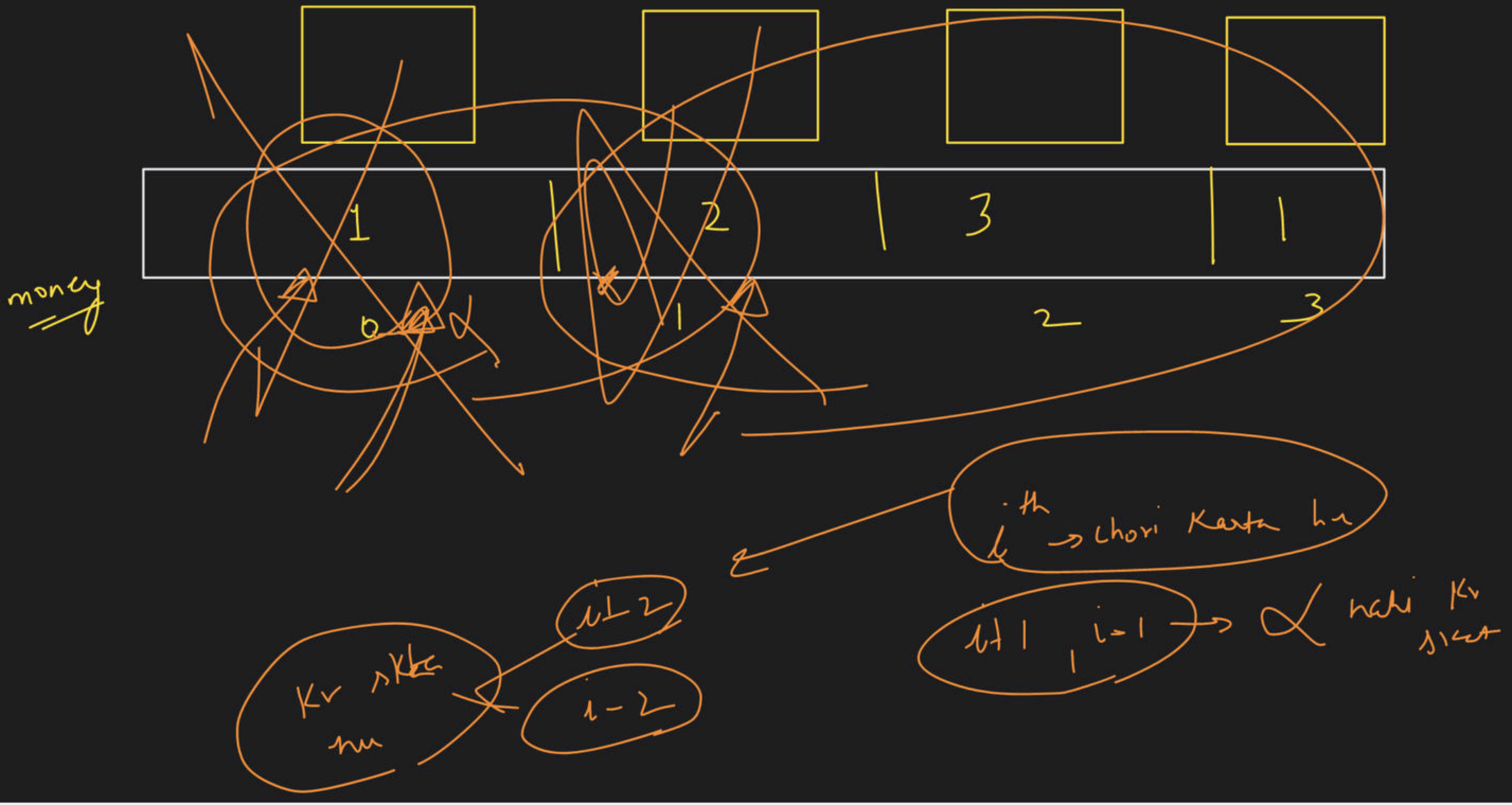
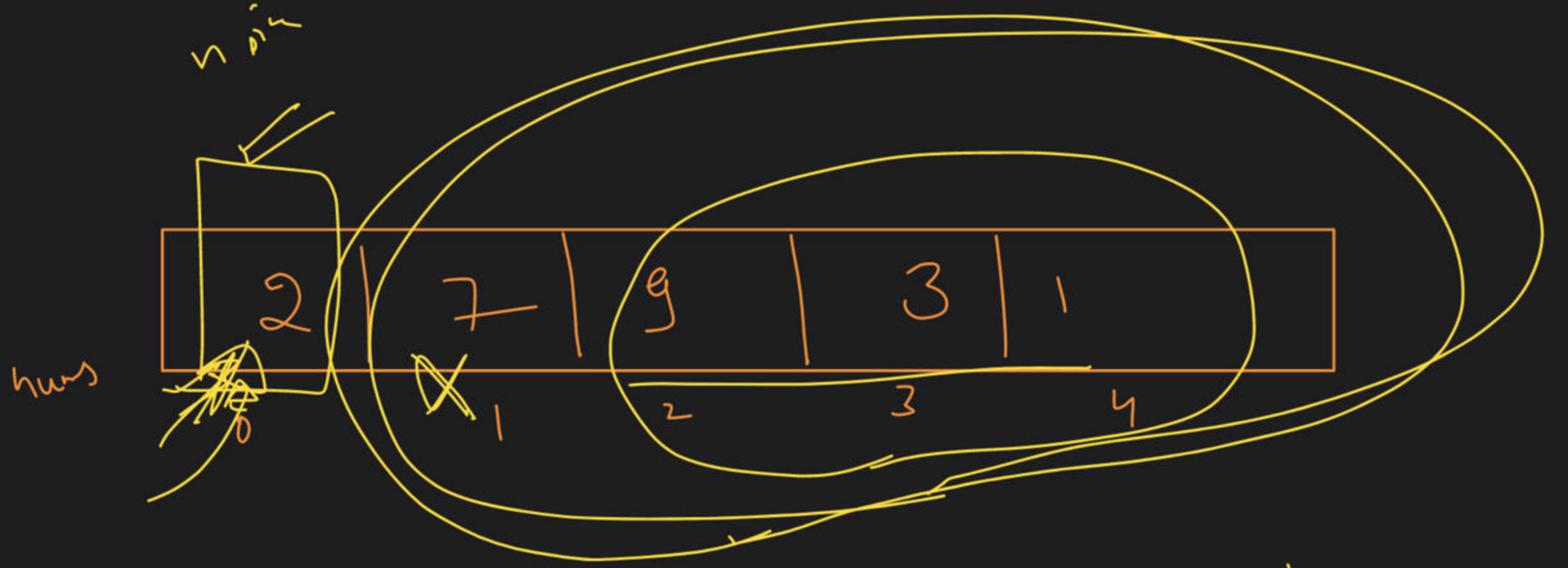
Dynamic Programming Class-2

Special class

Love Babbar • Jan 8, 2024


money
1
2
3
1
0
1
2
3
ith
chori Karta hu
nahi Kr
sakta



case → include → $2 + f(n-2)$ ⟶ ans1

case → exclude → $0 + f(n-1)$ ⟶ ans2

ans1, ans2 → max → final Ans

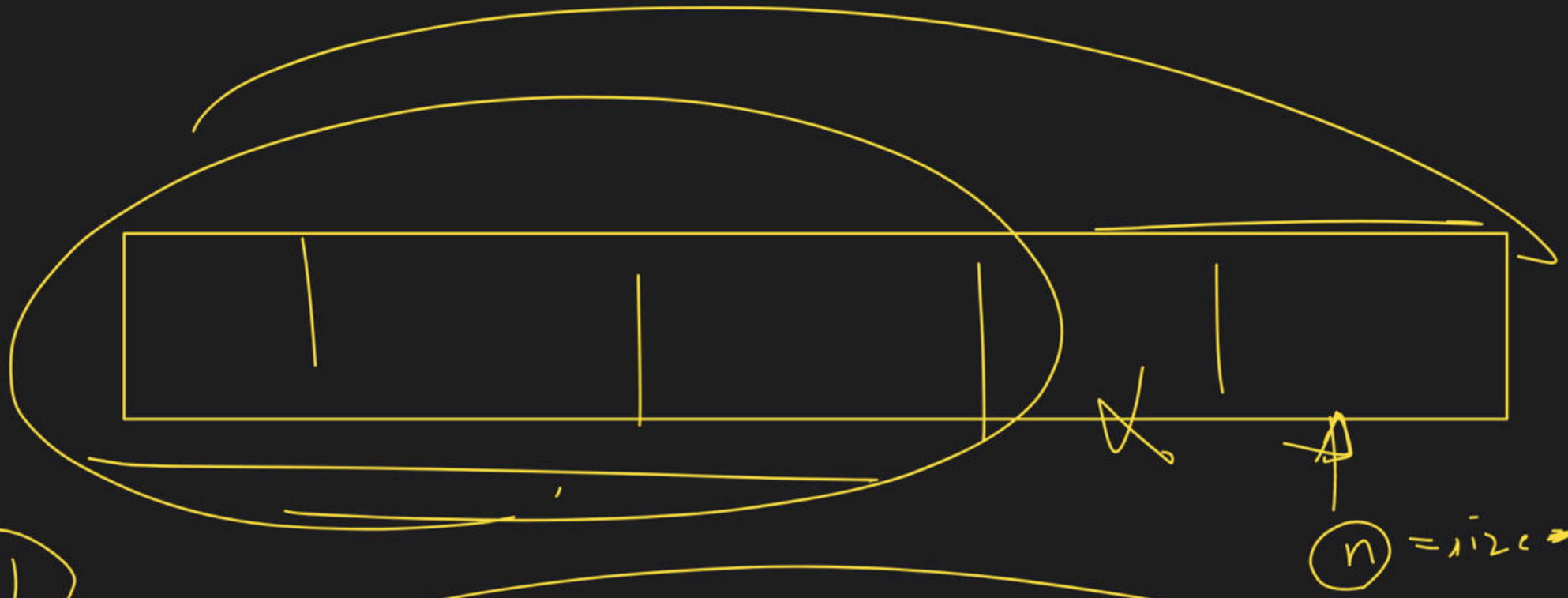
i=0

i+1

i+2

L → R

```
int included = nums[i] + solve(i+2)
int excluded = 0 + solve(i+1)
```

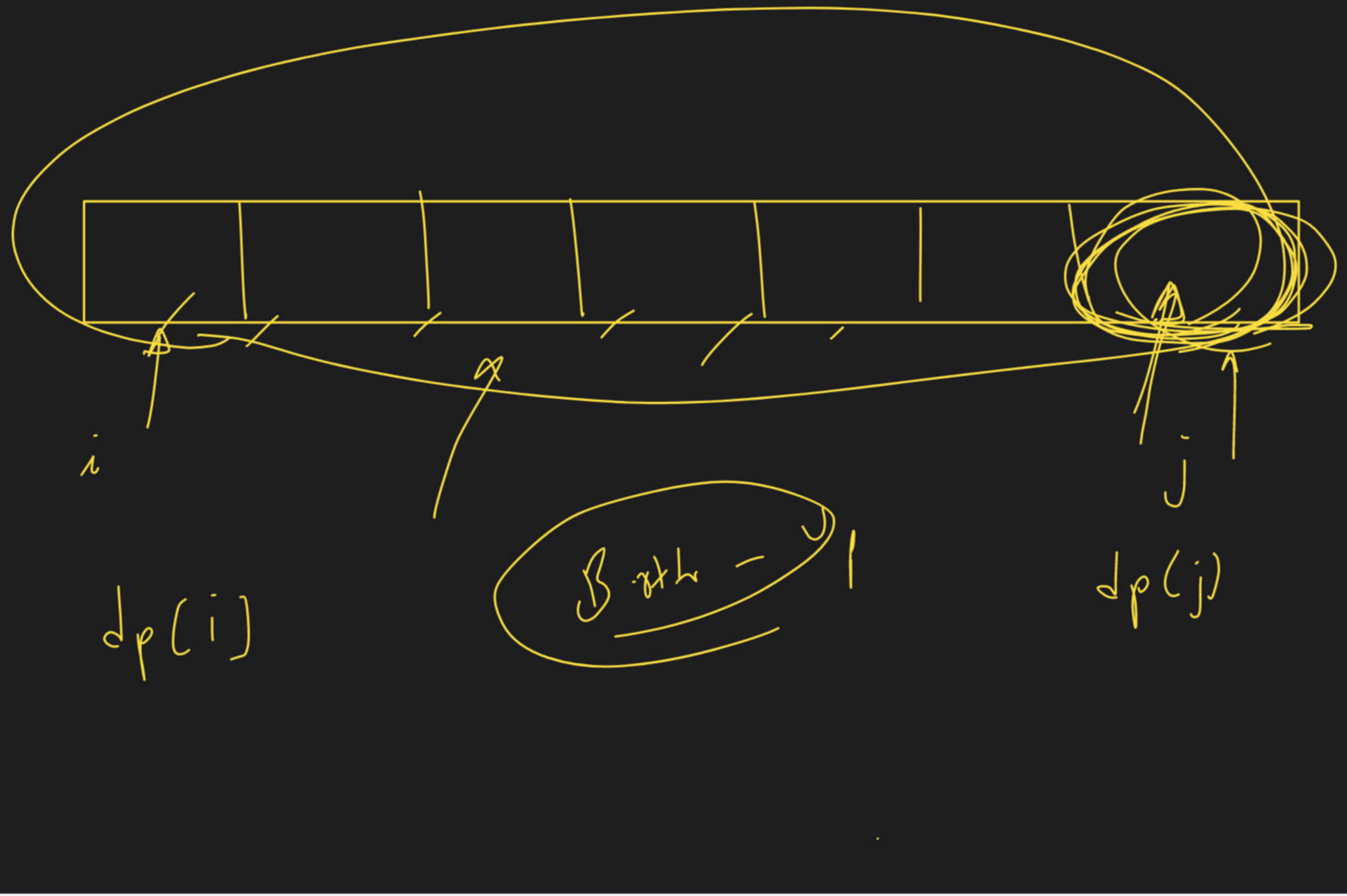


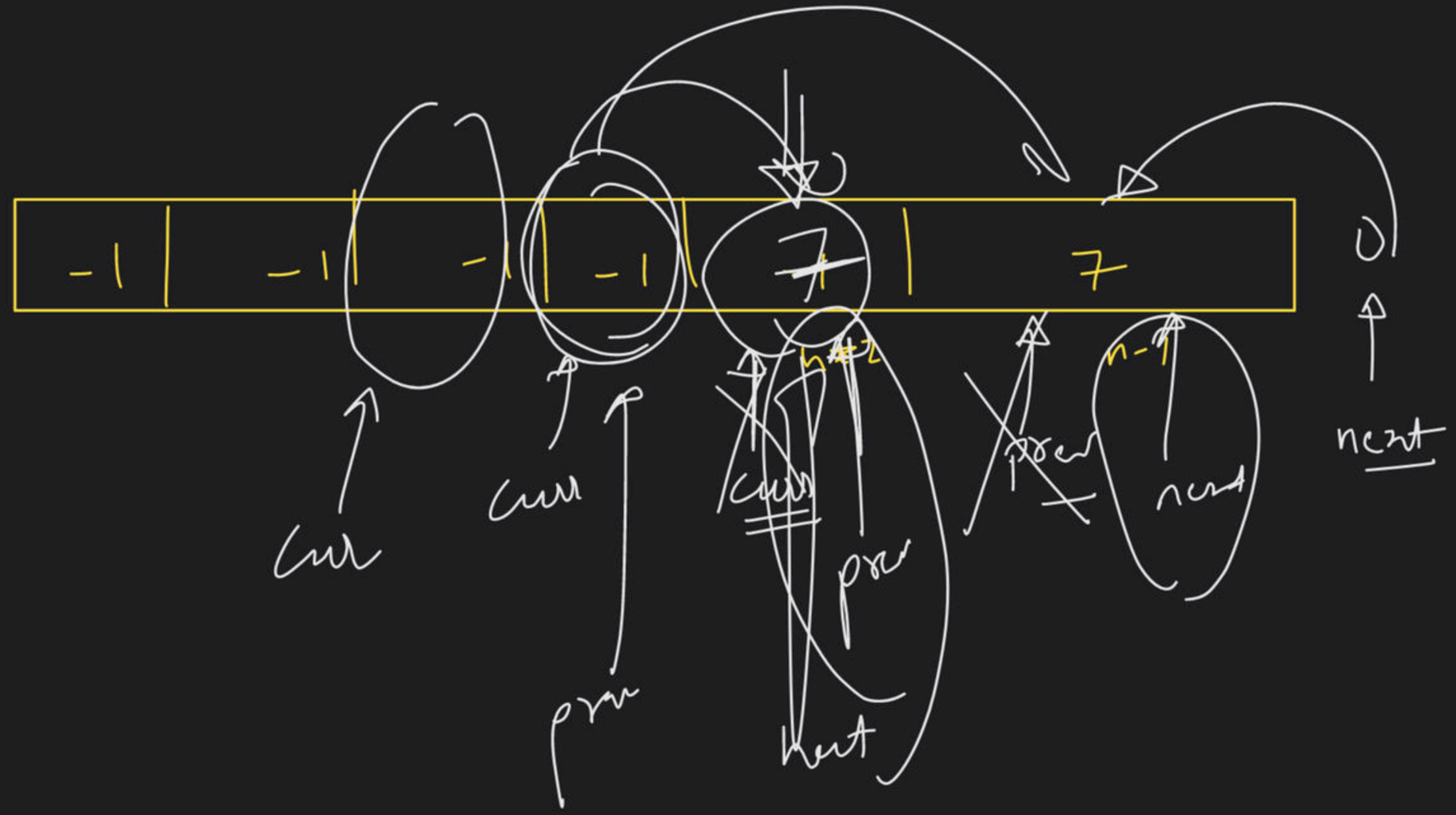
R -> L

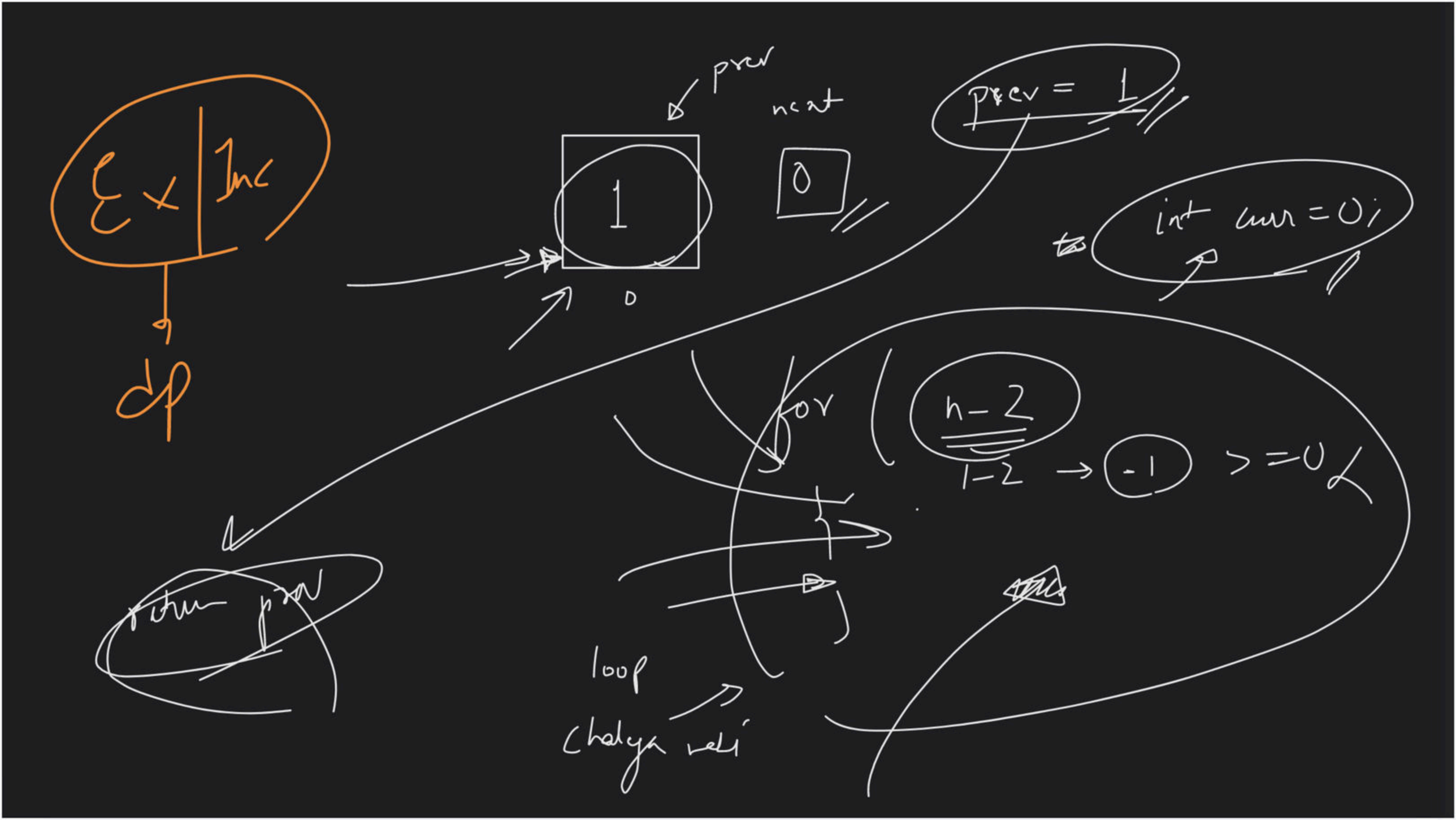
L -> R

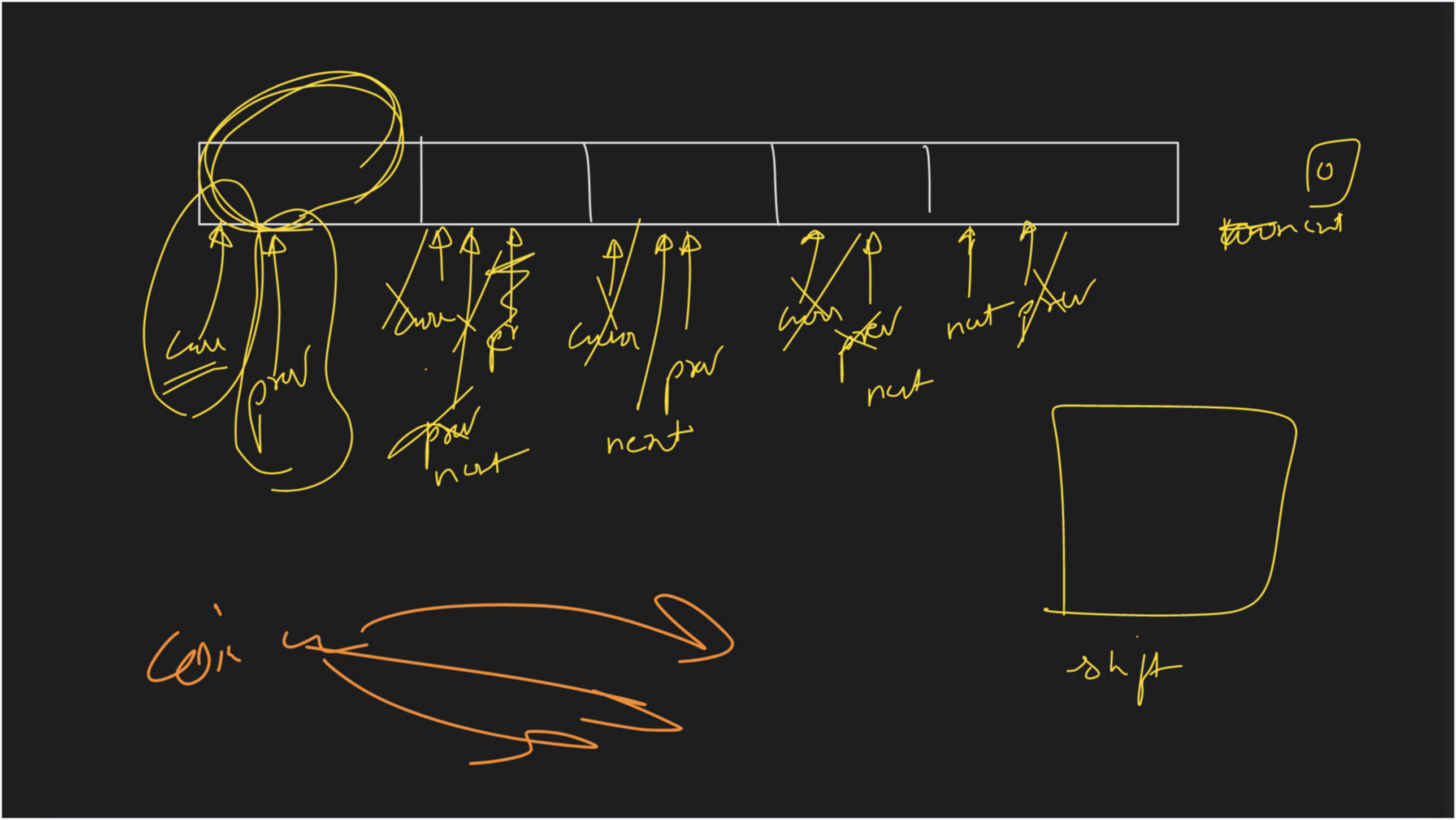
include = nums(n) + solve(n-2)

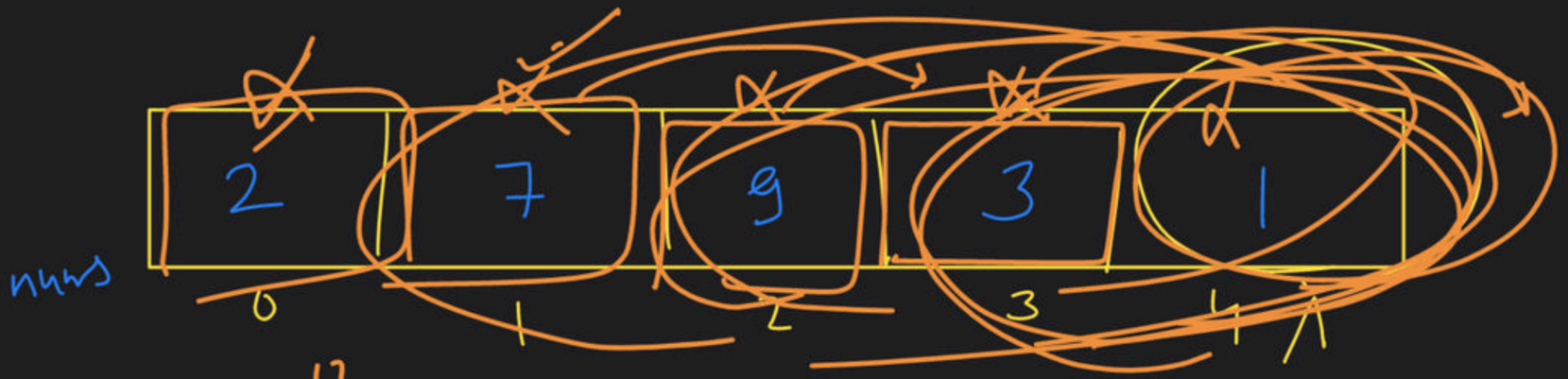
exclude = 0 + solve(n-1)


i
j
dp(i)
dp(j)


-1
-1
-1
-1
7
7
curr
curr
prev
curr
prev
next
prev
n-1
curr
next

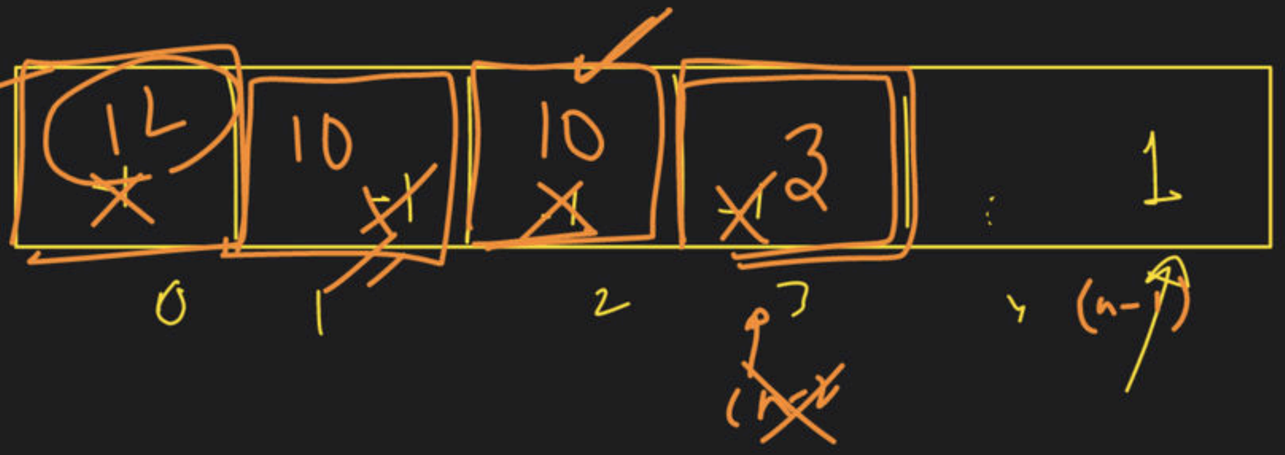

Ex | Inc
dp
prev
next
1
0
0
prev = 1
int curr = 0;
for (n-2
1-2 → -1 >=0
return prev
loop
Chalya rahi


curr
prev
curr
prev
next
curr
prev
next
curr
prev
next
prev
next
0
shift



ind → 10 –
ex → 10/

ind = 2 + 10 = 12
ex = 10

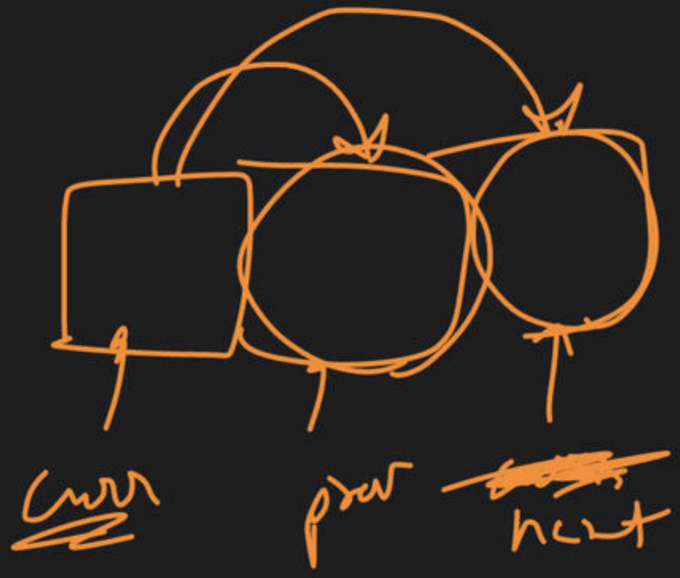


5 = 0

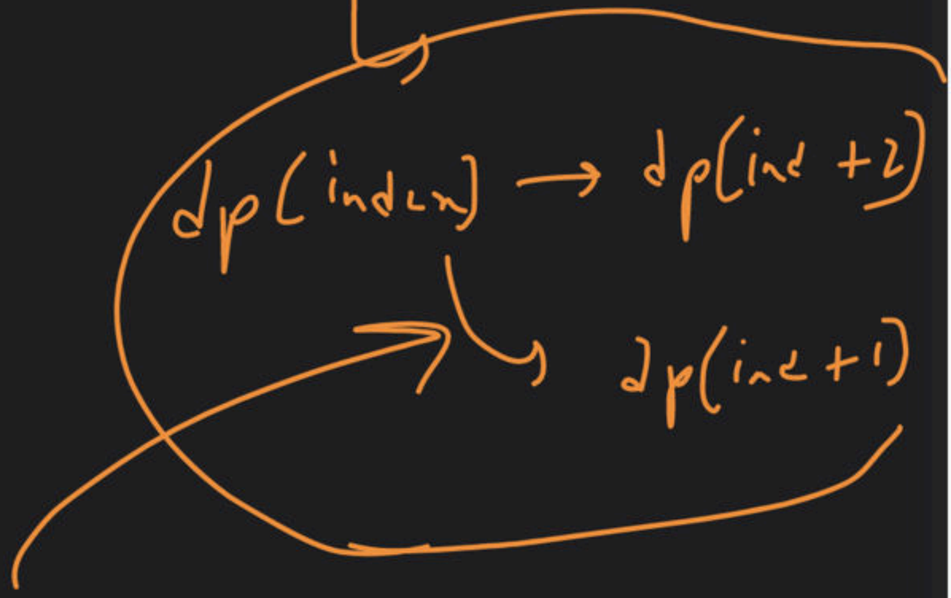
ind = 10
ex - 3

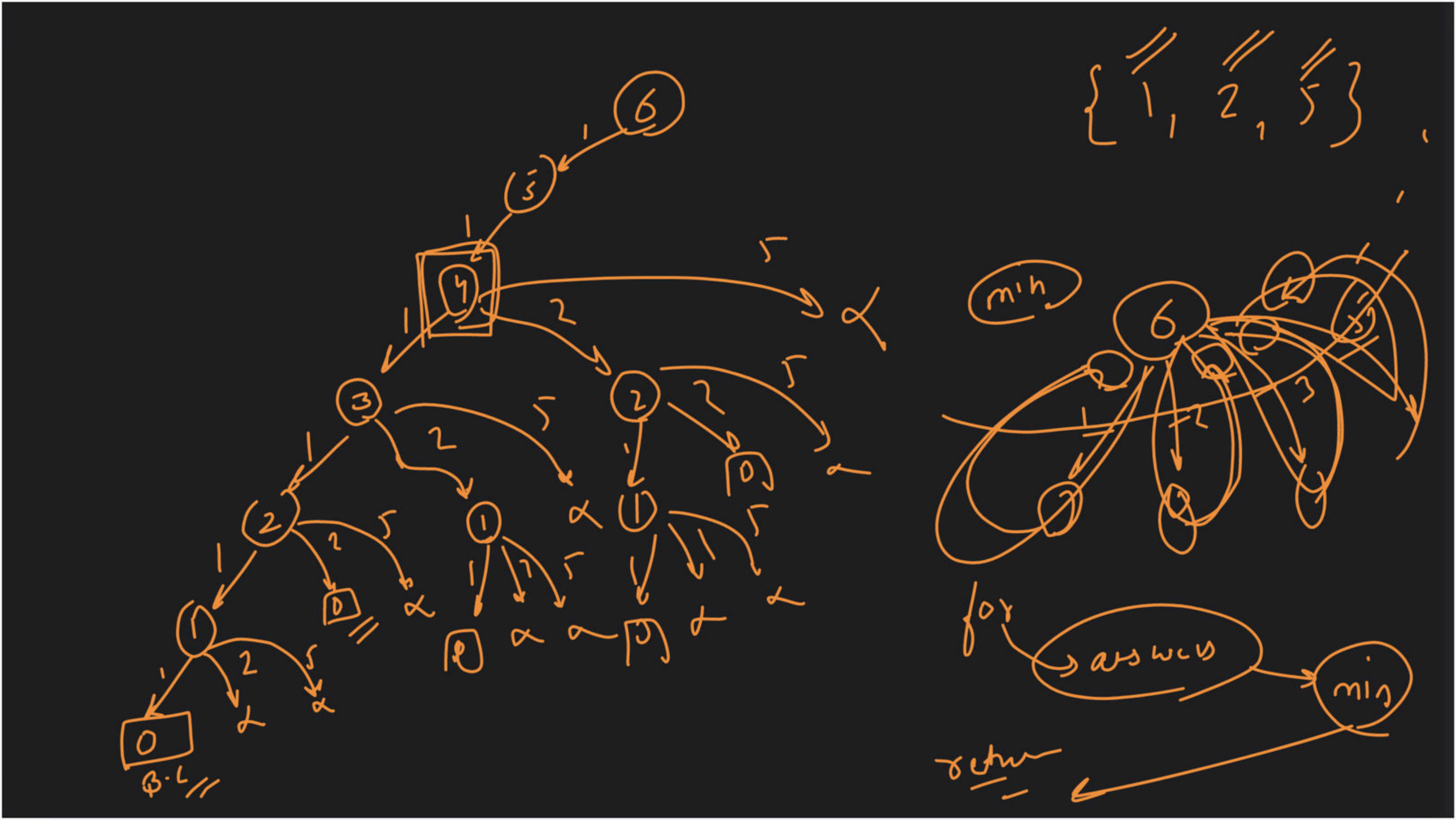
for (n - 2 ⟶ 0

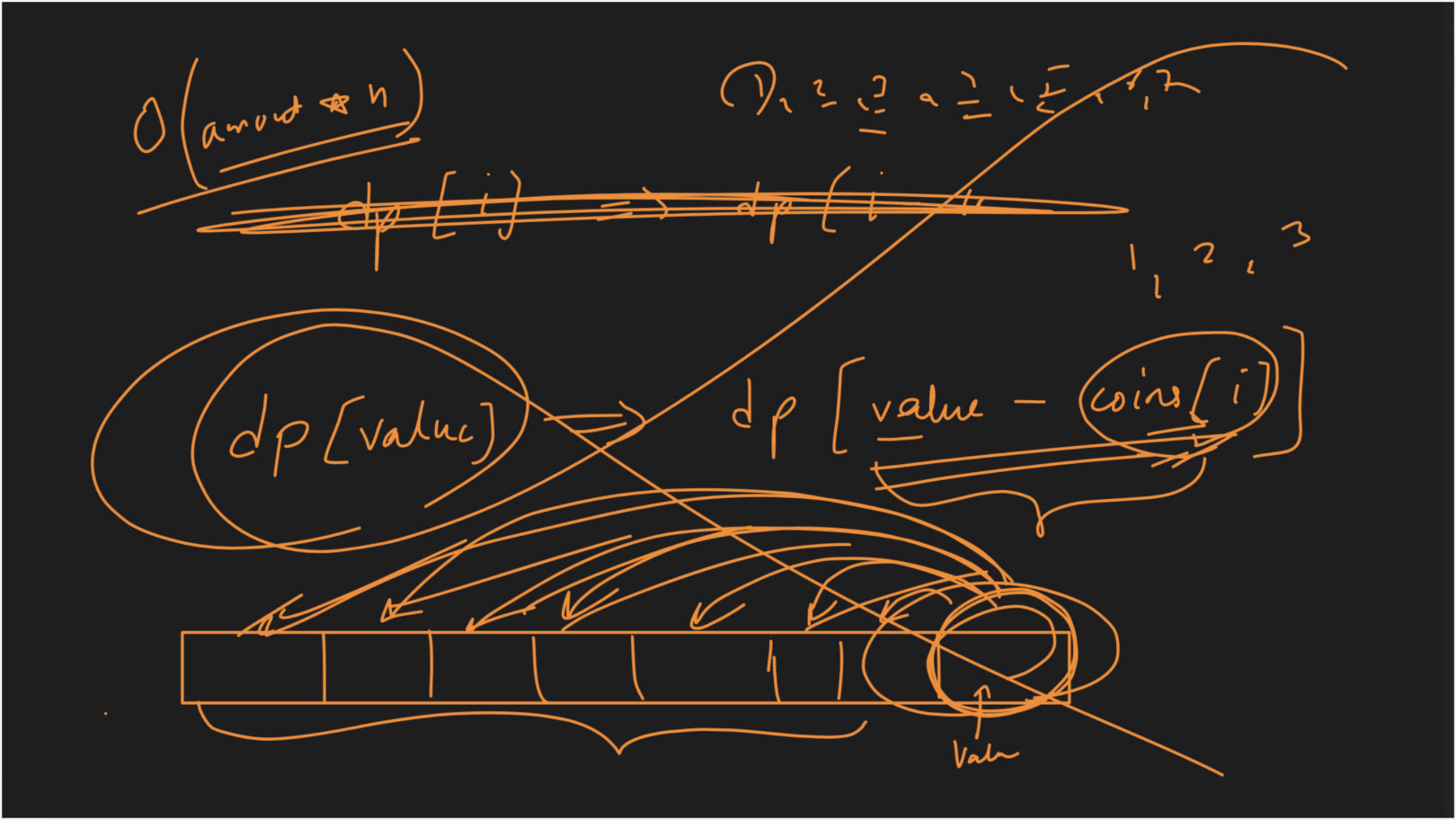
include = 3 + 0
= 3
ex → 2

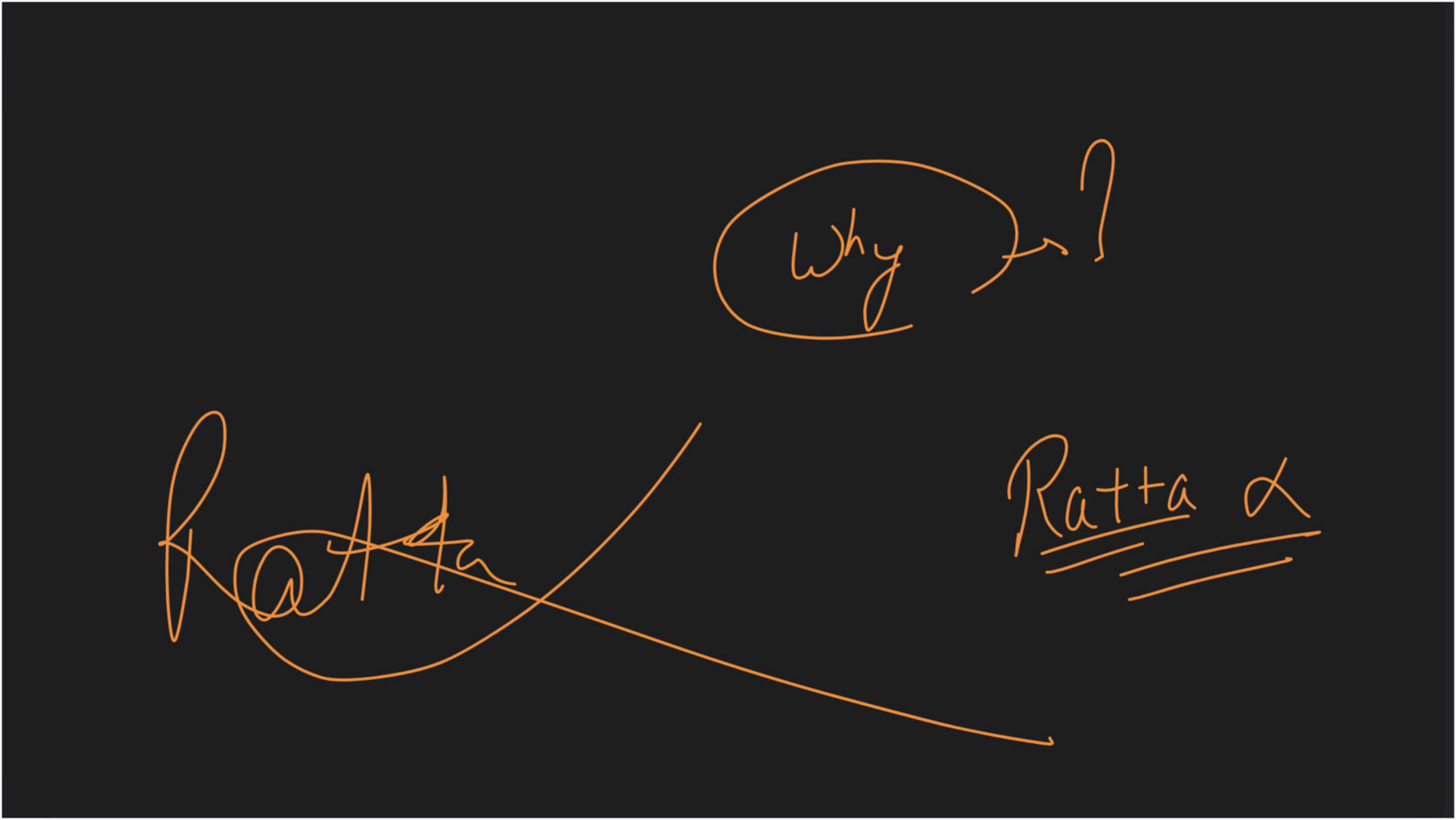


Code → Tabulation




{1, 2, 5}
min
for
answers
min
return


O(amount * n)
dp[i] => dp[i
1, 2, 3
dp[value]
dp[value - coins[i]]
Value


Why ?
Ratta