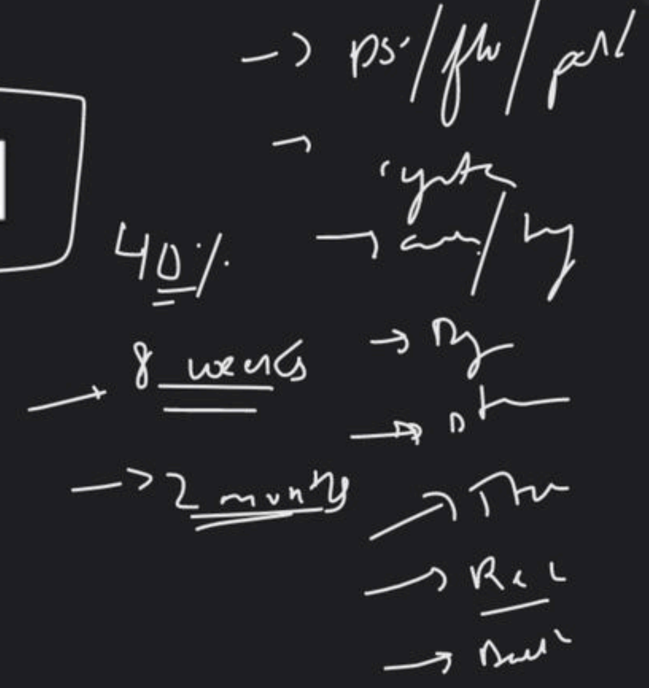
Doubt Clearing Session - Part VIII

Foundation Course on Data Structures & Algorithm - III



Love Babbar • Lesson 8 • June 5, 2022

H/W

→ K^{th} character in Grammer

→ Red Zone Problem

→ Median of 2 sorted arrays (same size)

→ different size

6 Question

→ STL Video

→ 318 LC → Max prod of word length

→ inversion count

Video Solutions

Doubt Clear

Rec & BS

H/W

↳ P

↳ SS

↳ RM

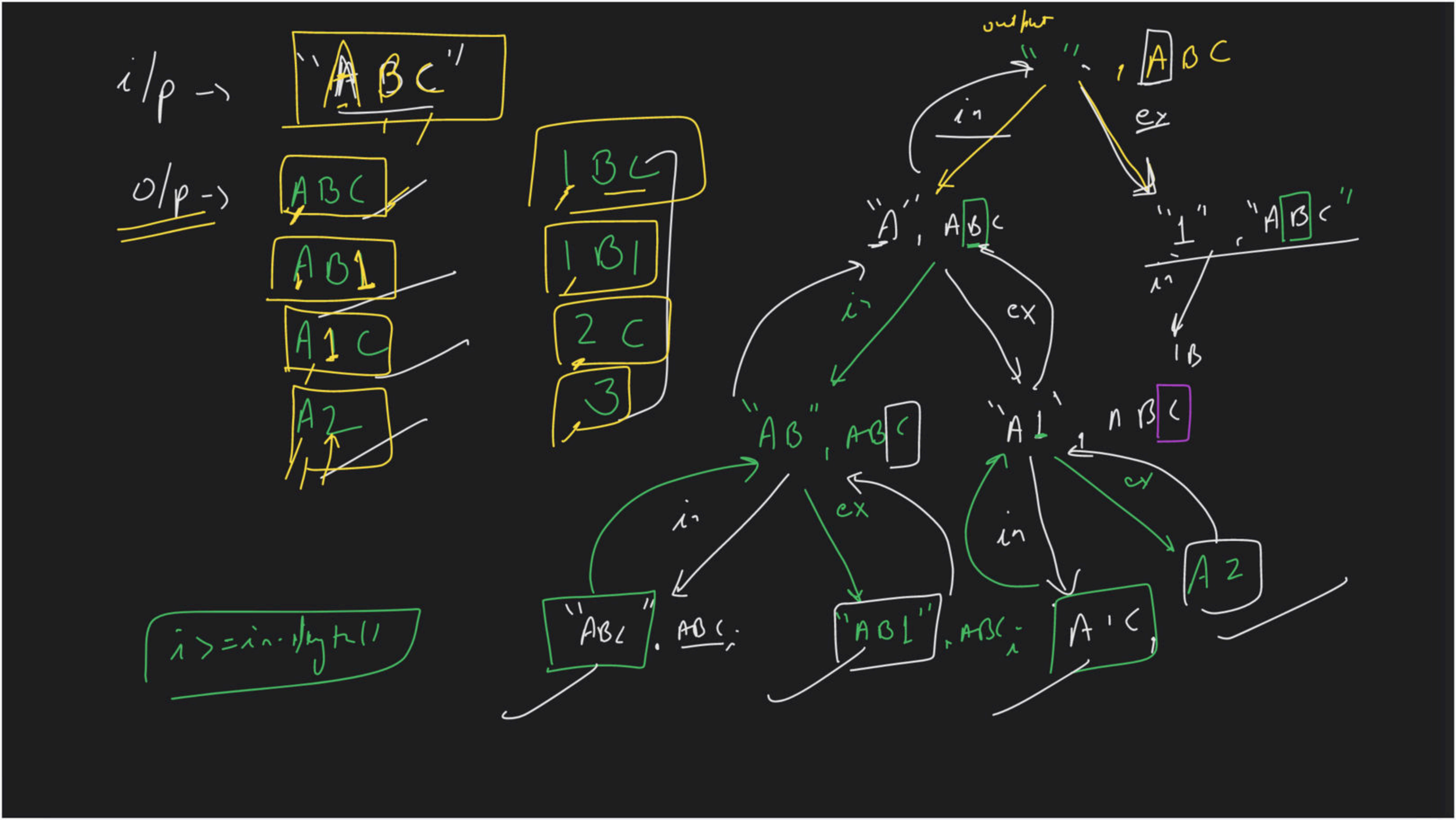
↳ n4

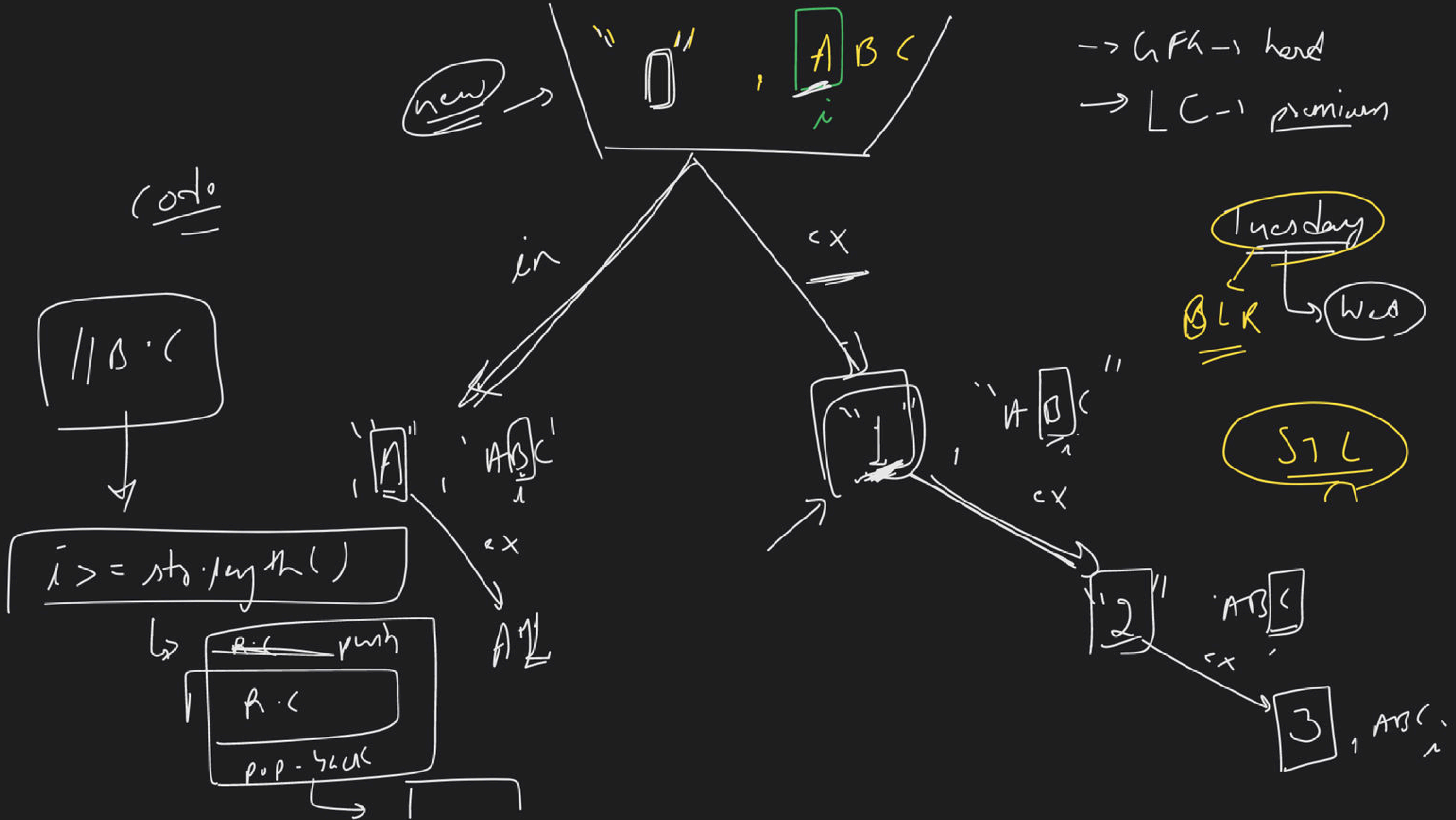
50 Question

22K

2500

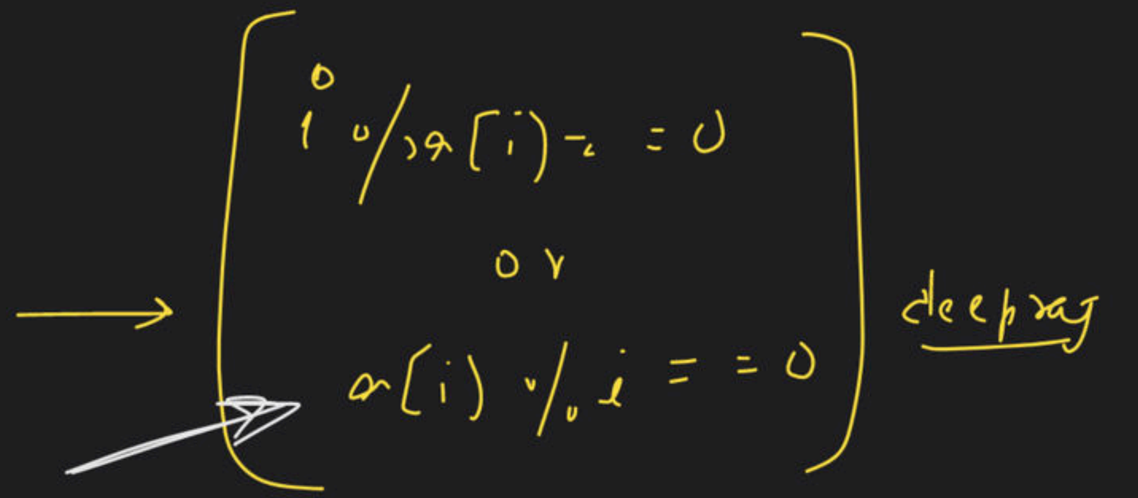
7 Ques

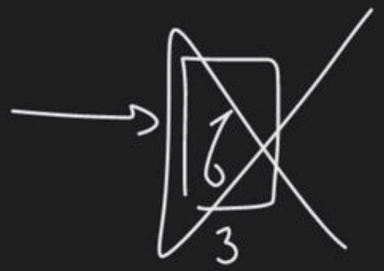

i/p ->
"ABC"
o/p ->
ABC
AB1
A1C
A2
1BC
1B1
2C
3
output
ABC
in
ex
"A"
ABC
"1"
"ABC"
in
1B
in
ex
"AB"
ABC
"A1"
1BC
in
ex
in
ex
"AB1"
ABC
"AB1"
ABC
A1C
A2
i>=in.length()


new
ABC
in
ex
-> GFG -> hard
-> LC -> premium
code
Tuesday
BLR
Wed
STL
i >= str.length()
push
pop - stack

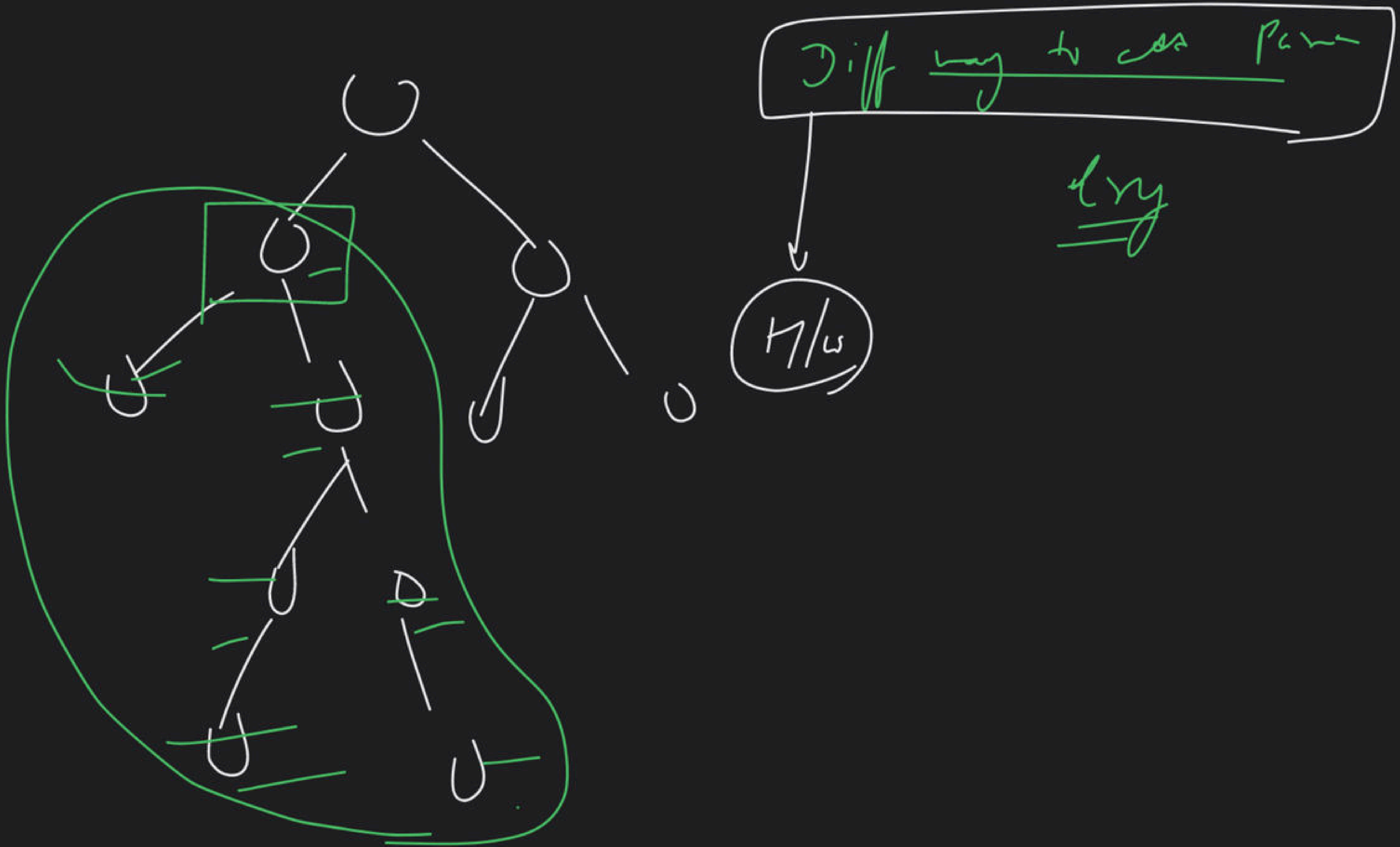
→ elimination game

→ Beautiful arrangement





exams



→ ((2 - 1) - 1)

1 - 1 = 0

DP

(2 - (1 - 1)) = 2

2 - 0

$(2 * (3 - (4 * 5))) \rightarrow$

$2 * (3 - 20)$

$2 * (-17) = -34$

$((2 * 3) - (4 * 5)) = 6 - 20 = -14$

$((2 * (3 - 4)) * 5) = -10$

$(2 * ((3 - 4) * 5)) = 2 * (-5) = -10$

$(((2 * 3) - 4) * 5) = 10$

$6 - 4$

observation

↓

Soln

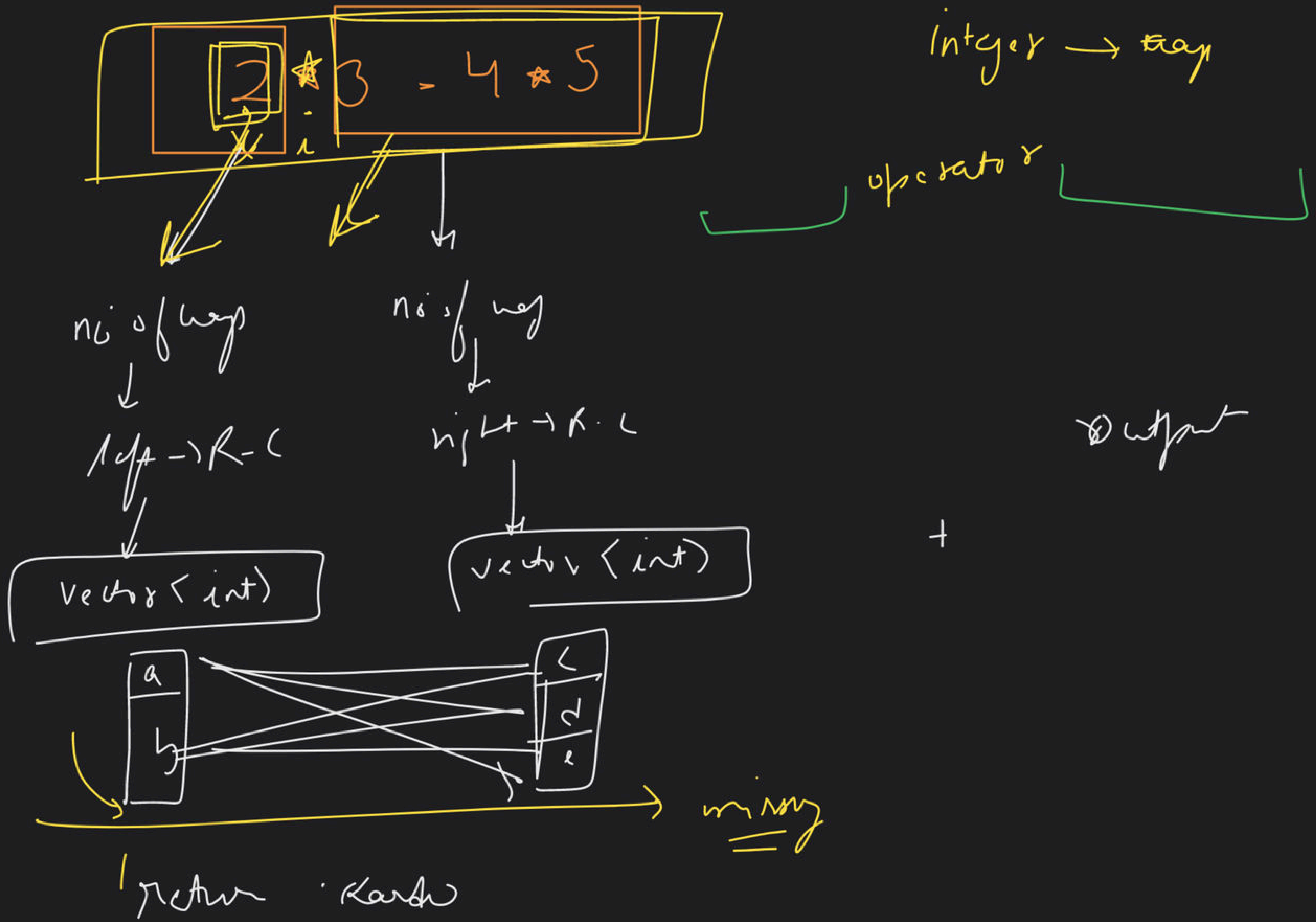
↳ Division

→ operation

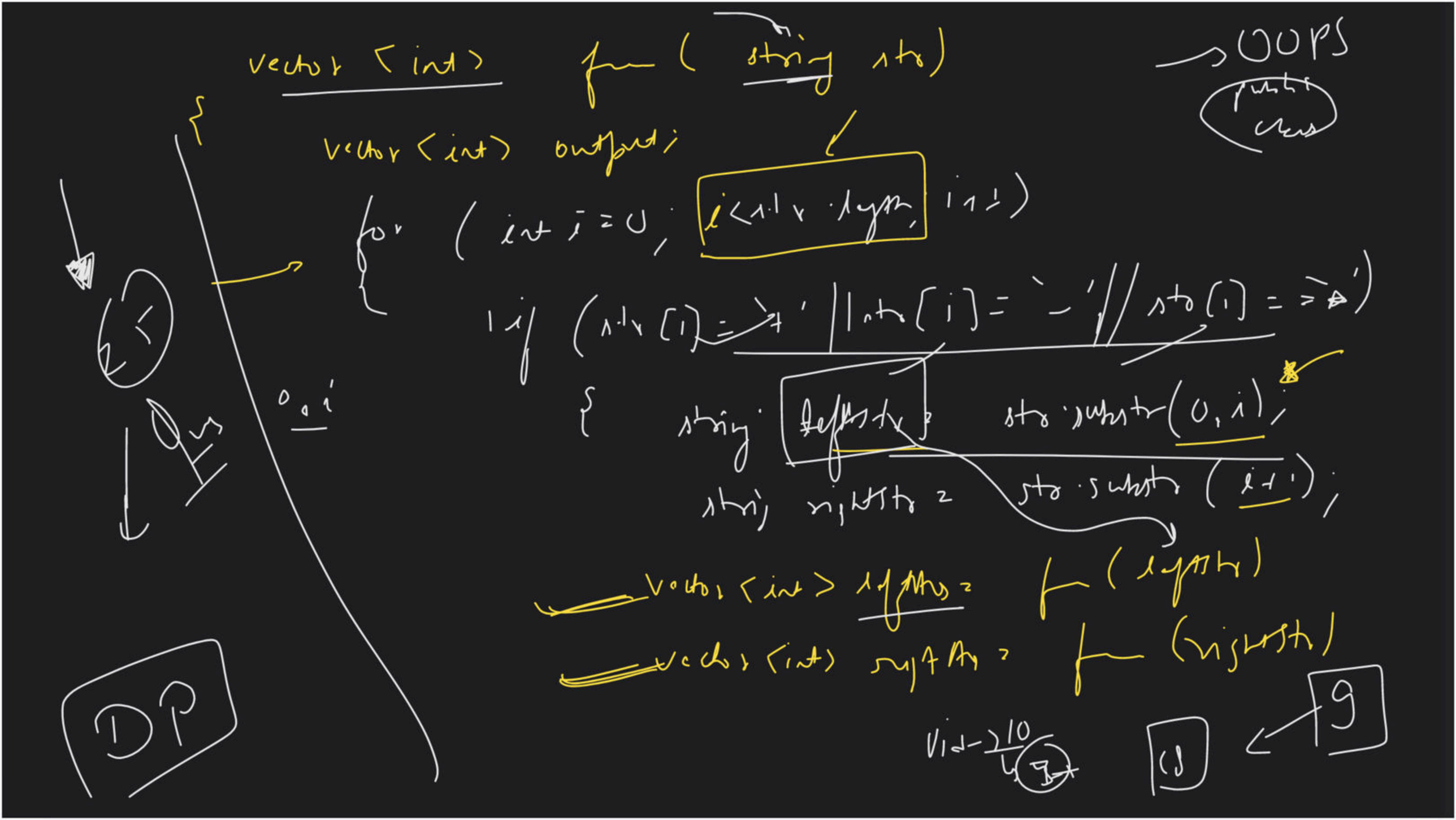
→ Brute force

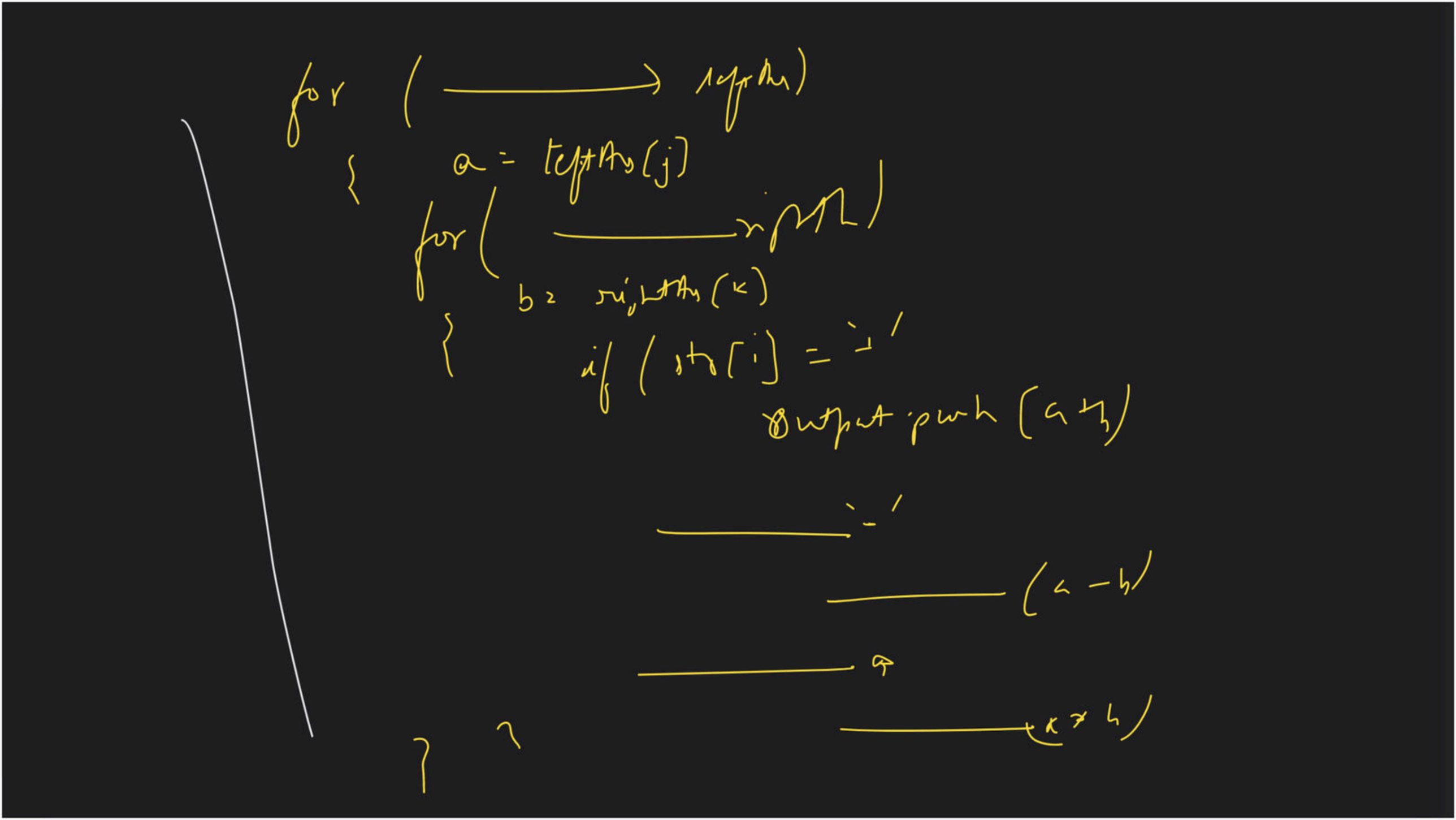
bracket

[I can solve Large
a bracket recursively
Kar dye]



2 * 3 - 4 * 5
integer → array
operator
no of ways
left → R.C
no of ways
right → R.C
vector<int>
vector<int>
a
b
c
d
e
output
missing
return


vector<int> fun(string str)
{
vector<int> output;
for(int i=0; i<str.length(); i++)
if(str[i]=='+' || str[i]=='-' || str[i]=='*')
{
string leftstr = str.substr(0,i);
string rightstr = str.substr(i+1);
vector<int> leftans = fun(leftstr)
vector<int> rightans = fun(rightstr)
0 to i
OOPS
DP
Vid-210
9
(1)



return output;

stoi ( )

30sec

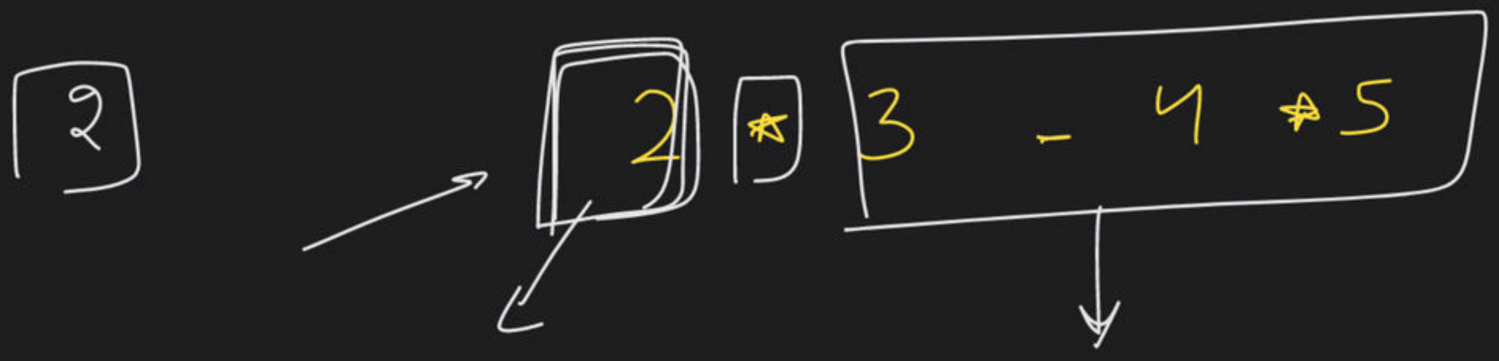
30sec

i/p → 2

output

if (output.size == 0)

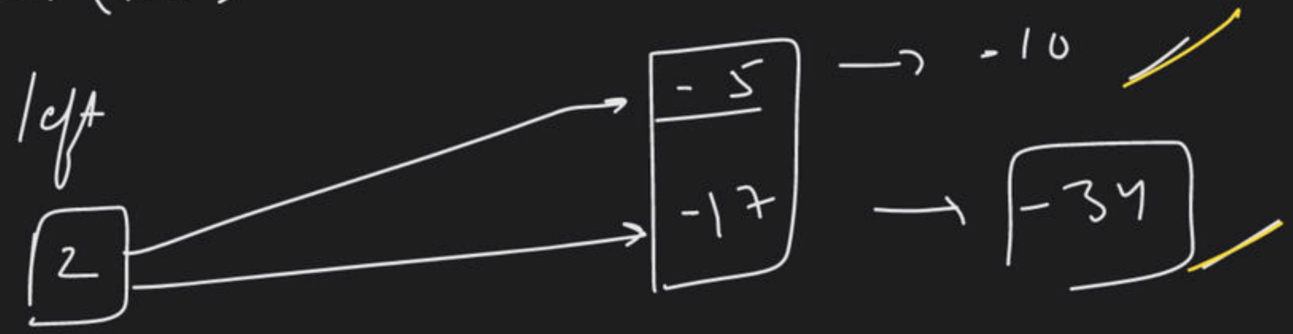
output.push_back (stoi(exp));



(3 - 4)▷5 = -5

3 - (4▷5)

3 - 20 : -17



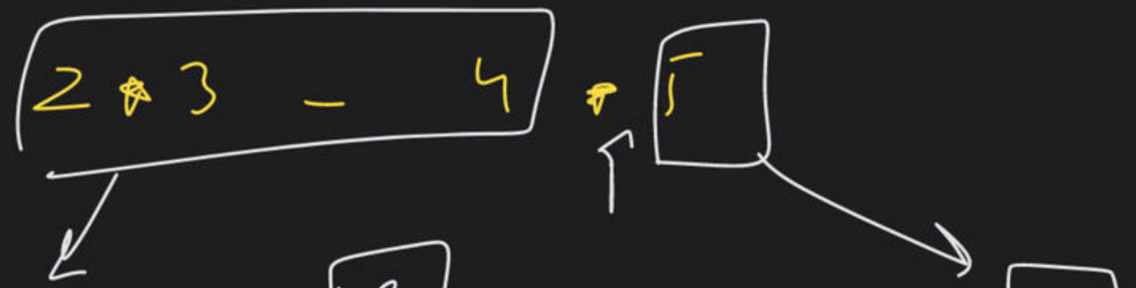
-10
-34
-17
10
-10

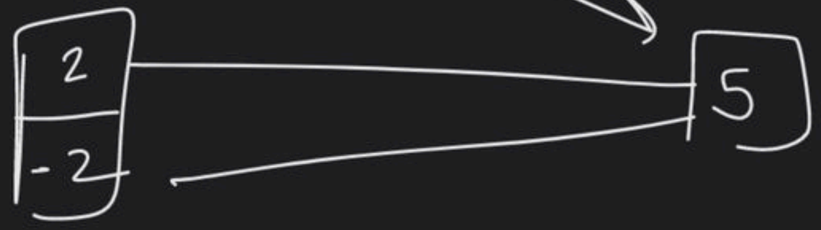
2 * 3 = 4▷5

4▷5
↓
20

2 * 3
↓
6

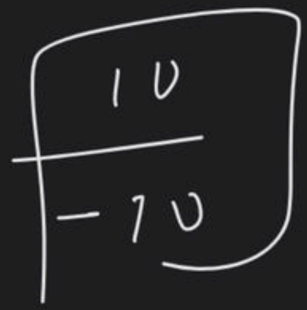
6 - 20 = -14

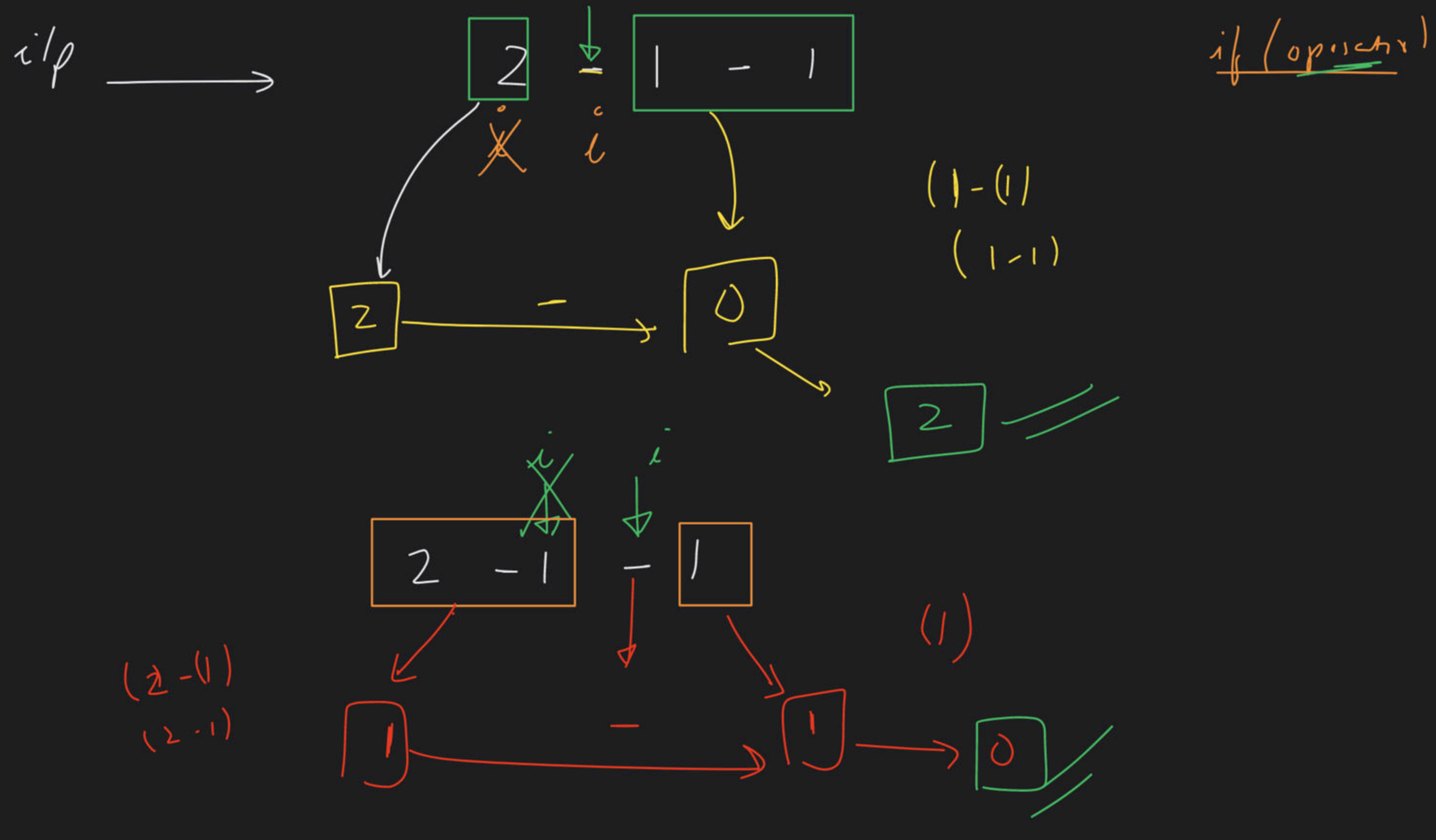


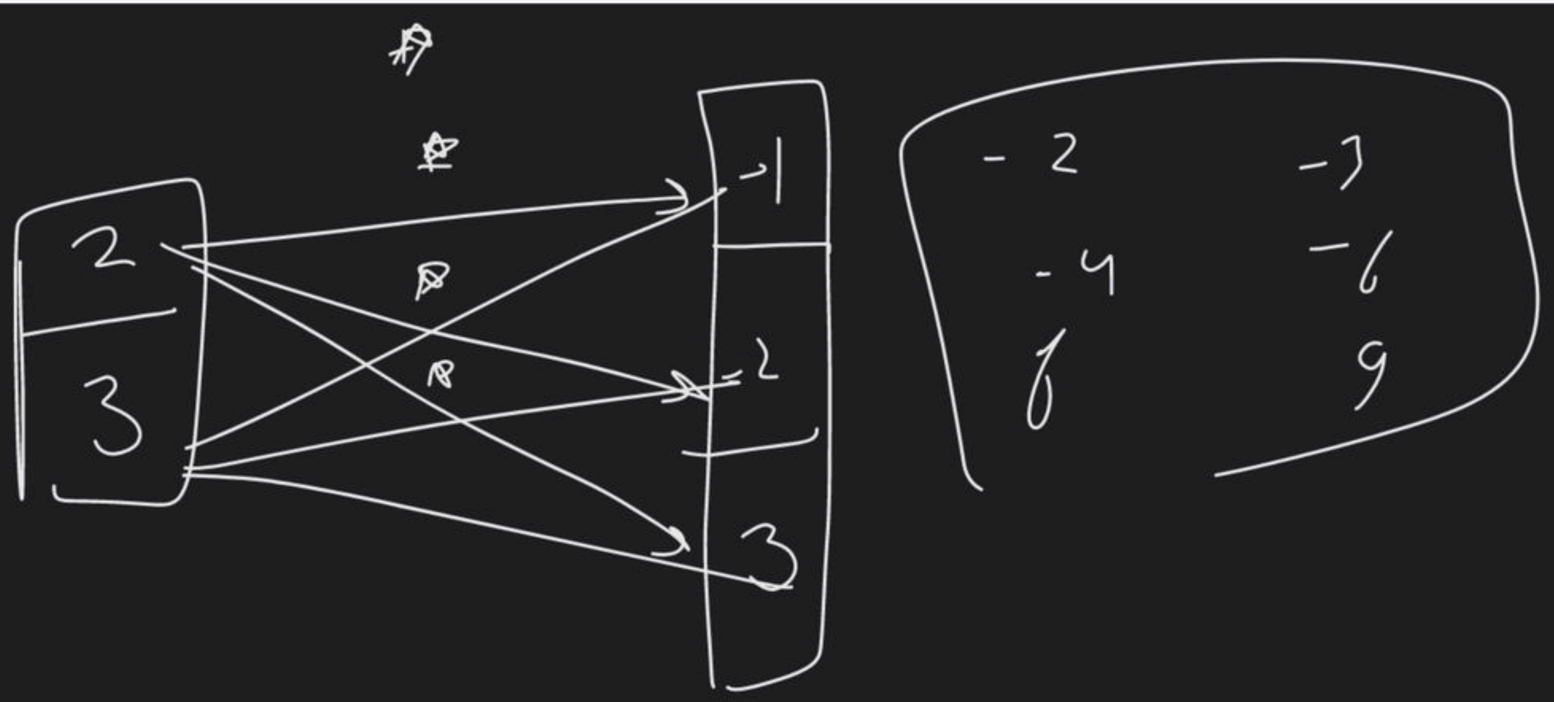


$(2 * 3) - 4 \rightarrow 2$

$2 * (3 - 4) \rightarrow -2$




i/p
2
1 - 1
if (op==chr)
(1-(1)
(1-1)
2
-
0
2
2 - 1
-
1
(2-(1)
(2-1)
1
1
0


2
3
-1
-2
-3
-4
-6
7
9

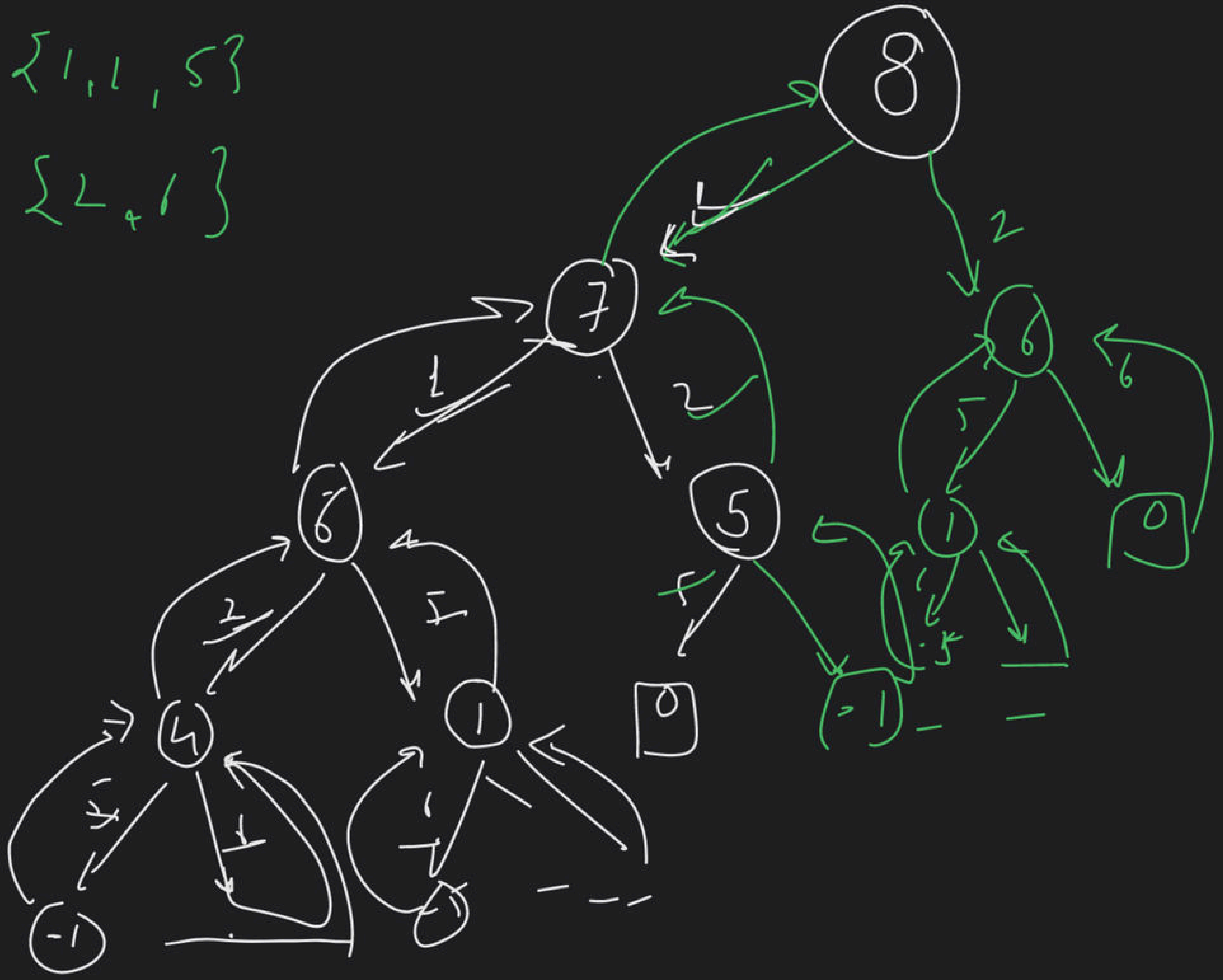
→ CS 2 :-

1 1 2 5 6 7 10

(1) → { 1, 1, 2, 5, 6, 7, 10 }

{1, 1, 5}

{2, 1}



{1, 1, 2, 5, 6, 7, 10}

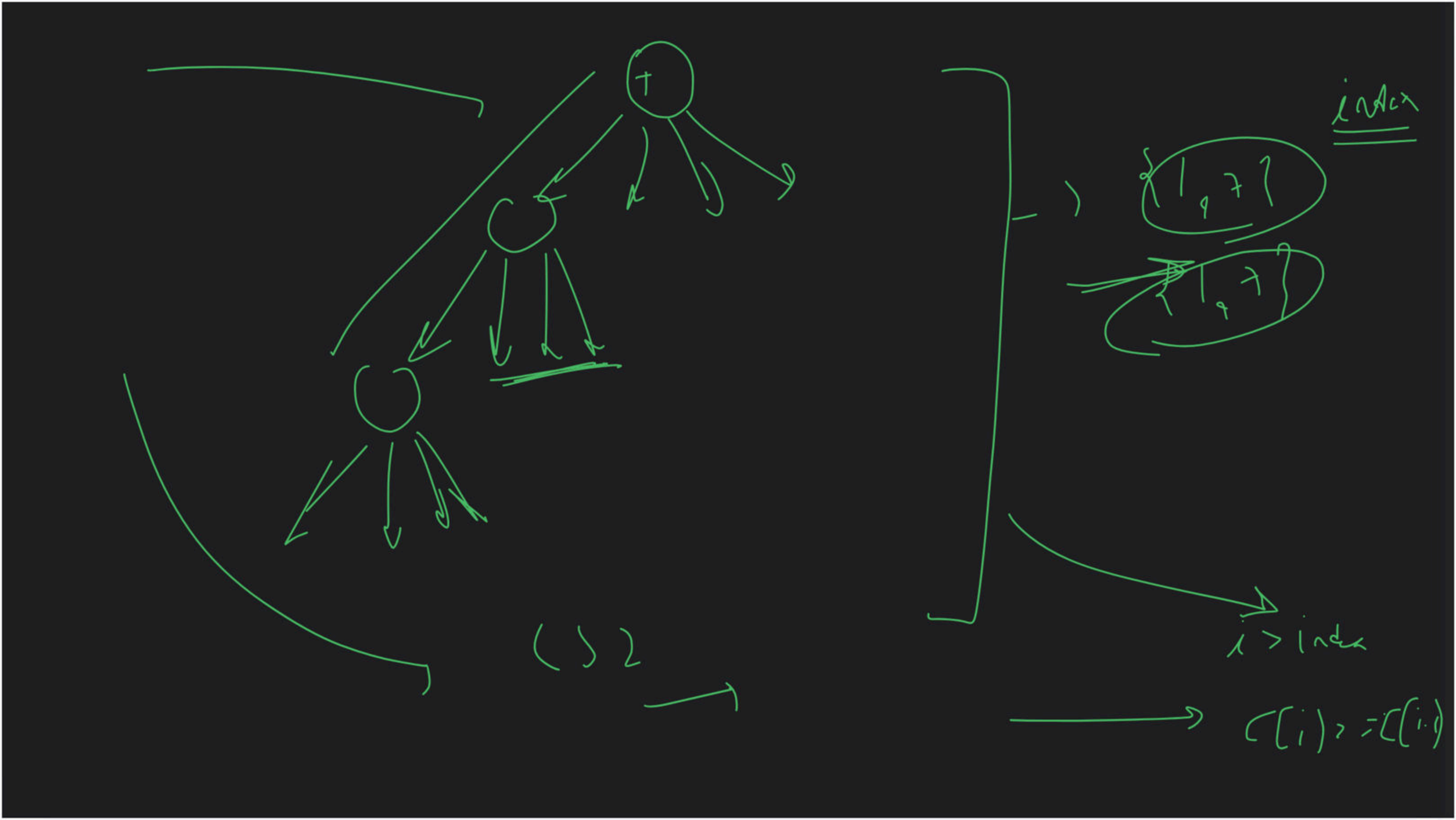
index

i > index

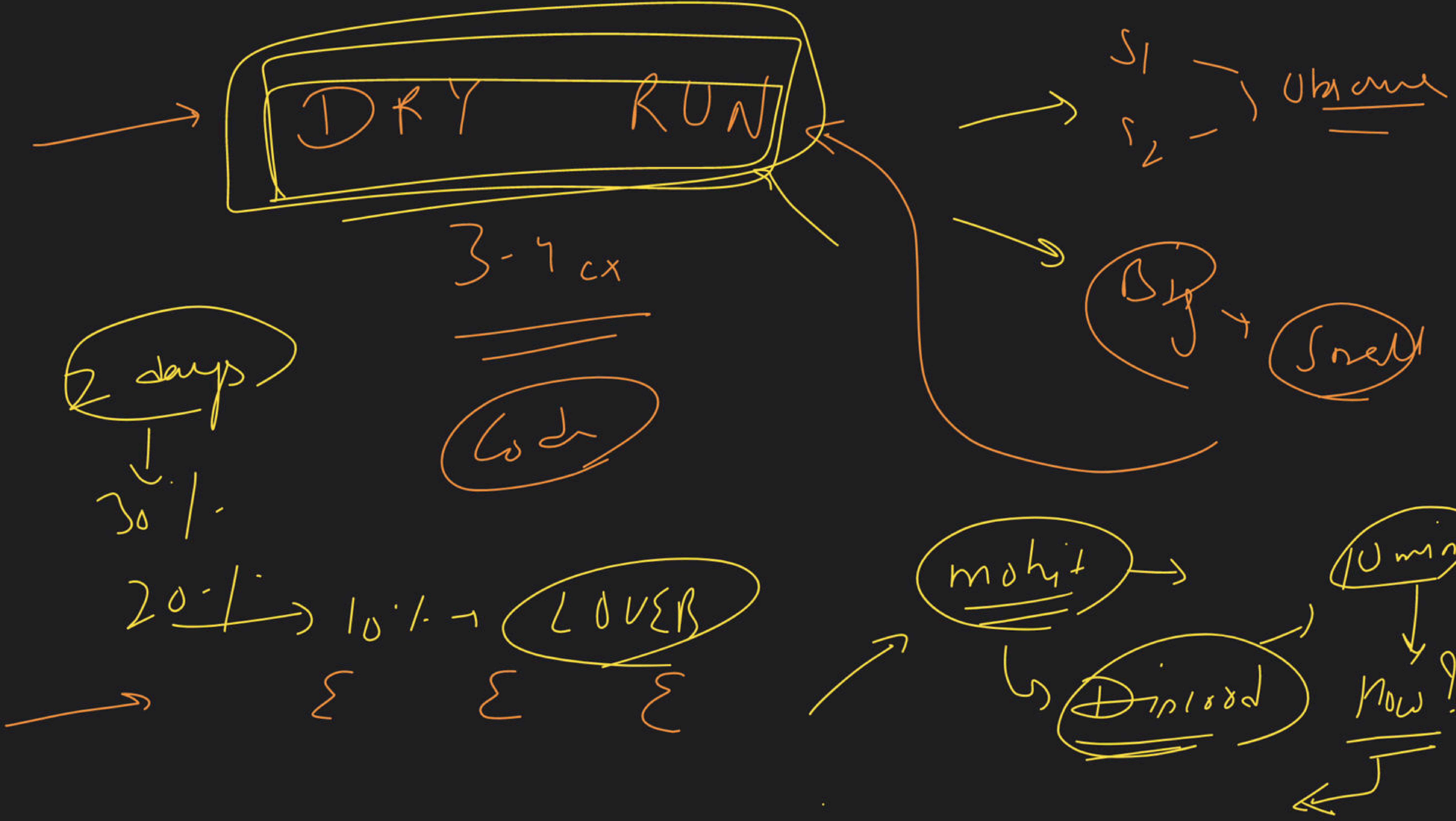
if

c[i] == c[i-1]

continue


index
i > index

((n - 1, k) + k - 1) % n + 1
How?
H/w
{2, 3, 7}
%
Yaah
circlar game
R.C & %
n K
n-1, k-1


DRY RUN
3-4 cx
S1
S2
Big
Small
2 days
Code
30%
20%
10%
mohit
10 min
How?