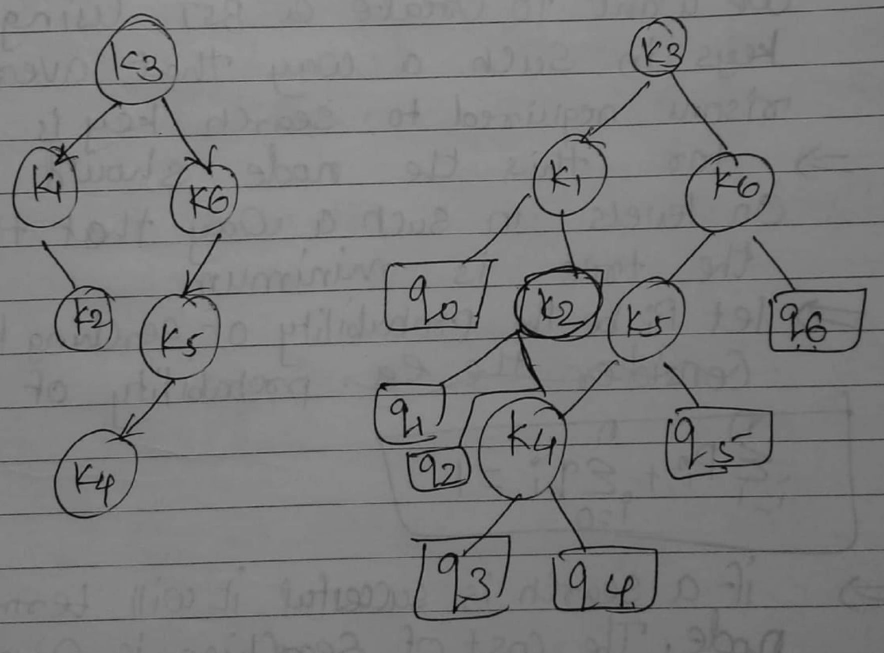
Optimal Binary search trees

⇒ An optimal binary search tree is a binary search tree for which the nodes are arranged on levels such that the tree cost is minimum

⇒ for better representation purpose of OBST we will consider "Extended binary search tree" which have the keys stored at their internal nodes. suppose n keys k1, k2, kn are stored at the internal nodes of a binary search tree.

⇒ It is assumed that keys are given in sorted order $k1 < k2 < \dots < kn$.

⇒ An extended binary search tree is obtained from the binary search tree by adding successor nodes to each of its internal nodes.



In the extended tree

- squares represents terminal nodes. These
- terminal nodes represent unsuccesful searches of the tree for key values. The searches did not end succesfully. because key

Scanned by CamScanner

value being searched are not stored in the tree

— round nodes represent internal nodes, these are actual keys stored in the tree

→ If a user searches a particular key in the tree 2 cases can occur

Case 1 – The key is found, so the corresponding weight p is incremented.

Case 2 – The key is not found so the corresponding q value is incremented.

p & q represents frequencies of successful searches & unsuccessful searches respectively.

⇒ n keys with their probabilities are given to us.

we want to create a BST using these n keys in such a way that average comparisons required to search key is minimum.

⇒ for this the node should be arranged on levels in such a way that the hight of the tree is minimum.

⇒ let P_i be the probability of searching key i, q_i be ~~Consider the eg.~~ probability of unsuccessful search

$$\boxed{\sum_{i=1}^{n} P_i + \sum_{i=0}^{n} q_i = 1}$$

⇒ if a search is successful it will terminate on internal node. The cost of searching is given by

Cost(k_i) = (level of k_i + 1) * P_i

⇒ if a search is unsuccessful, it terminates on external nodes (terminal nodes). The cost of searching is given by, level(e_i) * q_i

Contd on next pg

Scanned by CamScanner

The terminal node in the extended tree that is left successor of k_1 can be interpreted as representing all key value that are not stored and are less than k_1. Similarly, the terminal node in the extended tree that is the right successor of k_n, represents all key values not stored in the tree that are greater than k_n.

An obvious way to find an optimal binary search tree is to generate each possible binary search tree for the keys, calculate the weighted path length & keep the tree with minimum weighted path.

This search through all posible sol^n is not feasible since the no of search trees grows exponentially with "n".

An alternative is a recursive algo. a binary search tree has root, left subtree and right subtree. both subtrees must be the optimal binary search trees.

OBST(i,j) denotes the optimal binary search tree containing the keys k_i, k_{i+1}, k_j.

$w(i,j)$ - denotes the weight matrix for OBST(i,j)

$w(i,j)$ can be defined using formula.

$$w_{i,j} = \sum_{k=i+1}^{j} p_k + \sum_{k=i}^{j} q_k$$

$C(i,j)$ denote the cost matrix for OBST(i,j)

$$C_{i,i} = w_{i,j}$$

Scanned by CamScanner

$$C_{i,j} = w_{i,j} + \min_{i<k\leq j} \{ C_{i,k-1} + C_{k,j} \} \quad \text{--- (1)}$$

$R_{i,j}$ $0 \leq i \leq j \leq n$ denotes the root matrix for OBST (i, j)

Assign to $R_{i,j}$ the k value for which we obtain a minimum for eq^n (1)

Q: Find optimal binary search tree for keys do < if < int < while probability of succesful search p_1, p_2, p_3, p_4 and unsuccesful search

$$\begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 3 & 3 & 1 & 1 \end{array}$$

$$\begin{array}{ccccc} q_0 & q_1 & q_2 & q_3 & q_4 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 3 & 1 & 1 & 1 \end{array}$$

OBST is given by OBST (0,4) — involving the given 4 keys.

q_0 q_1 q_2 q_3 q_4 ← q
2 3 1 1 1

w \ j		0	1	2	3	4
	q_0	2	8	12	14	16
p_1	3 → 1		3	7	9	11
p_2	3 → 2			1	3	5
p_3	1 → 3				1	3
p_4	1 → 4					1

$w(0,1) = p_1 + q_1 + w(0,0)$
$= 3+3+2$
$= 8$

$w(1,2) = p_2 + q_2 + w(1,1)$
$= 3+1+3$
$= 7$

$w(2,3) = p_3 + q_3 + w(2,2)$
$= 1+1+1$
$= 3$

$w(3,4) = p_4 + q_4 + w(3,3)$
$= 1+1+1$
$= 3$

$$\boxed{w(i,j) = p_j + q_j + w(i, j-1)}$$

$w(0,2) = p_2 + q_2 + w(0,1)$
$= 3 + 1 + 8 = 12$

$w(1,3) = p_3 + q_3 + w(1,2)$
$= 1 + 1 + 7 = 9$

$w(2,4) = p_4 + q_4 + w(2,3)$
$= 1 + 1 + 3$
$= 5$

$w(0,3) = p_3 + q_3 + w(0,2)$
$= 1 + 1 + 12 = 14$

$w(1,4) = p_4 + q_4 + w(1,3)$
$= 1 + 1 + 9 = 11$

$w(0,4) = p_4 + q_4 + w(0,3)$
$= 1 + 1 + 14$
$= 16$

Cost

~~Root~~ (write k values)

i \ j	0	1	2	3	4
0	0	8	19	25	32
1		0	7	12	19
2			0	10	8
3				0	3
4					0

~~Cost~~ Root

			2	3	1	1	1
			q_0	q_1	q_2	q_3	q_4
		i \ j	0	1	2	3	4
		0	0	1	1	2	2
3	P_1	1		0	2	2	2
3	P_2	2			0	2	3 or 4
1	P_3	3				0	4
1	P_4	4					0

$$\boxed{c(i,j) = \min_{i < k \le j} \{ c(i, k-1) + c(k, j) \} + w(i,j)}$$

k is $\underline{1}$

$Cost(0,1) = \cancel{\omega(i,j) + min}$

$= \omega(0,1) + min\{Cost(0,0) + Cost(1,1)\}$

$= 8 + 0 + 0$

$= 8$

$Cost(1,2) = \omega(1,2) + min\{Cost(1,1) + Cost(2,2)\}$ k = 2

$= 7 + 0 + 0$

$= 7$

$Cost(2,3) = \omega(2,3) + min\{Cost(1,2) + Cost(3,3), \cancel{C(1,2) + C(3,3)}\}$ k = 3

$\cancel{= 9 + min\{0+}$

$= 3 + min\{7 + 0\}$

$= 10$

$Cost(2,4) = \omega(2,4) + min\{Cost(2,2) + Cost(3,4), Cost(2,3) + Cost(4,4)\}$ k = 3, k = 4

$= 5 + min\{0 + 3, 10 + 0\}$ k = 3

$= 5 + 3$

$= 8$

$Cost(3,4) = \omega(3,4) + min\{Cost(3,3) + Cost(4,4)\}$

$= \cancel{3 + min(0)}$

$= 3$

$$\text{Cost}(0,2) = w(0,2) + \min \begin{Bmatrix} \text{Cost}(0,0) + \text{Cost}(1,2), & k=1 \\ \text{Cost}(0,1) + \text{Cost}(2,2) & k=2 \end{Bmatrix}$$

$$= 12 + \min \begin{Bmatrix} 0+7, \\ 8+0 \end{Bmatrix}$$

$$= 12+7$$

$$= 19$$

$$\text{Cost}(1,3) = w(1,3) + \min \begin{Bmatrix} \text{Cost}(1,1) + \text{Cost}(2,3), & k=2 \\ \text{Cost}(1,2) + \text{Cost}(3,3) & k=3 \end{Bmatrix}$$

$$= 9 + \min \begin{Bmatrix} 0+3, \\ 7+0 \end{Bmatrix}$$

$$= 12$$

$$\text{Cost}(0,3) = w(0,3) + \min \begin{Bmatrix} \text{Cost}(0,0) + \text{Cost}(1,3), & k=1 \\ \text{Cost}(0,1) + \text{Cost}(2,3) & k=2 \\ \text{Cost}(0,2) + \text{Cost}(3,3) & k=3 \end{Bmatrix}$$

$$= 14 + \min \begin{Bmatrix} 0+12, \\ 8+3, \\ 19+0 \end{Bmatrix}$$

$$= 14 + 11$$

$$= 25 \qquad k=2$$

$$\text{Cost}(1,4) = w(1,4) + \min \begin{Bmatrix} \text{Cost}(1,1) + \text{Cost}(2,4), & k=2 \\ \text{Cost}(1,2) + \text{Cost}(1,3), & k=3 \\ \text{Cost}(1,3) + \text{Cost}(4,4) & k=4 \end{Bmatrix}$$

$$= 11 + \min \begin{Bmatrix} 0+8, \\ 7+12, \\ 12+0 \end{Bmatrix}$$

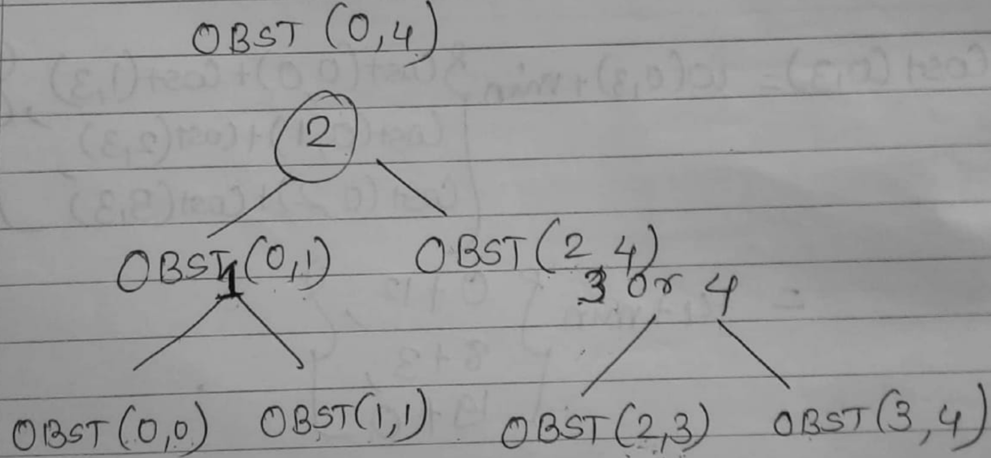
$$= 11+8 \qquad = 19$$

$$cost(0,4) = w(0,4) + \min \begin{cases} cost(0,0) + cost(1,4), & k=1 \\ cost(0,1) + cost(2,4), & k=2 \\ cost(0,2) + cost(3,4), & k=3 \\ cost(0,3) + cost(4,4) & k=4 \end{cases}$$

$$= 16 + \min \begin{cases} 0+19, \\ 8+8, \\ 19+3, \\ 25+0 \end{cases}$$

$$= 16 + 16$$

$$= \underline{\underline{32}}$$



for

Q.2] OBST

	k_1		k_2		k_3		k_4
	Cout	<	float	<	if	<	while

$p_1 \frac{1}{20}$ $p_2 \frac{1}{5}$ $p_3 \frac{1}{10}$ $p_4 \frac{1}{20}$

$q_0 \frac{1}{5}$ $q_1 \frac{1}{10}$ $q_2 \frac{1}{5}$ $q_3 \frac{1}{20}$ $q_4 \frac{1}{20}$

→ The ultimate cost is given by COST(0, 4)
Here in order to be convinient multiply prop by 20 and recomputed probabilities are

Scanned by CamScanner

$p_1 = 1$ $\qquad q_0 = 4$

$p_2 = 4$ $\qquad q_1 = 2$

$p_3 = 2$ $\qquad q_2 = 4$

$p_4 = 1$ $\qquad q_3 = 1$

$\qquad\qquad q_4 = 1$

W

	i \ j	q_0 (4) 0	q_1 (2) 1	q_2 (4) 2	q_3 (1) 3	q_4 (1) 4
	0	4	7	15	18	20
1 — p_1	1		2	10	13	15
4 — p_2	2			4	7	9
2 — p_3	3				1	3
1 — p_4	4					1

Root

i \ j	0	1	2	3	4
0	0	1	2	2	2
1		0	2	2	2
2			0	3	3
3				0	4
4					0

Cost

	i \ j	q_0 0	q_1 1	q_2 2	q_3 3	q_4 4
	0	0	7	22	32	$\boxed{39}$
p_1	1		0	10	20	27
p_2	2			0	7	12
p_3	3				0	3
p_4	4					0

$k=1$

$$Cost(0,1) = W(0,1) + C(0,0) + C(1,1) = 7+0+0 = 7$$

$$Cost(0,2) = W(0,2) + \min \begin{cases} C(0,0) + C(1,2) \\ C(0,1) + C(2,2) \end{cases} = 15 + 7 = 22$$

$k=2$ & $k=3$

$$Cost(1,3) = W(1,3) + \min \begin{cases} C(1,1) + C(2,3) & k=2 \\ C(1,2) + C(3,3) & k=3 \end{cases}$$

$$= 13 + 7 = 20$$

Scanned by CamScanner

k=3 or k=4.

$$Cost(2,4) = w(2,4) + \min\begin{cases} \overset{0}{c(2,2)} + \overset{3}{c(3,4)} & k=3 \\ \underset{7}{c(2,3)} + \underset{0}{c(4,4)} & k=4 \end{cases}$$

$$= 9 + 3 = 12$$

k=1 or k=2 or k=3

$$Cost(0,3) = w(0,3) + \min\begin{cases} \overset{0}{c(0,0)} + \overset{20}{c(1,3)} & k=1 \\ \underset{7}{c(0,1)} + \underset{7}{c(2,3)} & k=2 \\ \underset{22}{c(0,2)} + \underset{6}{c(3,3)} & k=3 \end{cases}$$

$$= 18 + 14$$

$$= 32$$

k=2,3,4

$$Cost(1,4) = w(1,4) + \min\begin{cases} \overset{0}{c(1,1)} + \overset{12}{c(2,4)} & k=2 \\ \overset{10}{c(1,2)} + \overset{3}{c(3,4)} & k=3 \\ \underset{20}{c(1,3)} + \underset{0}{c(4,4)} & k=4 \end{cases}$$

$$= 15 + 12$$

$$= 27$$

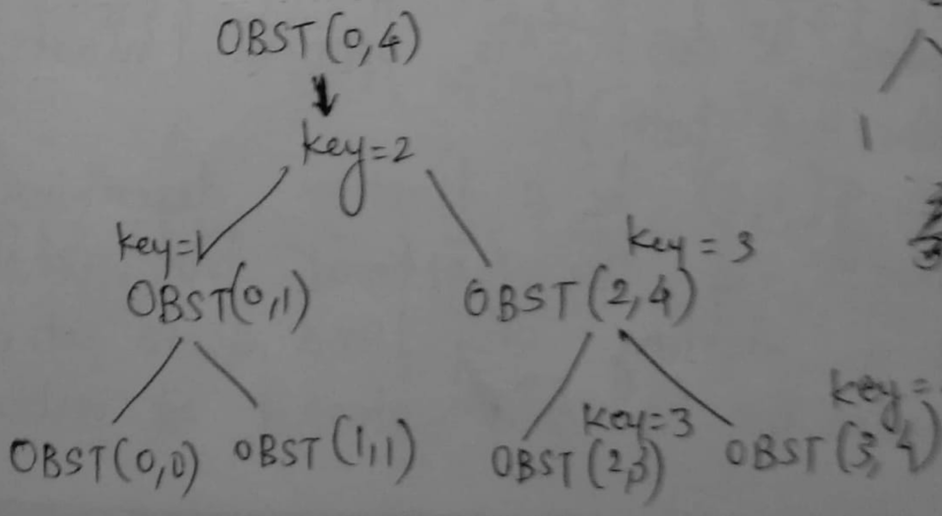
k=1,2,3,4

$$Cost(0,4) = w(0,4) + \min\begin{cases} \overset{0}{c(0,0)} + \overset{27}{c(1,4)} & k=1 \\ \overset{7}{c(0,1)} + \overset{12}{c(2,4)} & k=2 \\ \overset{22}{c(0,2)} + \overset{3}{c(3,4)} & k=3 \\ \overset{32}{c(0,3)} + \overset{0}{c(4,4)} & k=4 \end{cases}$$

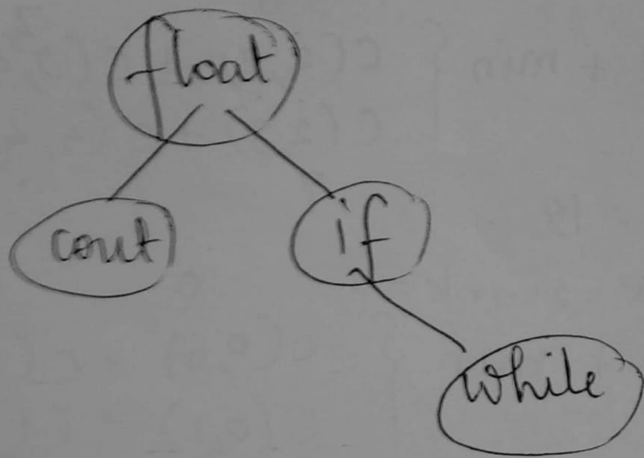
$$= 20 + 19$$

$$= \underline{39}$$

(*) OBST



Resultant OBST



Scanned by CamScanner