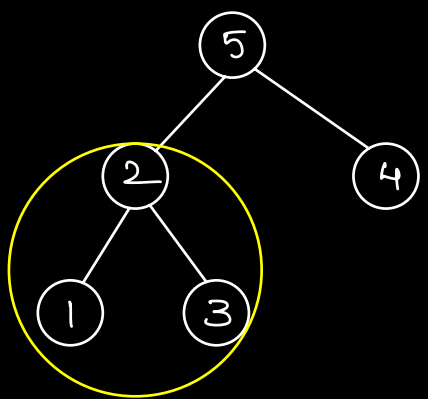
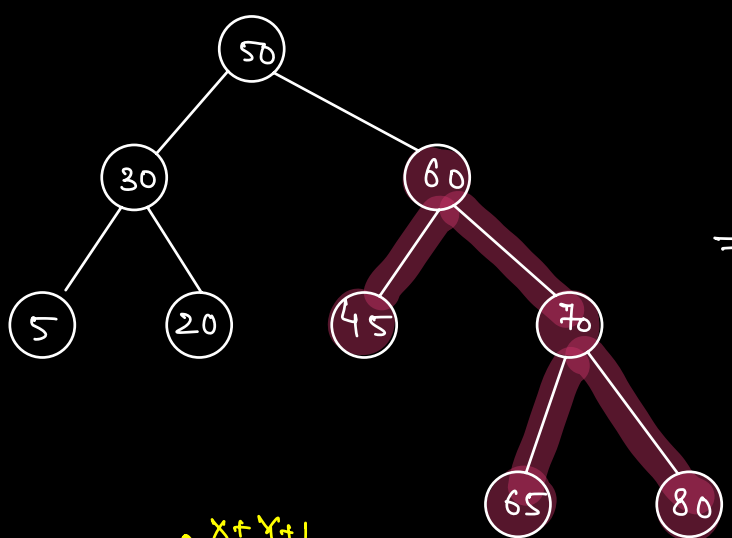
Q. Given a Binary Tree, Return the size of max BST subtree inside it.

Amazon
MS.
Atlassian.

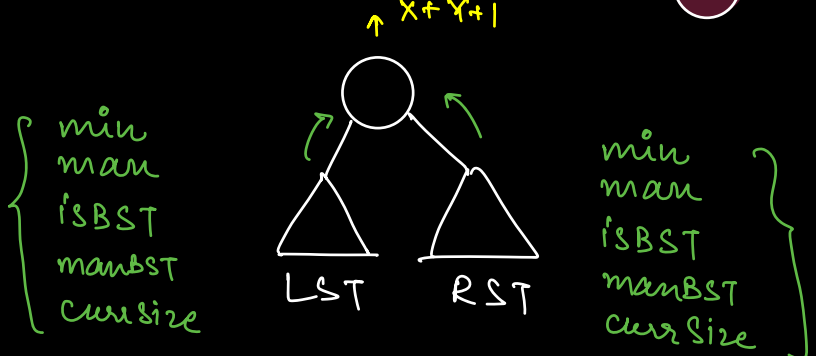
# of Nodes.



⇒ 3



⇒ 5

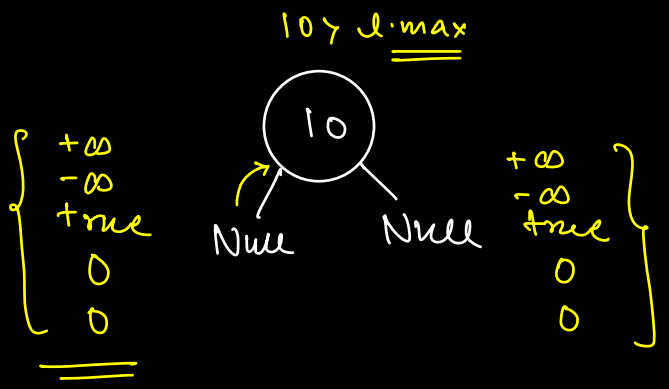


```
Class TreeInfo {
    int min;
    int max;
    bool isBST;
    int maxBST;
    int currSize;
}

TreeInfo maxBST(root) {
    if (root == NULL) {
        return new TreeInfo(+∞, -∞, true, 0, 0);
    }
    TreeInfo l = maxBST(root.left);
    TreeInfo r = maxBST(root.right);
    if(l.isBST && r.isBST && root.val > l.max
                && root.val < r.min)
        new TreeInfo(l.min, r.max, true,
            l.maxBST + r.maxBST + 1,
            l.currSize + r.currSize + 1);
    }

    return new TreeInfo(min(l.min, r.min, root.val),
```

max (l.max, r.max, root.val), false, max (l.maxBST r.maxBST),

l.currSize + r.currSize + 1) ;

3



Q. Find the rank of a given string A, among its permutations, sorted lexicographically.
(No duplicates)

A: "dcba"

abcd	bacd	cabd	dabc
abdc	badc	cadb	dacb
acbd	bcad	cbad	dbac
acdb	bcda	cbda	dbca
adbc	bdac	cdab	dcab
adcb	bdca	cdba	dcba

⇒ 24

dcba

1. First character less than d.

a/b/c _ _ _ (d)

a _ _ _ → 3!

b _ _ _ → 3!

c _ _ _ → 3!

$\Rightarrow 3 \times 3!_0 = 18$

2. d □ _ _ < dcba

Strings with 2nd char less than c.

d □ _ _ ⇒ (a / b)

d a _ _ → 2!

d b _ _ → 2!

$2 \times 2!_0 = 4$

3. d c + _ < dcba

a 1!

⇒ 1

Count = 23

rank = Count + 1

Str: TODAY
0 1 2 3 4

(I) 3 _ _ _ _ < A _ _ _ _ ⇒ 4!
↓ T D _ _ _ _ ⇒ 4!
T O _ _ _ _ ⇒ 4!

⇒ 3 × 4! = 72

(II) T _ _ _ _ < 2 × 3! = 12
D A
TA _ _ _
TD _ _ _

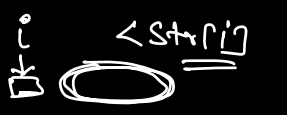
(III) T O _ _ _ ⇒ TOA _ _ = 2!
↓ = 2
< D
↓
(A)

(IV) T O D _ _ ⇒ 0 × 1! = 0
↑
< A

TODA ⇒

Code :

```
rank ( String str ) {
    count = 0
    for( i=0; i< N; i++ ) {
        Char ch = str[i];
        // Count the no. of characters
        // < ch in str from i+1 to N-1.
O(N) ← n = countCharacters(str, ch, i);
O(N) ← fact = factorial( N-i-1 )
        count += n * fact;        Use MOD.
    }
    return count + 1;        (n%M * fact%M)%M
}
```



TC : $O(N^2)$

SC : $O(1)$

⇒ With duplicates :-

N distinct characters ⇒ N!

N characters with duplicates.

$\underbrace{aa}_{2}\underbrace{bb}_{2} \Rightarrow \frac{4!}{2!\,2!} = \frac{24}{4} = 6$

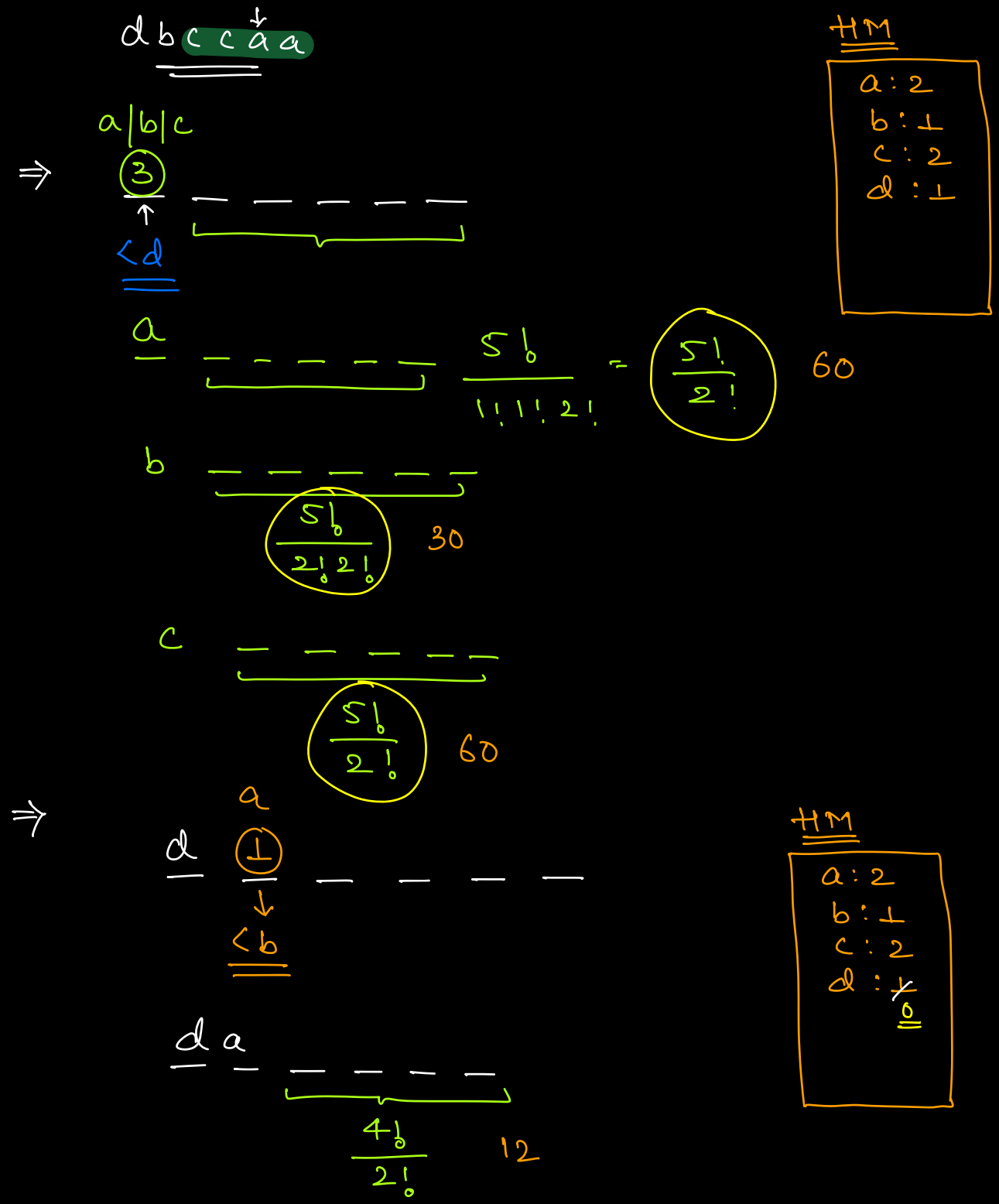
aabb
abab
abba
baab
baba
bbaa

} 6

$aabbcc \Rightarrow \frac{6!}{2!\,2!\,2!} = \frac{720}{8} = 90$

$aaaxxz22pppp = \frac{12!}{3!\,2!\,3!\,4!} = \text{L}$

$aaabbc \Rightarrow \frac{6!}{3!\,2!\,1!} = \frac{6 \cdot 120}{6 \cdot 2} = 60$


d b c c a a
HM
a : 2
b : 1
c : 2
d : 1
a | b | c
3
< d
a
5!/1!1!1!2! = 5!/2!
60
b
5!/2!2!
30
c
5!/2!
60
a
d
1
< b
d a
4!/2!
12
HM
a : 2
b : 1
c : 2
d : ✗ 0

⇒ d b (1)^{a} _ _ _

↑ ↖c

$\frac{3!}{2!_0} = 3$

HM

a : ~~2~~ 1	
b : ~~1~~	
c : 2	
d : ~~1~~	

⇒ d b c (1)^{a} _ _

↑ ↖c

$2!_0 \Rightarrow 2!_0 = 2$

HM

a : ~~2~~ 1	
b : ~~1~~	
c : ~~2~~ 1	
d : ~~1~~	

⇒ d b c c _ _ $= 0$

↑ ↖a

HM

a : 2	
b : ~~1~~	
c : ~~2~~ 1	
d : ~~1~~	

d b c c a _ $= 0$

↑ ↖a

⇒ 168

* $\left(\frac{a}{b}\right) \% M \Rightarrow$ Inverse Modulo.
(Fermat's Little Theorem)

* $$fact[i] = (fact[i-1] \% M * i \% M) \% M$$

* Maintain a HashMap.