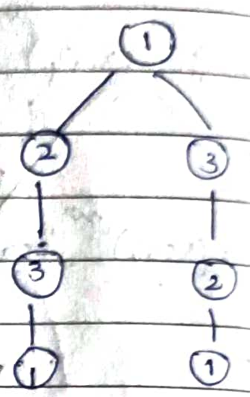
Min. Cost = 35

$1 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 1$ is the path we have choosen

$$\begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 5 & 6 \\ 2 & 3 & 0 & 2 \\ 3 & 4 & 3 & 0 \end{array}$$



$g(1, \{2,3\}) = \min_{k=2,3} \{ c_{12} + g(2, \{3\}),\ c_{13} + g(3, \{2\}) \}$



$g(1, \{2, 3\}) \rightarrow 11$

k=2: $\overset{5}{c_{12}} + \overset{6}{g(2, \{3\})} \downarrow c_{23} + g(3, \phi) \rightarrow 2 + 4$

k=3: $\overset{6}{c_{13}} + \overset{6}{g(3, \{2\})} \downarrow c_{32} + g(2, \phi) \rightarrow 3 + 3$

$g(2, \phi) - 3$
$g(3, \phi) - 4$
$g(2, \{3\}) - 6$
$g(3, \{2\}) - 6$
$g(1, \{2, 3\}) - 11$

→ min. Cost value = 11
1→2→3→1 is the path choosen.

Rod-Cutting problem:-

Problem Statement:

→ Given a rod of length 'n'
→ Array of pieces (which include length, profit of each piece
→ Problem is to find max profit obtainable by cutting rod and selling the pieces.

Eqⁿ → profit max is

$Profit_{max}$ = max. [(profit excluding new piece, profit including new piece)]

L=5

piece len	1	2	3	4
profit	2	5	9	6

→ Total rod length

len of piece ↓	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	2	4	6	8	10
2	0	2	5	7	10	12
3	0	2	5	9	11	14
4	0	2	5	9	11	14

$x_1 = 0$ (5-5=0) (0 got from x_0 so $x_1 = 0$.)

$x_2 = 1$ (14-9 = 5, check for 5) (got it self so 1)

$x_3 = 1$ (∵ not got from x_2)

$x_4 = 0$. (∵ we got it from x_3)

14 is max profit.

We will get max profit by including the piece len 2 and 3

2+3

= 5+9

= 14

piece len 1 2 3 4 5

L=7

profit 1 5 7 5 2

→ rod length

len of piece ↓	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	1	5	6	10	11	15	16
3	0	1	5	7	10	12	15	17
4	0	1	5	7	10	12	15	17
5	0	1	5	7	10	12	15	17

(11-7)=4

$x_1 = 1$

(16-5)= 11, 10

$x_2 = \phi$

$x_3 = 0$

$x_4 = 0$

$x_5 = 0$

3+2+2 is max profit.

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	1	5	6	10	11	15	16
3	0	1	5	7	10	12	15	17
4	0	1	5	7	10	12	15	17
5	0	1	5	7	10	12	15	17

(11-7)

$x_1 = 4 = 0$

(16-5)=11

$x_2 = 1$

$x_3 = \phi$

$x_4 = 0$

$x_5 = 0$

5+5+7

$\boxed{2+2+3}$