SNS COLLEGE OF ENGINEERING

Coimbatore-107

COURSE NAME: ANALYSIS OF ALGORITHM

II YEAR/ IV SEMESTER

UNIT – III

DYNAMIC PROGRAMMING

Topic

Optimal Binary Search Tree

Dynamic Programming
Optimal Binary Search Tree (OBST)

⇒ It is an efficient searching algorithm, that works on sorted arrays.

⇒ It minimizes total search cost considering the frequency of searches [Number of times key is Searched]. for each element.

Example: (A Sorted Array)

Key	10	20	30	40	50
Freq	1	5	10	3	2

Step 1:
Pick the element with maximum frequency & make it root.

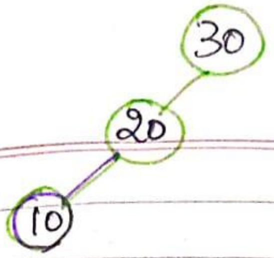
30

Step 2:
⇒ Build Left & Right Subtrees with key elements 10 and 20 (left) and 40, 50 (as right).

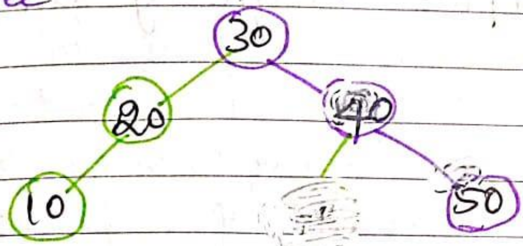
Step 3: Construct left subtree with key 10 and 20.

By Rule of 'BST' - 10 and 20 are smaller than root. So construct it as Left subtree

Also frequency of 20 is 5 closer to root. So take it as Left child Immediate



Step 4 : Construct Right subtree with key 40 and 50.
By Rule of BST - 40 and 50 are greater. So construct it as Right subtree



---x---x---x---x---

Construct OBST using Dynamic programming (for same problem)
We use Dynamic programming to minimize the search cost and construct tree.
Step 1 : Sort the keys in Ascending order
Step 2 : Calculate minimum cost of Searching for all possible subtrees.
Step 3 : Build optimal tree.
Define the cost :
(i) c[i,j] → Minimum search cost from index i to j.
(ii) w[i,j] → Sum of frequencies from i to j

Recurrence Relation:

cost of subtree with root 'k' can be given as

$$C[i,j] = \min_{i<k<j} [C(i,k-1) + C(k+1,j) + W(i,j)]$$

Goal:

To find 'k' that minimizes the cost for each subtree

Step 4: Compute Frequency Sum Table W[i,j]

1 { (i) W[10,10] = Frequency of 10 = 1
(ii) W(10,20) = Frequency of 10 & 20 = 1+5 = 6
(iii) W(10,30) = Frequency of 10,20,30 = 1+5+10 = 16
(iv) W(10,40) = Frequency of 10,20,30,40 = 1+5+10+3 =
(v) W(10,50) = Freq of 10,20,30,40,50 = 1+5+10+3+2 = 2

2 { (vi) W(20,20) = Freq of 20 = 5
(vii) W(20,30) = Freq of 20,30 = 5+10 = 15
(viii) W(20,40) = Freq. of 20,30,40 = 5+10+3 = 18
(ix) W(20,50) = Freq of 20,30,40,50 = 5+10+3+2 = 20

3 { (x) W(30,30) = Freq of 30 = 10
(xi) W(30,40) = Freq of 30,40 = 10+3 = 13
(xii) W(30,50) = Freq of 30,40,50 = 10+3+2 = 15

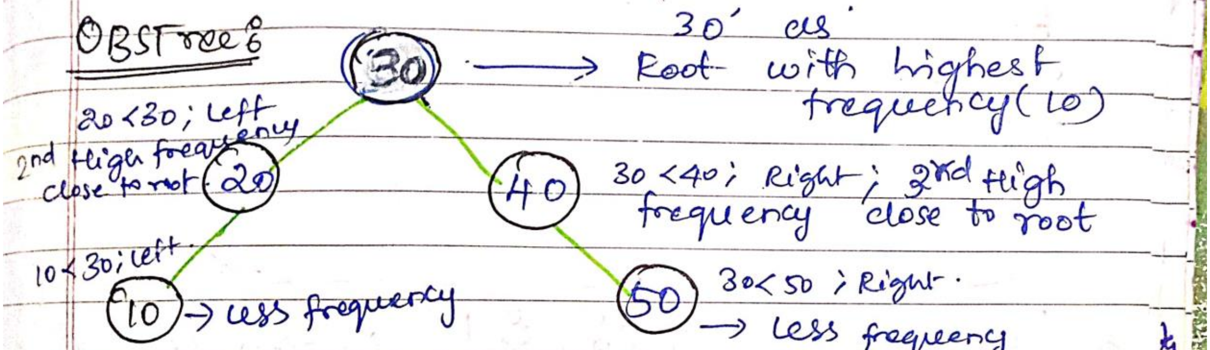
4 { (xiii) W(40,40) = Freq of 40 = 3
(xiv) W(40,50) = Freq of 40,50 = 3+2 = 5

5 ← (xv) W(50,50) = Freq of 50 = 2

Step 4: DP Table compute C[i,j] → minimum search cost

i	j	c[i,j]	
10	10	1	(5)
20	20	5	(2)
30	30	10	(1)
40	40	3	(3)
50	50	2	(4)

OBST ree:



— x — x —