Q:) flowers → K ~~the~~ the group occurrence for white

There is constraint of grouping on white but not on red

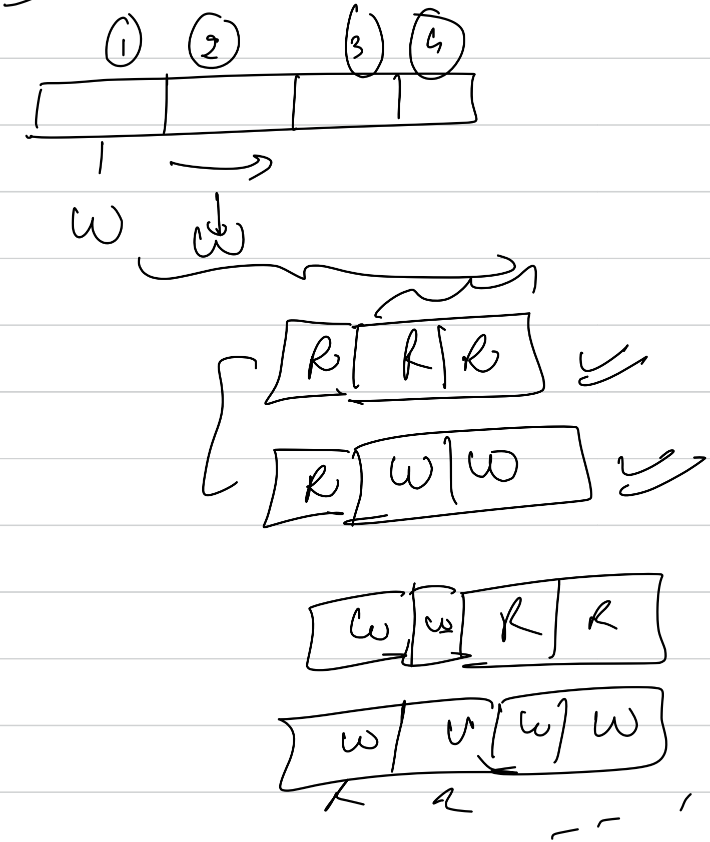
K = 2

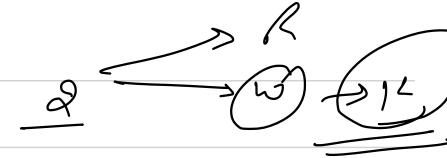
$\boxed{\text{4 flowers}}$

RRRR — 1
WWRR — 1
RWWR — 1
WWWW — 1
RRWW — 1

5

Is DP ensured ??





1 query → $O(1)$ or $(\log_2 n)$

How can we effecently calc no. of ways for any value i → i flowers will be calc'n

$\boxed{i=1}$ → K

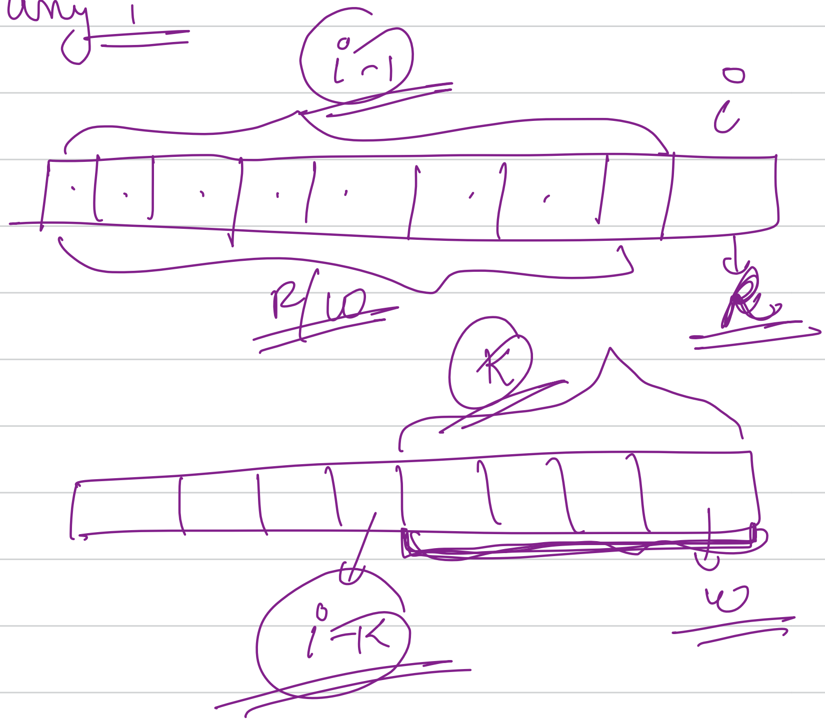
$K==1$ → 2

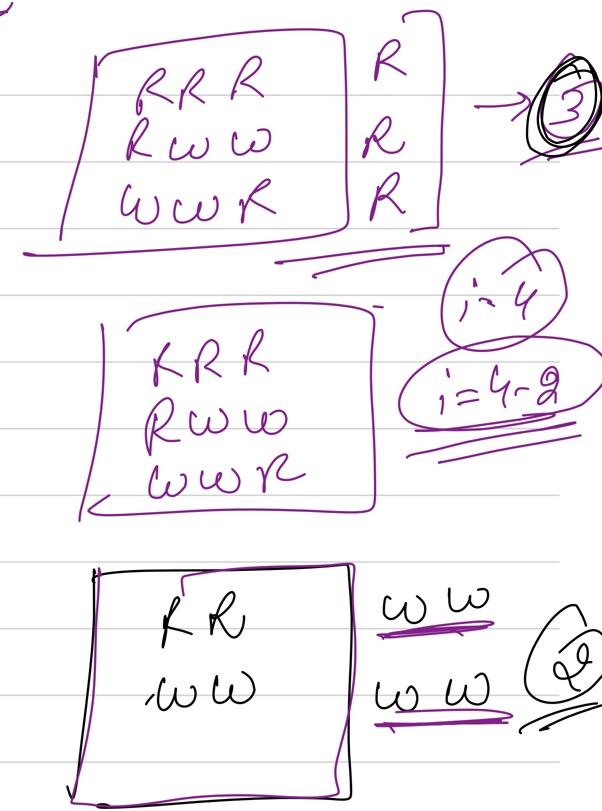
$K>1$

$\boxed{i=0}$ → one valid way → 1

for any i

i-K





$$\Rightarrow f(i, k) = \begin{cases} f(i-1, k) & \text{if } i < k \\ f(i-1, k) + f(i-k, k) & \text{if } i \geq k \end{cases}$$

fun^n which count the no of ways to eate i food dp array → precompute for all value of i.

↳ prefix sum

[a, b]

1	2	1	3	1	2	3

p →

1	3	4	7	8	10	13

$$\boxed{8 - 3 + 2}$$

$8 - 1 \Rightarrow 7$

$p[b] - p[a] + arr[a]$

$p[b] - p[a-1]$

i R

K-white flame

K=2

R R R R
R W W R
W W W R R

3

i=4

i=3

i=4

4-2 = 2

i=5

R R
W W

W W
W W

2

$t \le 10^5$

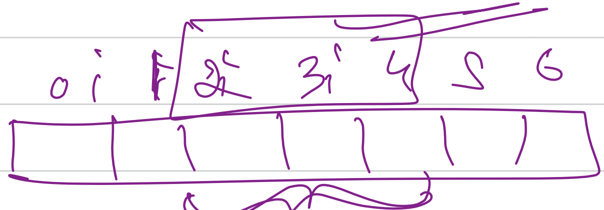
TLE

$n \le 10^5$

$O(n)$

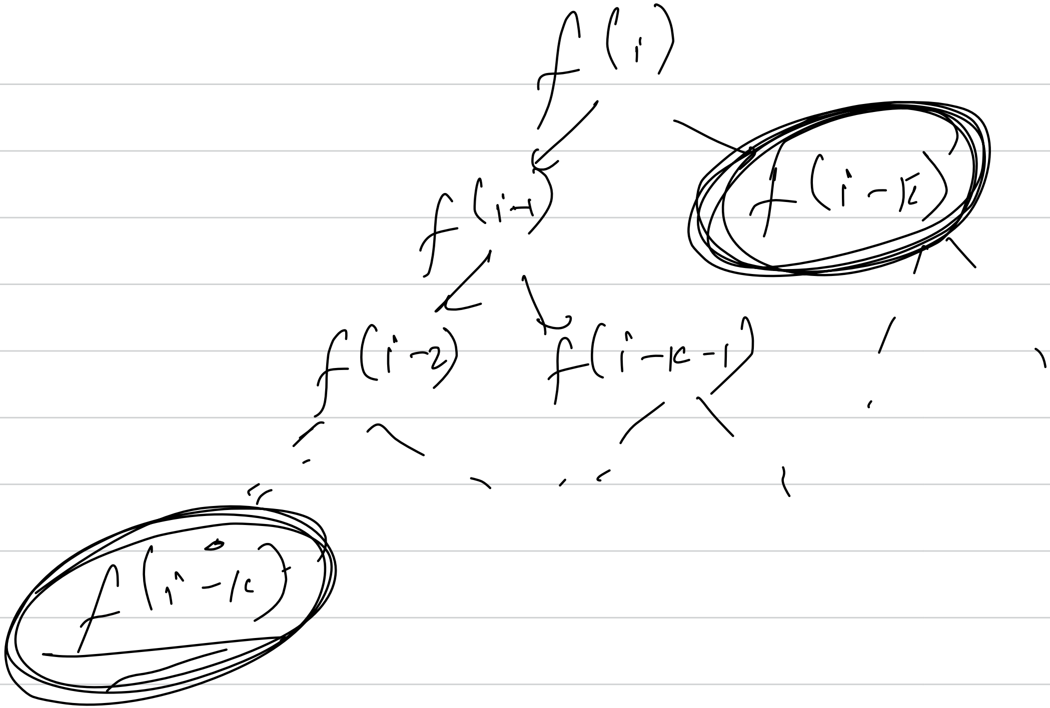
5

0 1 2 3 4 5 6



dp

prefix


f(i)
f(i-1)
f(i-k)
f(i-2)
f(i-k-1)
f(i-k)

$a_1 \quad a_2 \quad a_3 \quad - - - - - - \quad a_k$

$\rightarrow \quad p_1 \quad p_2 \quad p_3 \quad - - - - - \quad p_k$

$$\boxed{p_k = \sum_{i=0}^{k} a[i]} \leftarrow \text{prefix sum}$$

$$\text{Sum}(a, b) = \underline{\underline{p[b] - p[a-1]}}$$

0 1 2 3 4 5 6

0	0	0	0	0	0	0

+1

1

2

L, R

+1

0, 2

1, 4

0, 1

4, 5

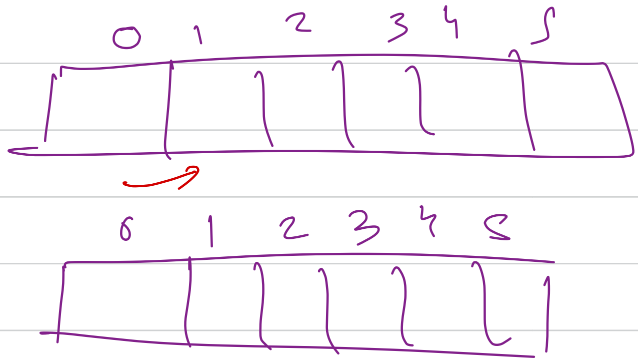
L, R → i → [L, R]

∀i → how many times 1 was added

order → 0 → only the ques starting from 0

order → 1 → all ques starting from 0 + starting from 1 − ending at 0

O(n) start



end

l, R

start[l]++

end[R]++

on

O(n)

idx → 0 → S_0

idx → 1 → $\boxed{S_0 + S_1 - e_0}$

idx → 2 → $S_0 + S_1 + S_2 - e_0 - e_1$

$\boxed{S_0 + S_1 - e_0} + S_2 - e_1$

idx → 3 → $S_0 + S_1 + S_2 + S_3 - e_2 - e_1 - e_0$

result of range addition → | 1 | 2 | 1 | 3 | 3 | 1 | 2 | ← dp[i]

↓

coins in the i^{th} box

suffix sum

X

[X, X+1, X+2 .....]

atleast X coins

coins | | 3 | 2 | 2 | | |

suffix sum

↳ stores that at the i^{th} index how many boxes have exactly i coins

q → X → coins[X]

Difference array

2, R

| 1 | 1 | 0 | 0 | -1 | -1 | 0 | 0 |

prefix sum

1, 2, 2, 2, 1, 0, 0, 0

1, 3

0, 4

2, 2

1, 3

+1

arr[l] += 1

arr[r+1] -= 1

Q

houses

1	2	3	2	1	4

→ houses → gold

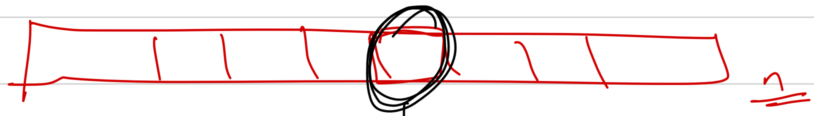
Robber → cannot rob 2 consecutive houses

Robber → max amount

1, 2, 3, 1)

Q

i



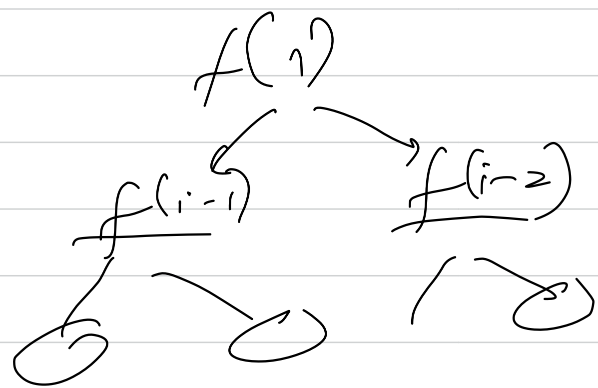
whether the robber should rob i^{th} house or not

if he is robbing i^{th} → $(i-1)^{th}$ house can't be robbed

→ **DP**

$$f(i) = \max\left(\underbrace{a[i] + f(i-2)}_{\text{when you're robbing the } i^{th} \text{ house}},\ f(i-1)\right)$$

$$\underbrace{f(i)}_{\boxed{\text{max profit till the } i^{th} \text{ house}}}$$



$dp[0] = a[0]$

$dp[1] = \underline{\underline{\max(a[0], a[1])}}$

$dp[i] = \max(a[i] + dp[i-2], \underline{\underline{dp[i-1]}})$

$\underline{\underline{dp[n]}}$