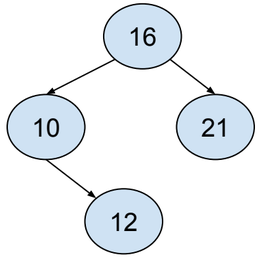
Optimal binary search trees:

Nodes: 10, 12, 16, 21
Freq 4, 2, 6, 3



cost(i,j) = sum (freq of i to j) +
min (
cost (i+1,j) → 1st node i as root,
cost(i,i) + cost (i+2,j) → 2nd node i+1 as root,
cost(i,i+1) + cost (i+3,j) → 3rd node i+2 as root,
.
.
cost (i,j-1) → last node j as root
)

Cost	0	1	2	3
0	4	8/0	20/2	26/2
1		2	10/2	16/2
2			6	12/2
3				3

= sum of frequencies of nodes + min(cost of possible branches)
= 4 + 2 + 6 + 3 + min (16 , (4+12), (8+3), 20)
= 15 + 11
= 26