Todays Content

1. Sum Root to leaf
2. Count Path Sums
3. Max Root Sum in BT
4. Max Path Sum in BT

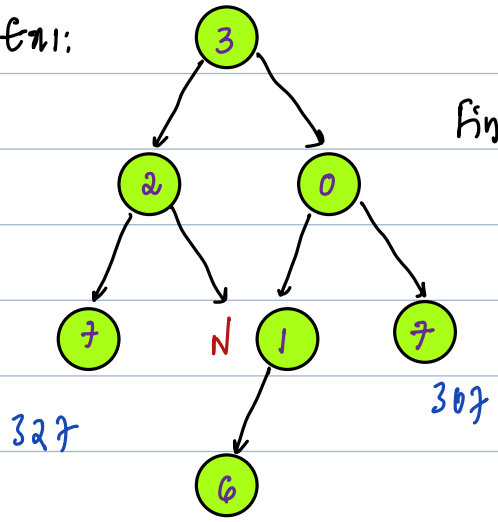
Q: Sum of paths:

Each Node has digit: 0-9

Calculate sum of all path going From root → leaf

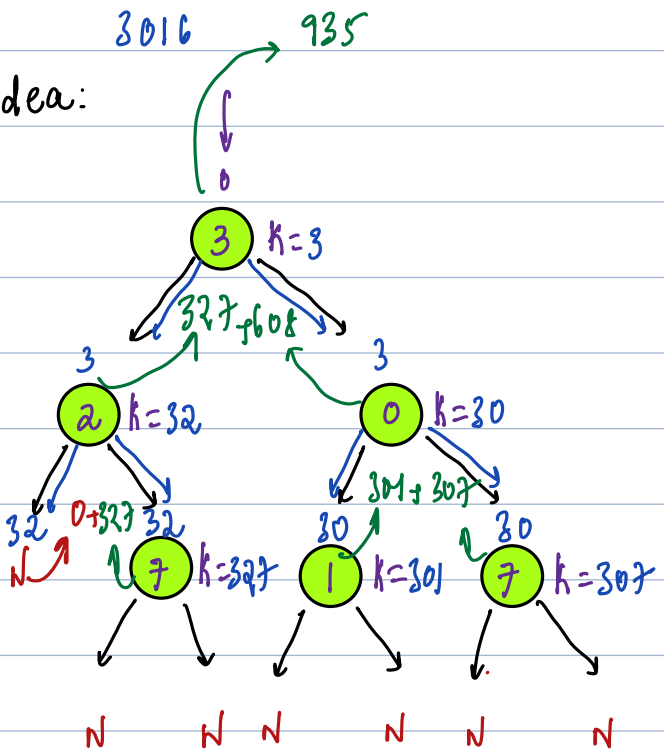
Note: Final ans will be int Range.

Ex1:




Final ans = 327 + 3016 + 307 = 3650

Idea:




```
int solve(Node *root){
    int ans = pathSum(root, 0);
    return ans;
}
```

TC: O(N)

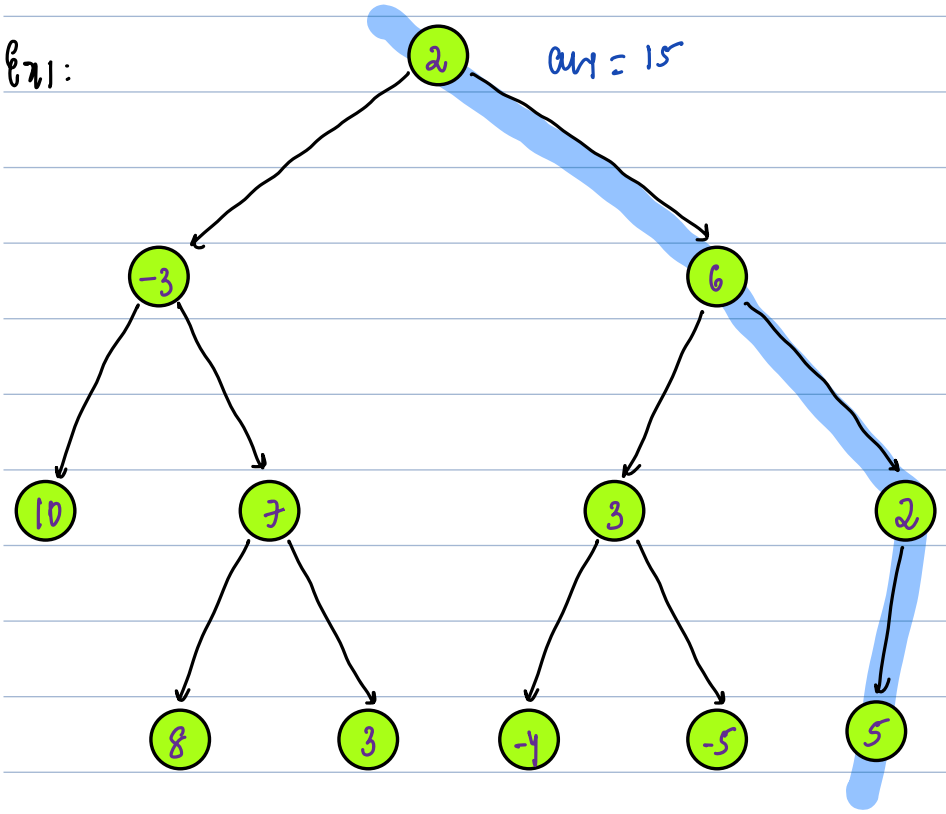
```
int pathSum(Node *root, int k){
    if(root==nullptr){return 0;}
    k = k*10 + root->data;
    if(root->left==nullptr &&
       root->right==nullptr){
        return k;
    }
    int l = pathSum(root->left, k);
    int r = pathSum(root->right, k);
    return l+r;
}
```

```
int leafSum(Node *root, int sum){

}
int Solve(Node *root){

}
```

# Root Sum

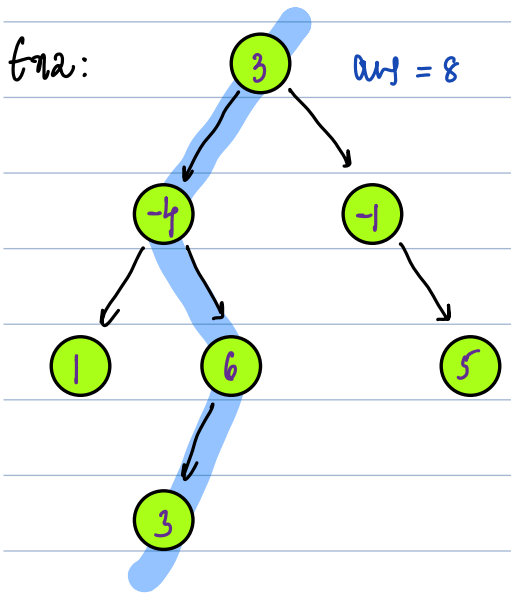
2B Given a B.T return max rootsum, whose path starts at root node but can end at any node

Ex1:



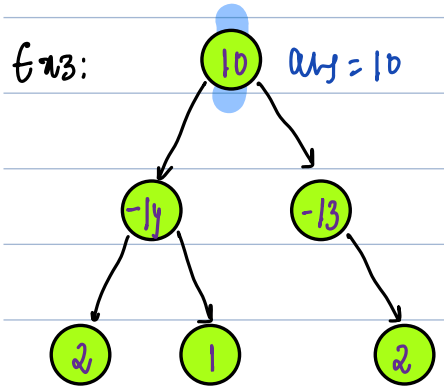

ans = 15

Ex2:




ans = 8

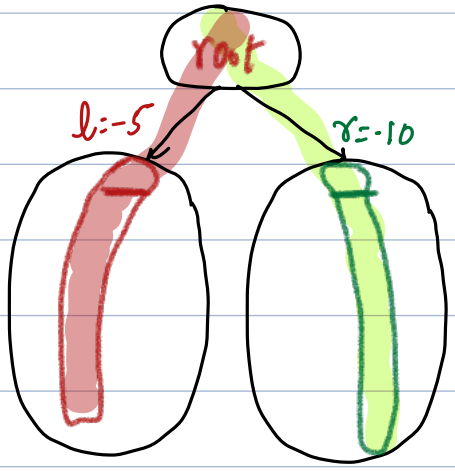
Ex3:



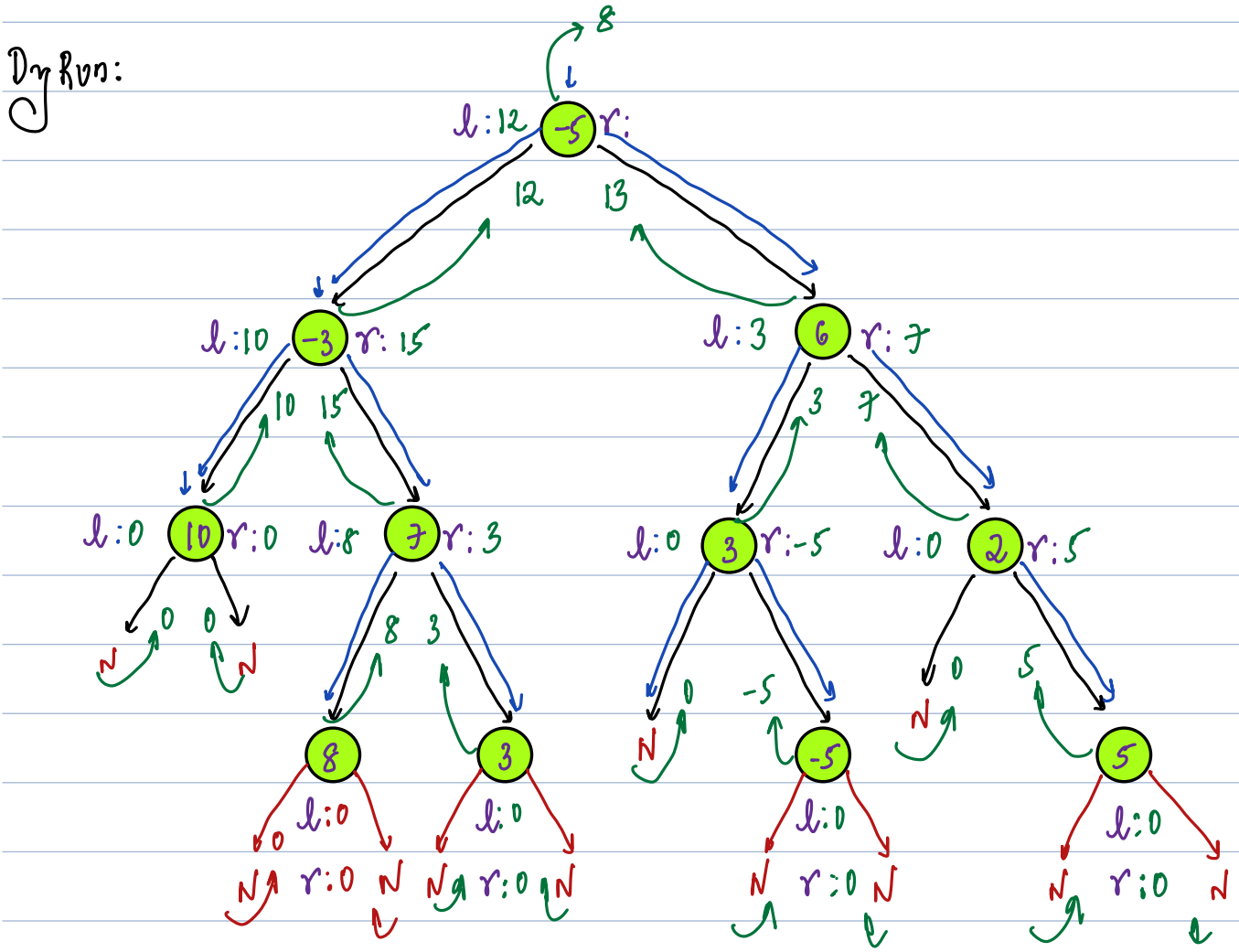

ans = 10

Ass: Given root node, calculate max root sum {start root} & return it.

```
int RootSum(Node* root){
    if(root==nullptr){return 0;}

    int l= RootSum(root->left);
    int r= RootSum(root->right);
    return root->data+max(l,max(r,0));
}
```




Dry Run:




#obs: Any Tree related questions:

Top down or Bottom up + { Tree Traversal }

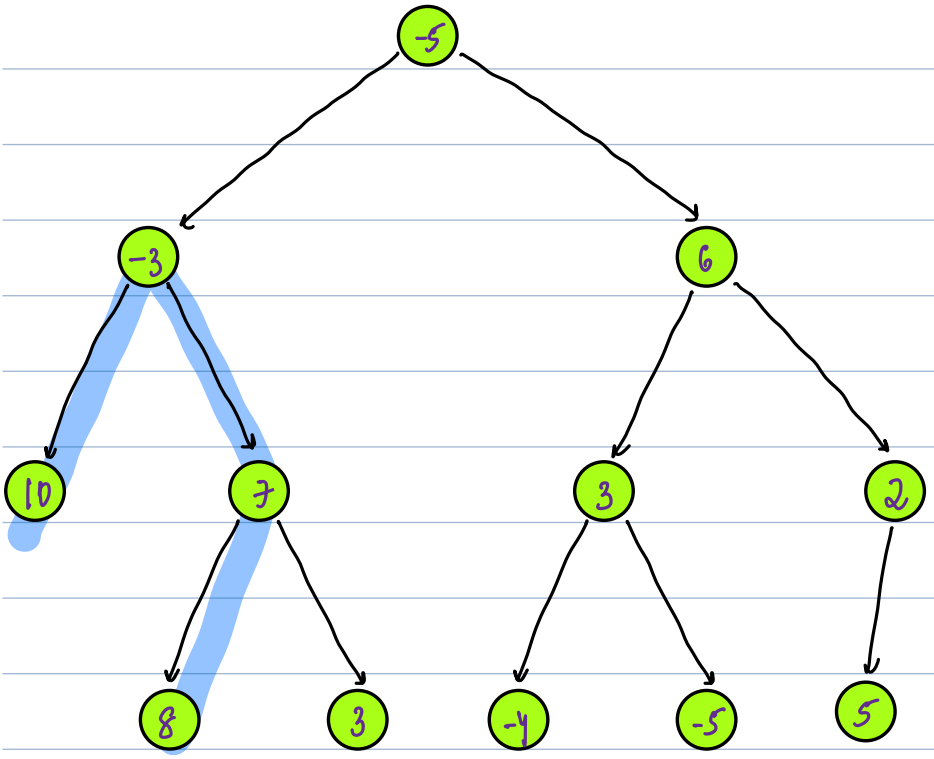
Top down: What info should root pass to children

Bottom up: What info should left & right subtree should pass.

38 Max Path Sum in BT

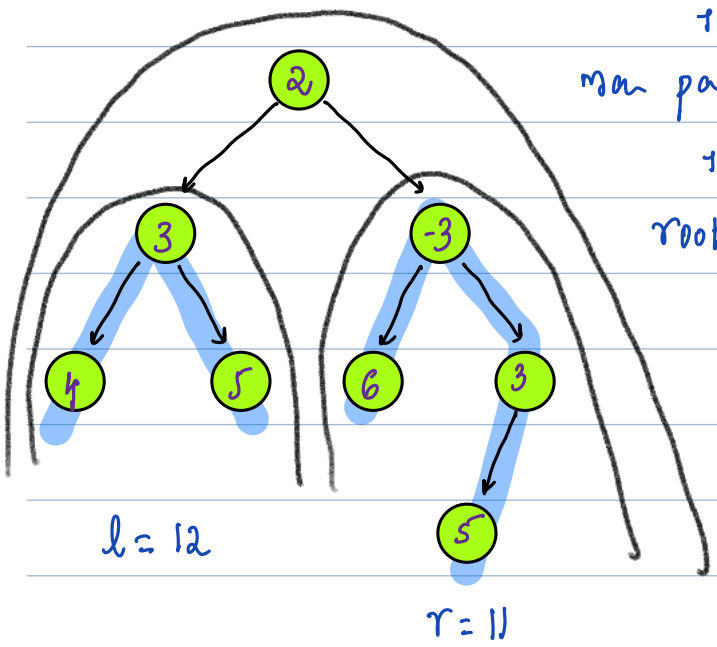
Path can start at any node and end at any node.

Ex: Ans = 22




Idea: Max path sum in BT = max path sum in LST + max path sum in RST + root

Invalid statement, we cannot combine them like that.

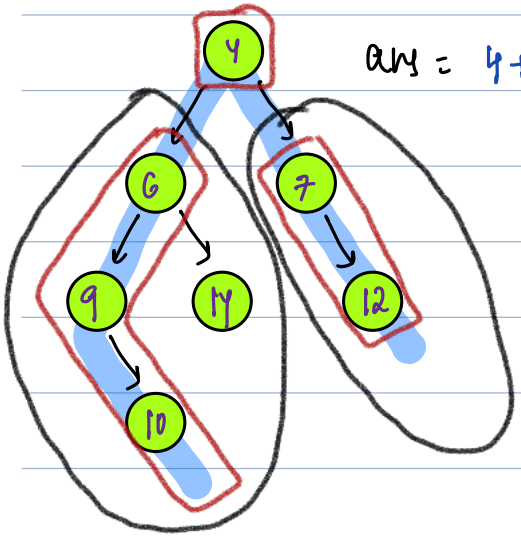



l = 12

r = 11

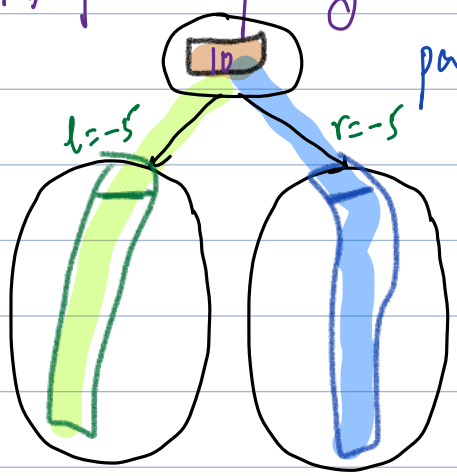
#Obs1: Max path sum path will pass through root node of a subtree

Ex: Max path sum path passing through 4?




ans = 4 + root sum of Lst = 25 + root sum of Rst = 19

#Con: Max path sum passing through rootnode of a subtree?




pathSum = root +
max(rootSum of Lst, 0) + max(rootSum of Rst, 0)

#Idea2: Since we are not sure, from which node max path sum will travel,
Take every node as root node of subtree
: Calculate max path sum passing through that node

```
l = rootSum(node->left);
r = rootSum(node->right);
ps = node + max(l,0) + max(r,0);
ans = max(ans, ps);
}
return ans;
```

Tc: O(N*N) = O(N^2)

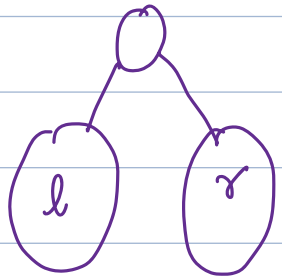
#Issue: In above approach, for every node we calculate root sum in left & right, which is time taking.

Ideas: Apply rootsum from root node once.

For every node, it will calculate:

Rootsum of LST & Rootsum of RST

```
int ans = INT_MIN;
int RootSum(Node* root){ TC: O(N)
    if(root==nullptr){return 0;}

    int l = RootSum(root->left); #
    int r = RootSum(root->right); #
    int ps = root->data + max(l,0) + max(r,0);
    ans = max(ans, ps);
    return root->data + max(l, max(r,0)); #
}
```

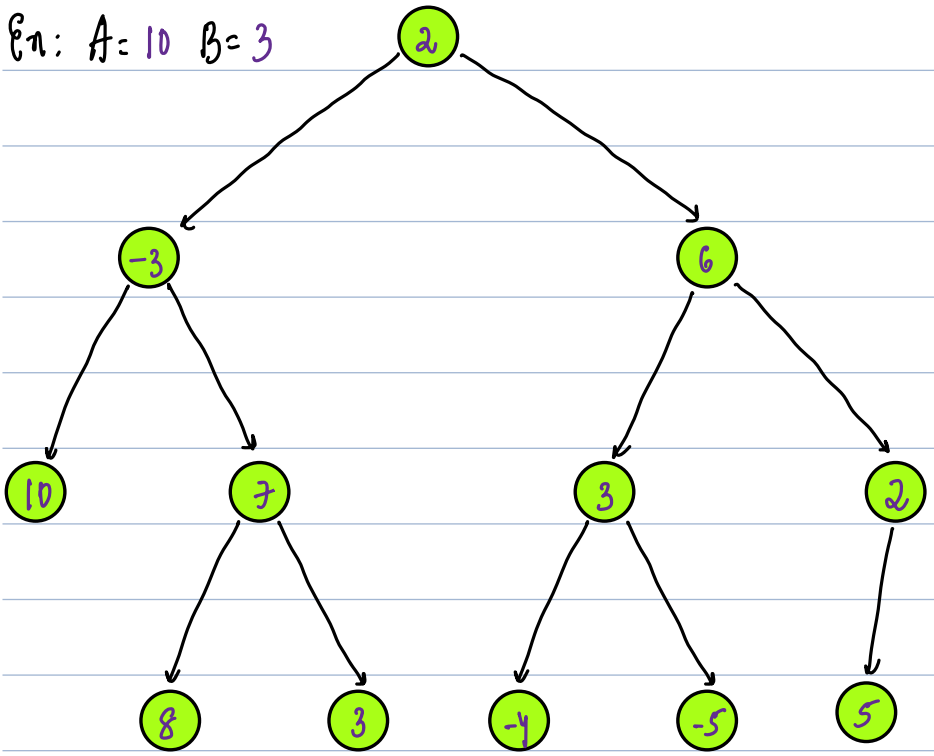



```
int Solve(Node *root){
    ans = INT_MIN;
    RootSum(root);
    return ans;
}
```

48 length of path

Given 2 Nodes A & B, print path between 2 nodes from A → B

Ex: A = 10 B = 3

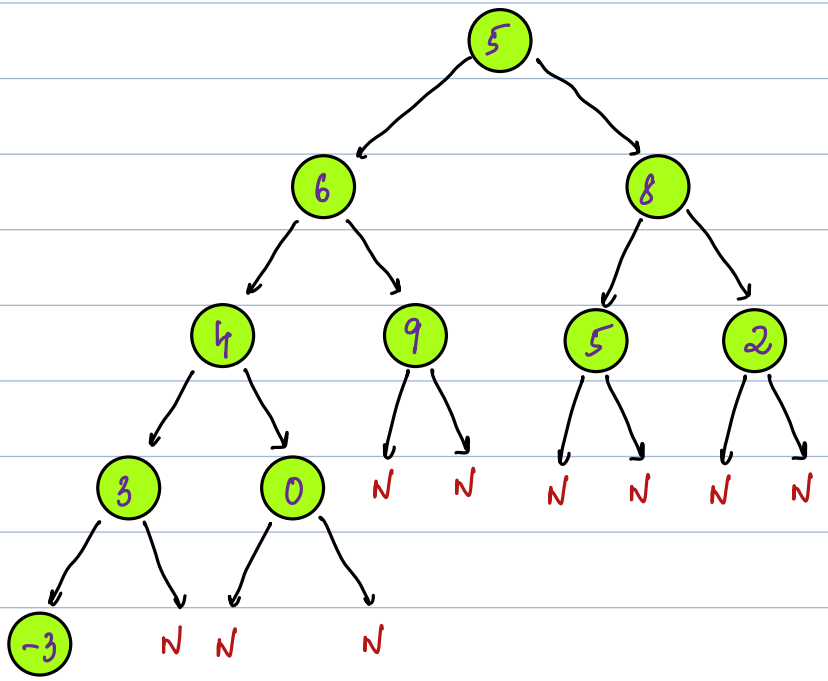



Idea:

SQ How many paths are there starting from root node with sum == k.

Note: Elements may be +ve or -ve.

Ex: k = 15




```
int countTarget(Node *root, int k){
```

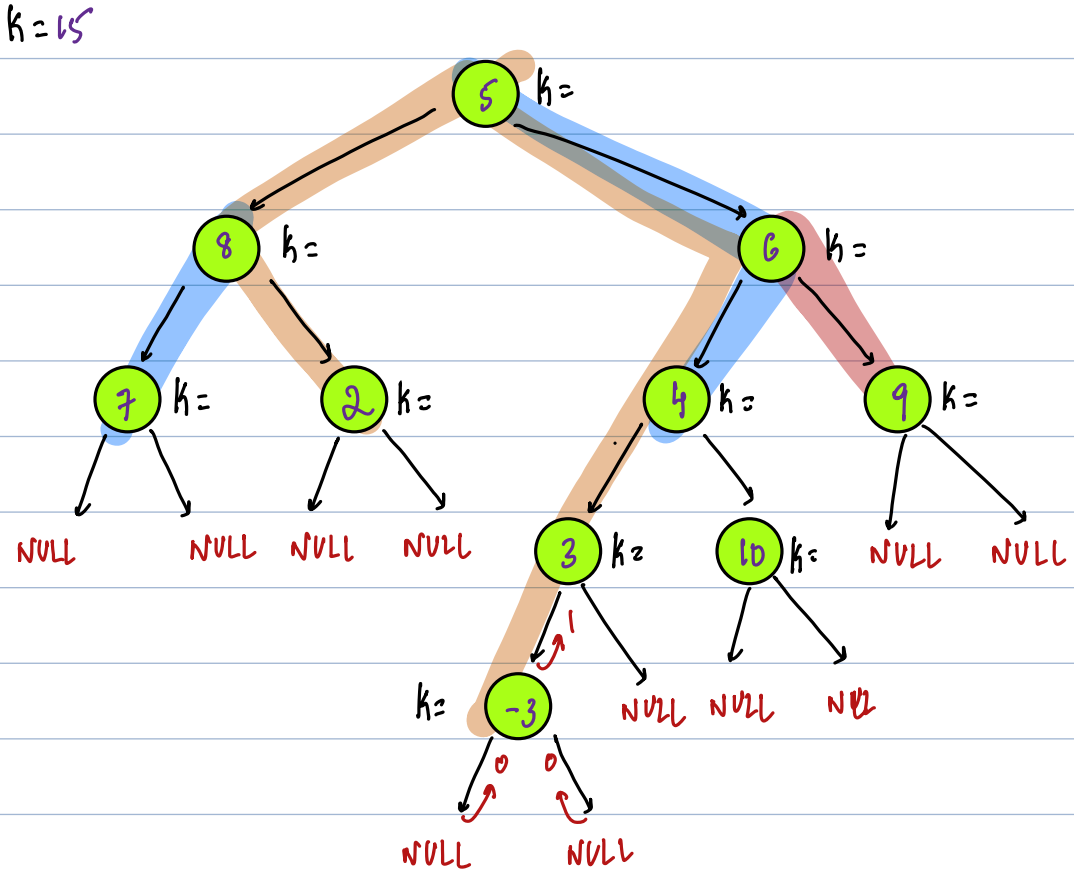
3

58: Count TargetSum : Hard

How many paths are there starting from a node with sum == k.

Path should come down

Note: Elements may be +ve or -ve.



BST1 } Linked List Basics + Recursion + 1st Class of BT : 1st
BST2 }

Heaps } Library
Tries } class Node | Struct

↳ BackTracking } Recursion
Dynamic Program } Recursion
Graphs { == }

**Alok** has three daughters. His friend **Shyam** wants to know the ages of his daughters. Alok gives him a first hint.

**1.** The product of their age is 72.
**Shyam** says this is not enough information **Alok** gives him a second hint.

**2.** The sum of their ages is equal to my house number.
**Shyam** goes out and looks at the house number and tells "I still do not have enough informati to determine the ages". **Alok** admits that **Shyam** can not guess and gives him the third hint

**3.** The oldest girl likes strawberry ice cream.
**Shyam** is able to guess after the third hint. Can you guess what are the ages of the three daughters?