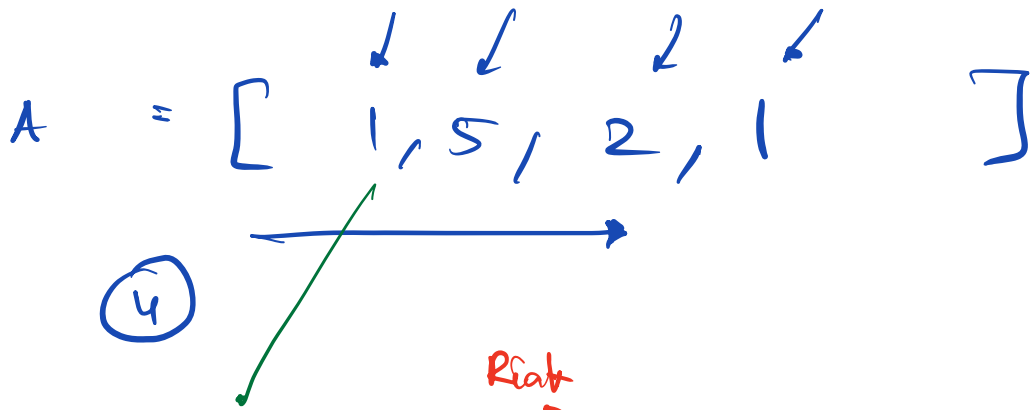
"Hello Everyone!"

① Distribute Candies

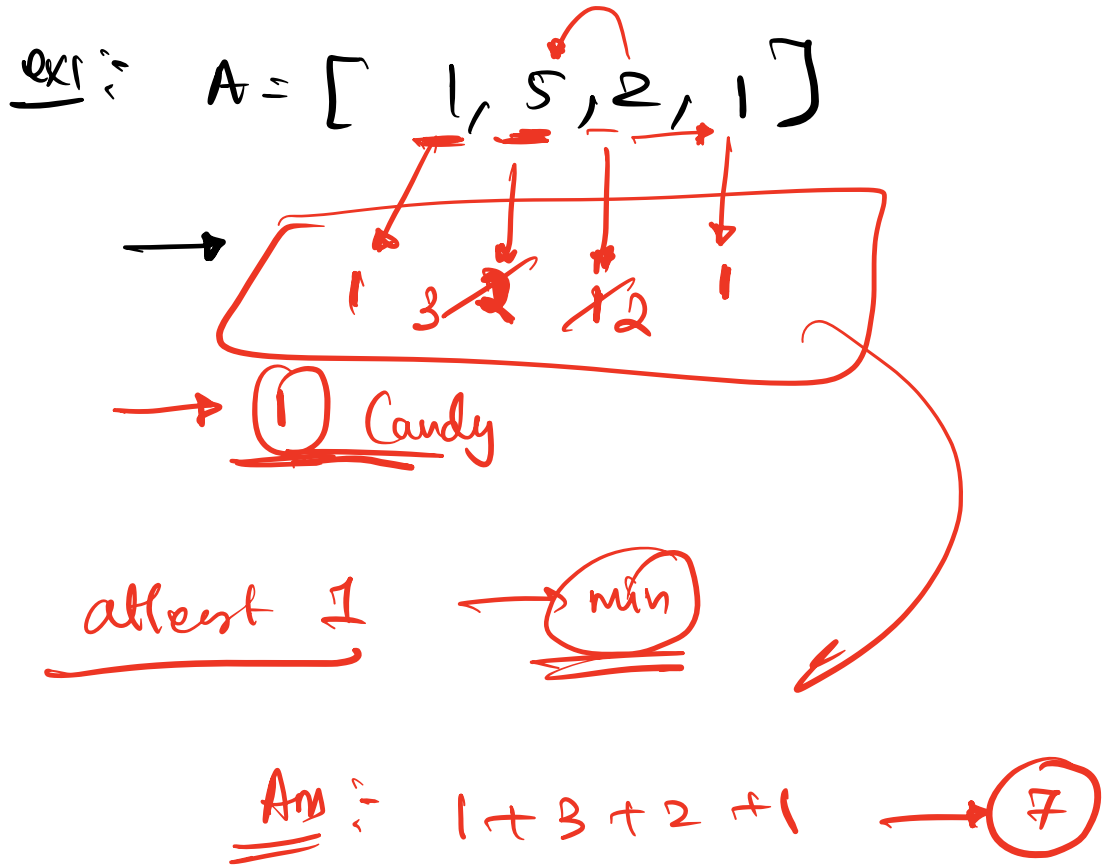


A(i) > A(i-1)

↳ c(i) > c(i-1)

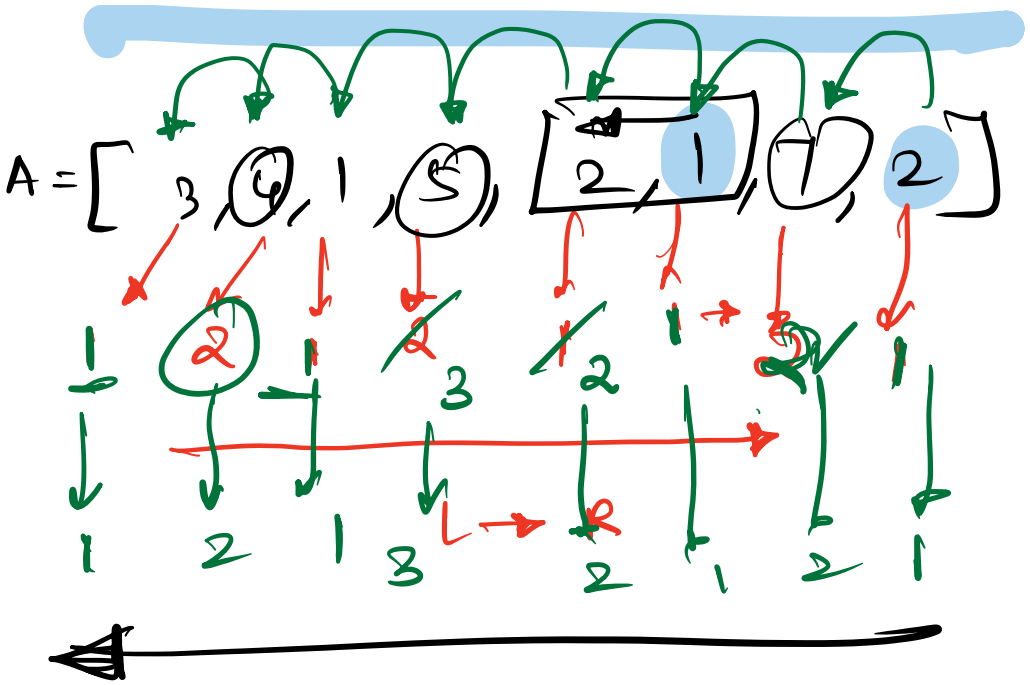
→ Min no. of candies

↳ sum



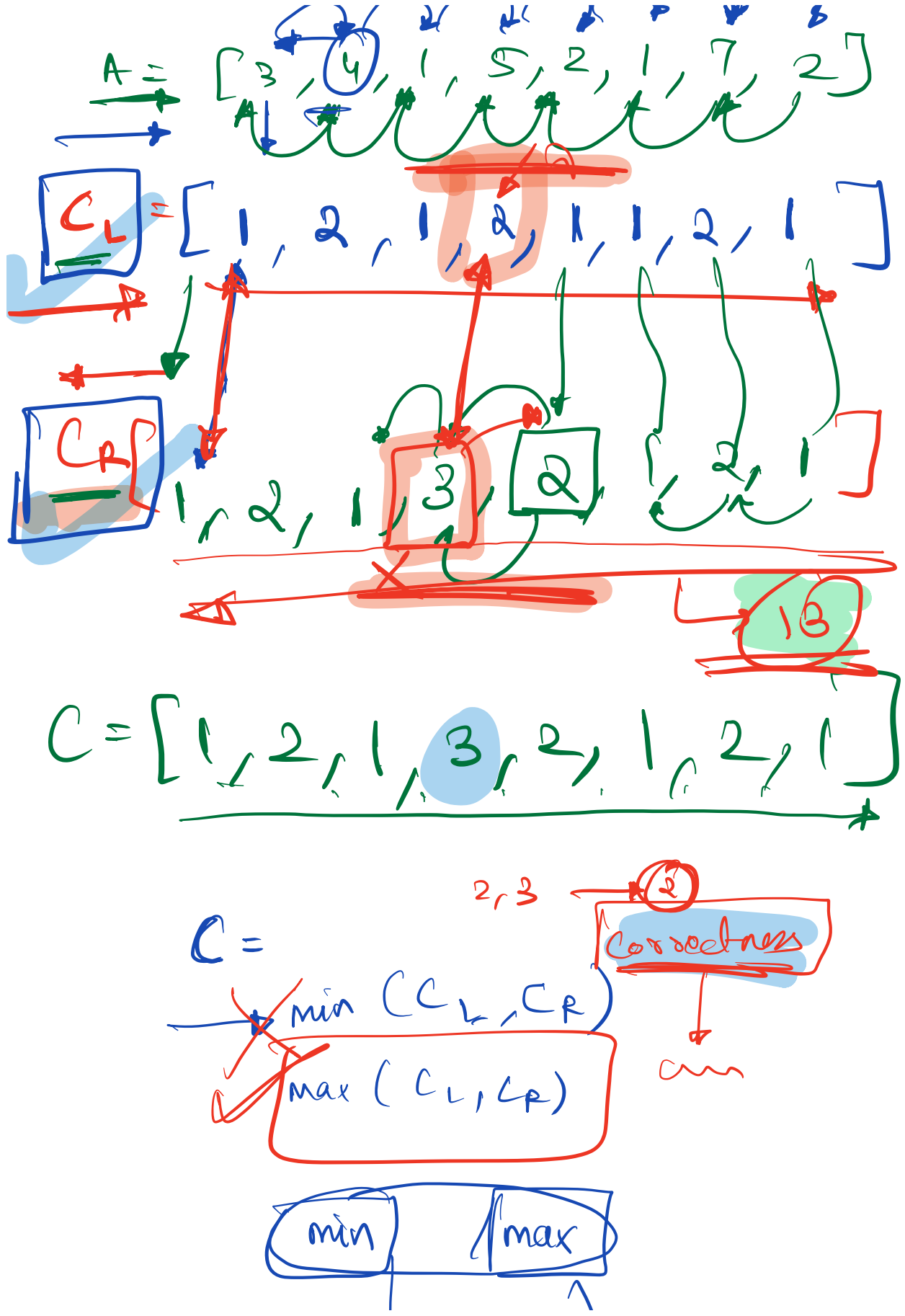
Ans : 1 + 3 + 2 + 1 → 7

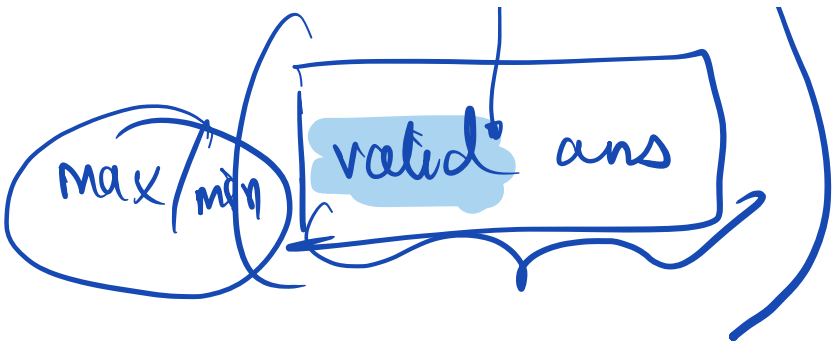
dx2 :-



Ans :- 1 + 2 + 1 + 3 + 2 + 1 + 2 + 1

Ans → 13


A = [3, 4, 1, 5, 2, 1, 7, 2]
C_L = [1, 2, 1, 2, 1, 1, 2, 1]
C_R [1, 2, 1, 3, 2, 2, 1]
18
C = [1, 2, 1, 3, 2, 1, 2, 1]
2, 3
2
Correctness
C =
min (C_L, C_R)
max (C_L, C_R)
min
max



Pseudocode :-

$C[0] = 1$

for i : 1 ⟶ n-1

L ⟶ R.

A(i) > A(i-1)

L ⟶ R

{ $C_L[i] = C_L[i-1] + 1$

}

else

{ $C_L[i] = 1$

}

}

R

$C_R[n-1] = 1$

for $i : n-2 \longrightarrow 1$

{

if $(A(i) > A(i+1))$

{

$C_R[i] = C_R[i+1] + 1$

}

else

$C_R[i] = 1$

$Sum = 0$

for $i : 0 \longrightarrow n-1$

{

$C[i] = \max(C_L[i], C_R[i])$

$Sum = Sum + C[i]$

}

TC: (n + n + n)

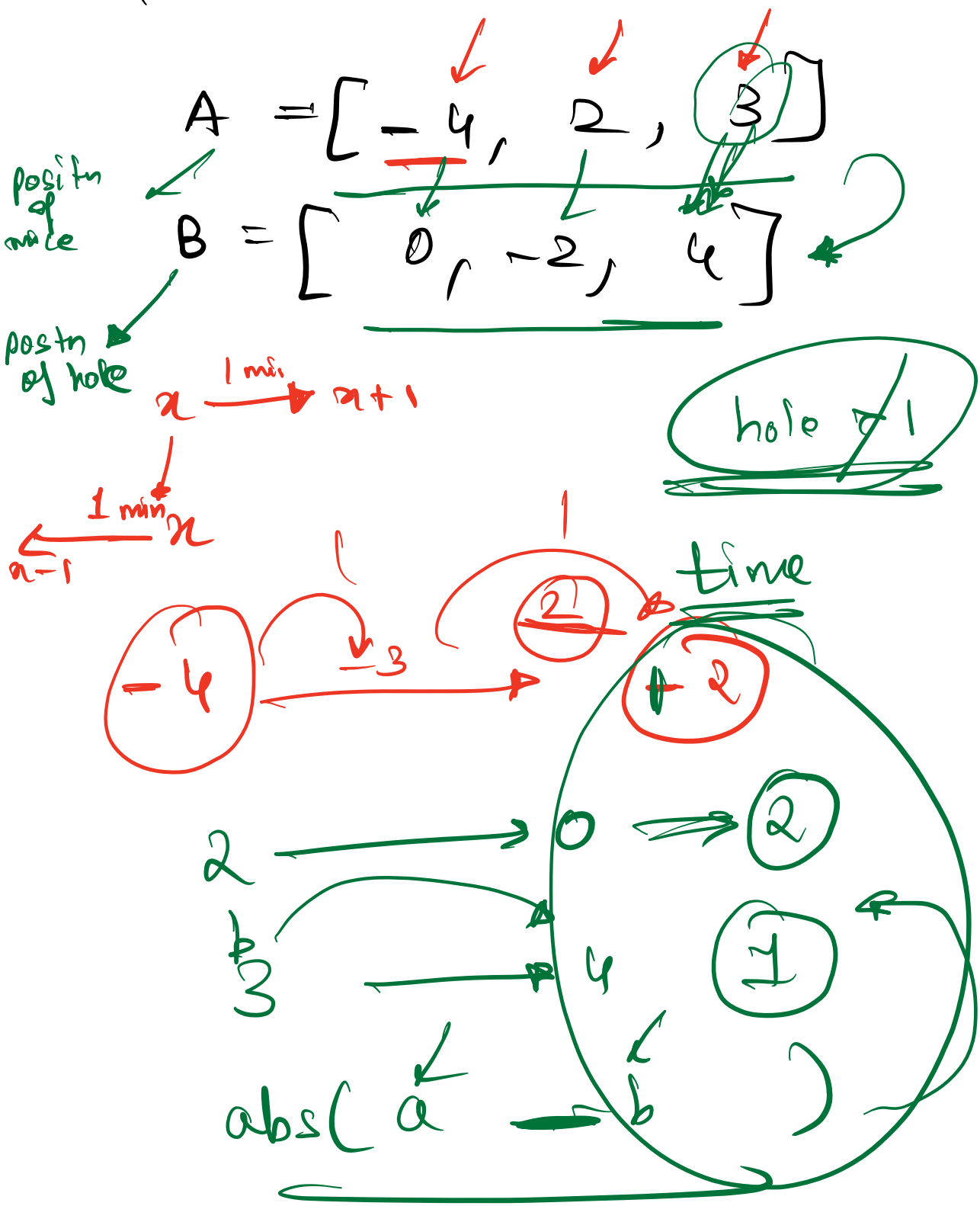
TC ⟶ O(n)

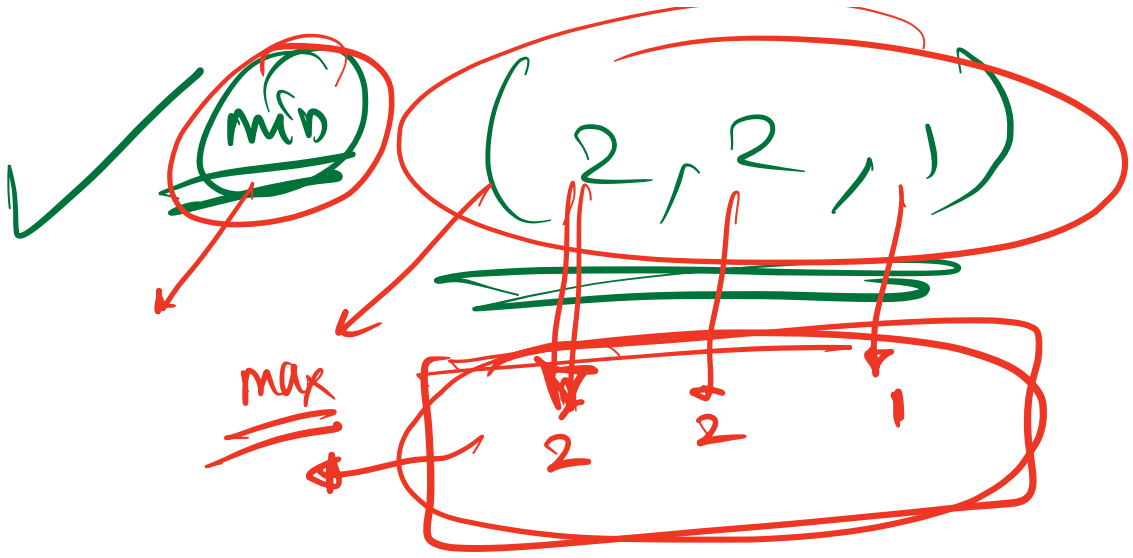
C_L, C_R, C → n

SC:

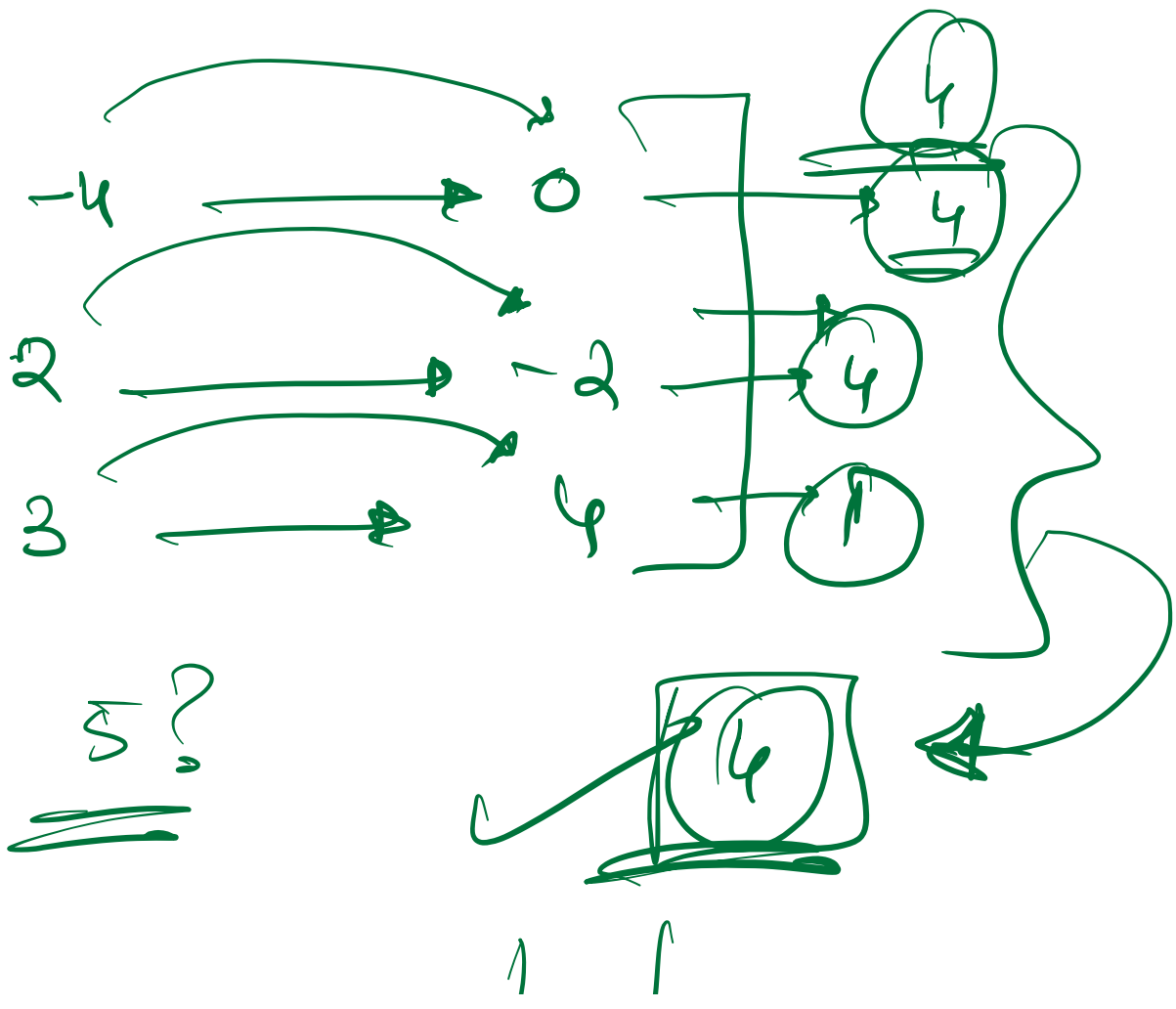
SC: O(n)

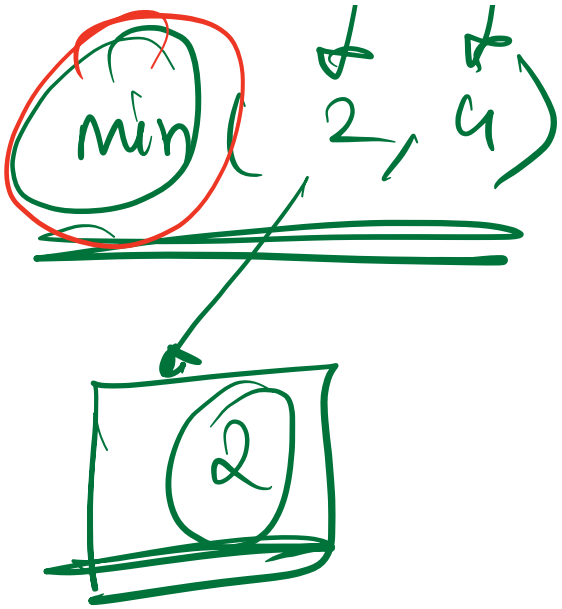
(2) Mice and Holes :





+ Correctness → min



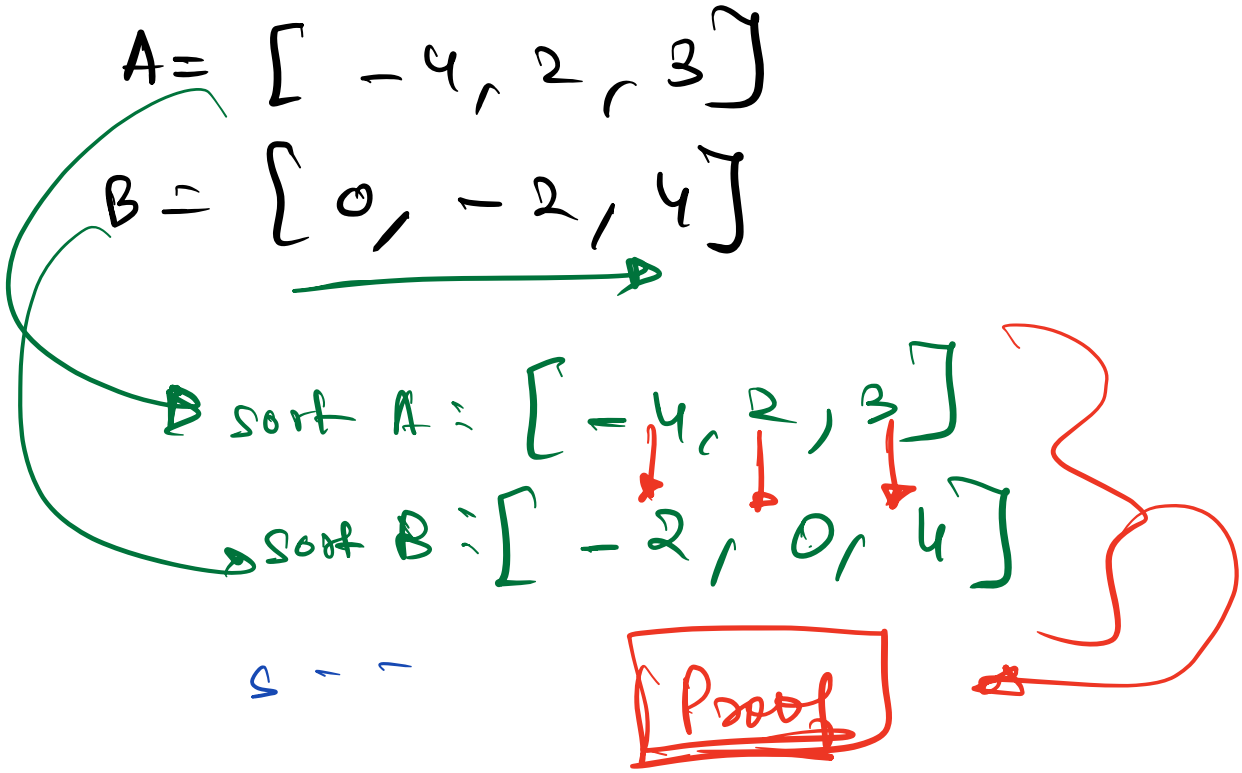


Logic :→ every mice gets assigned to the closest hole

unoccupied

that I have handled

greedy ✓


A = [-4, 2, 3]
B = [0, -2, 4]
sort A : [-4, 2, 3]
sort B : [-2, 0, 4]
Proof

Proof of Correctness

① let $i1 < i2$ be the pos of 2 mice

$i1 \neq i2$

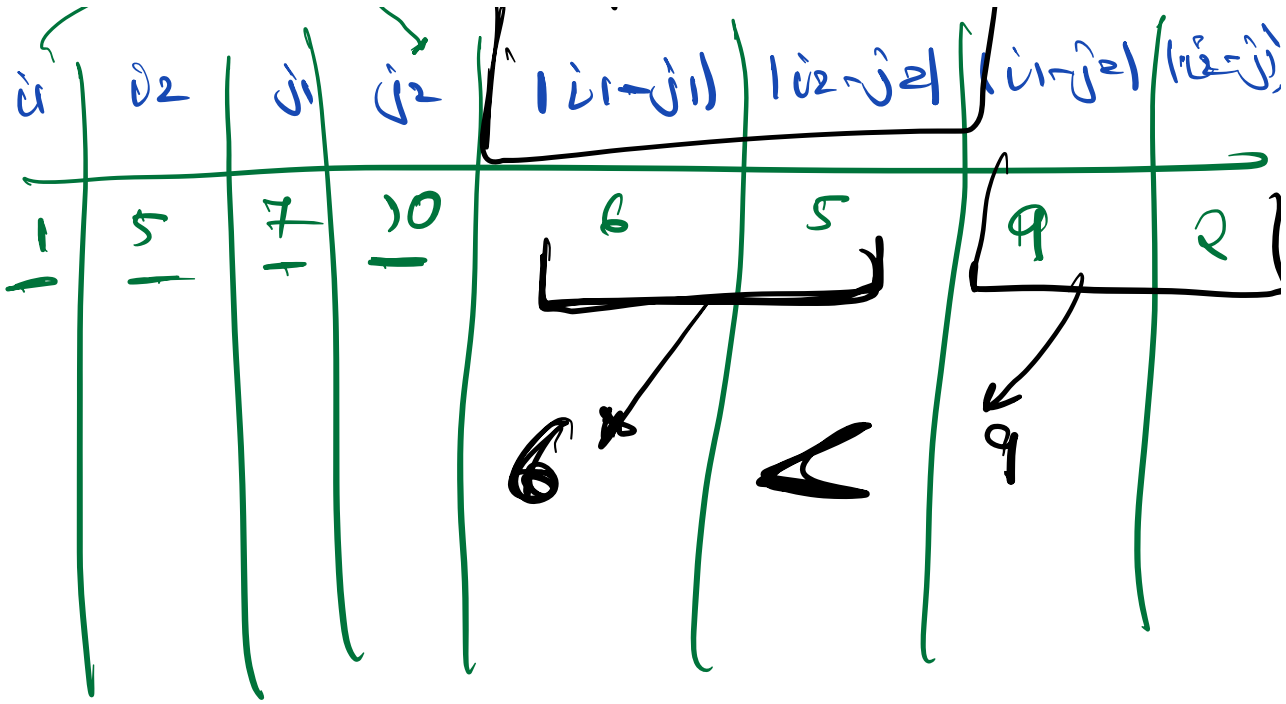
② let $j1 < j2$ // holes

max ($|i1 - j1|$ $|i2 - j2|$)

ans

$$\max\left(|i1 - j2|, |i2 - j1|\right)$$

where $|a-b|$ represents absolute diff. of $(a-b)$



$i1 < j1 < j2 < i2$

$i1 \to 2$

$i2 \to 10$

$j1 \to 4$

$j2 \to 7$

$|i1 - j1| \to 2$

$|i2 - j2| \to 3$

3

$|i1 - j2| \to 5$

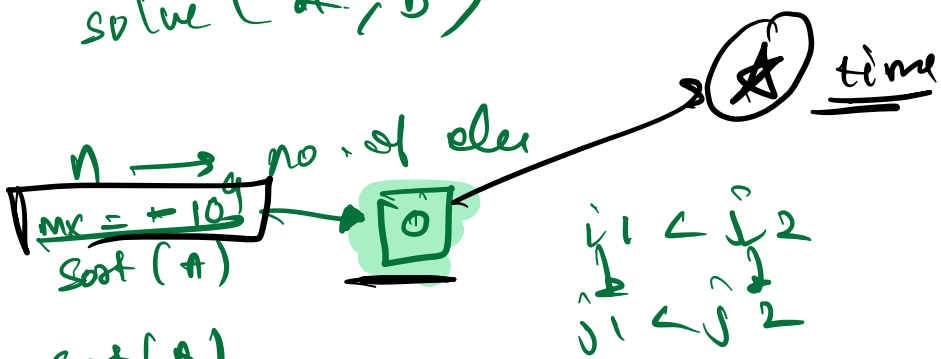
$|j1 - i2| \to 6$

6

TODO

Pseudocode :-

```
solve ( A , B )
{
   mx = -10^9
   sort ( A )
   sort ( B )
   for i : 0 ——————→ n-1
   {
   mx = max ( mx , abs( a(i) - b(i) ) )
   }
```



TC : O(n log n)

SC : Depends on Sorting Algo

10:22 → 10:32

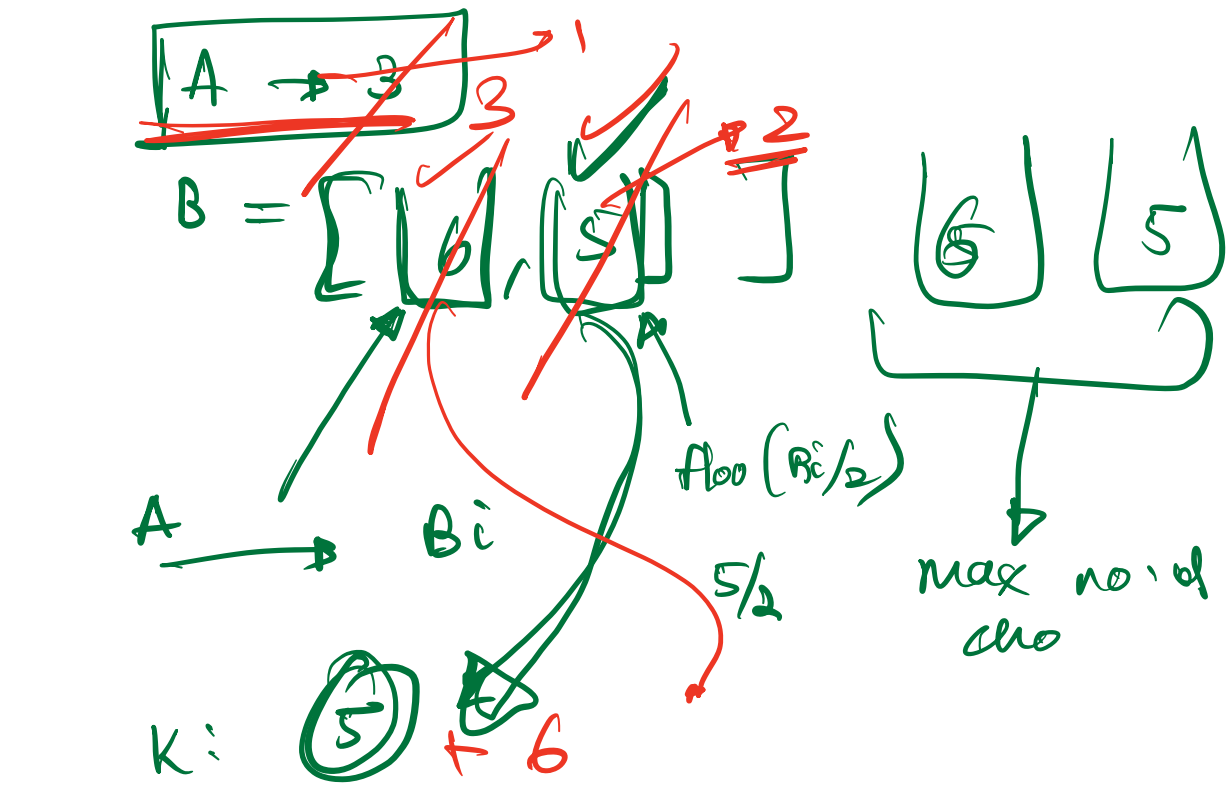
⋆ Magician and chocolats

A = 3 → 3

B = [6, 5]

t = 1 →

local opt → global optical



A=1

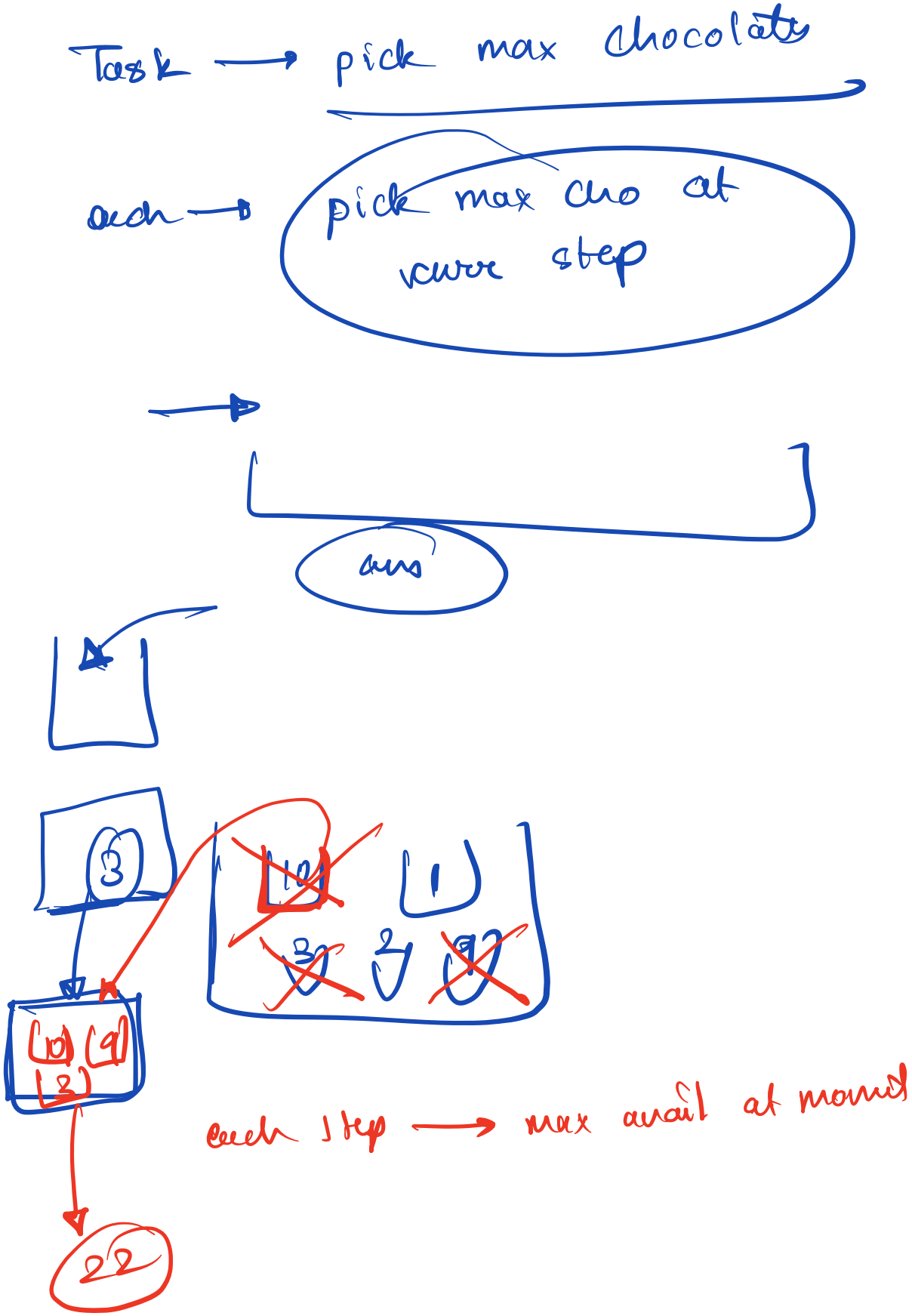
B = [3, 2]

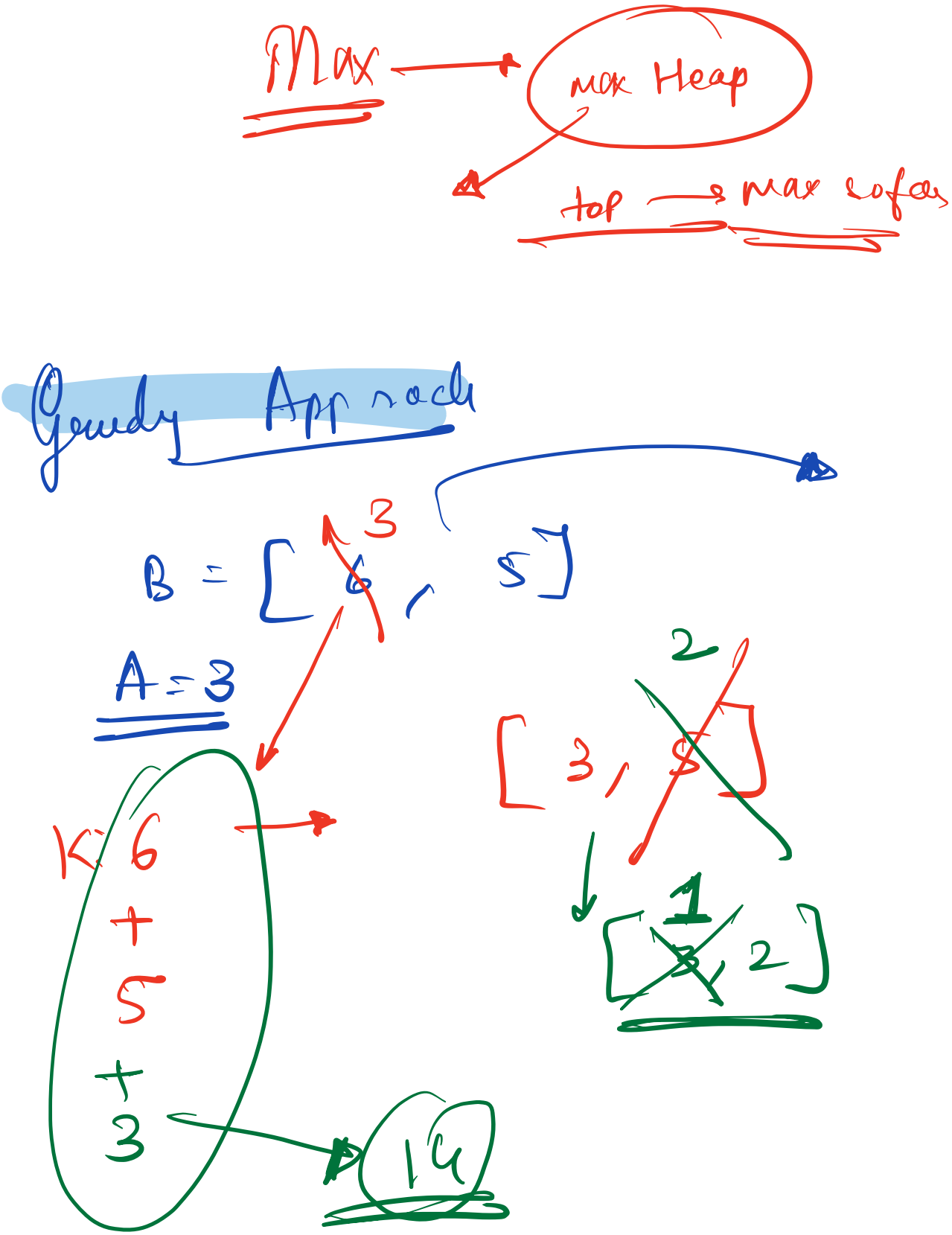
random

max (5+6+2) → Choclotes

13

14


Task → pick max chocolate
each → pick max cho at curr step
ans
10
1
3
2
9
3
22
each step → max avail at moment


Max
max Heap
top → max so far
Greedy Approach
B = [6 , 5]
3
A = 3
K: 6
+
5
+
3
14
[3 , 2]
2
1
[2]

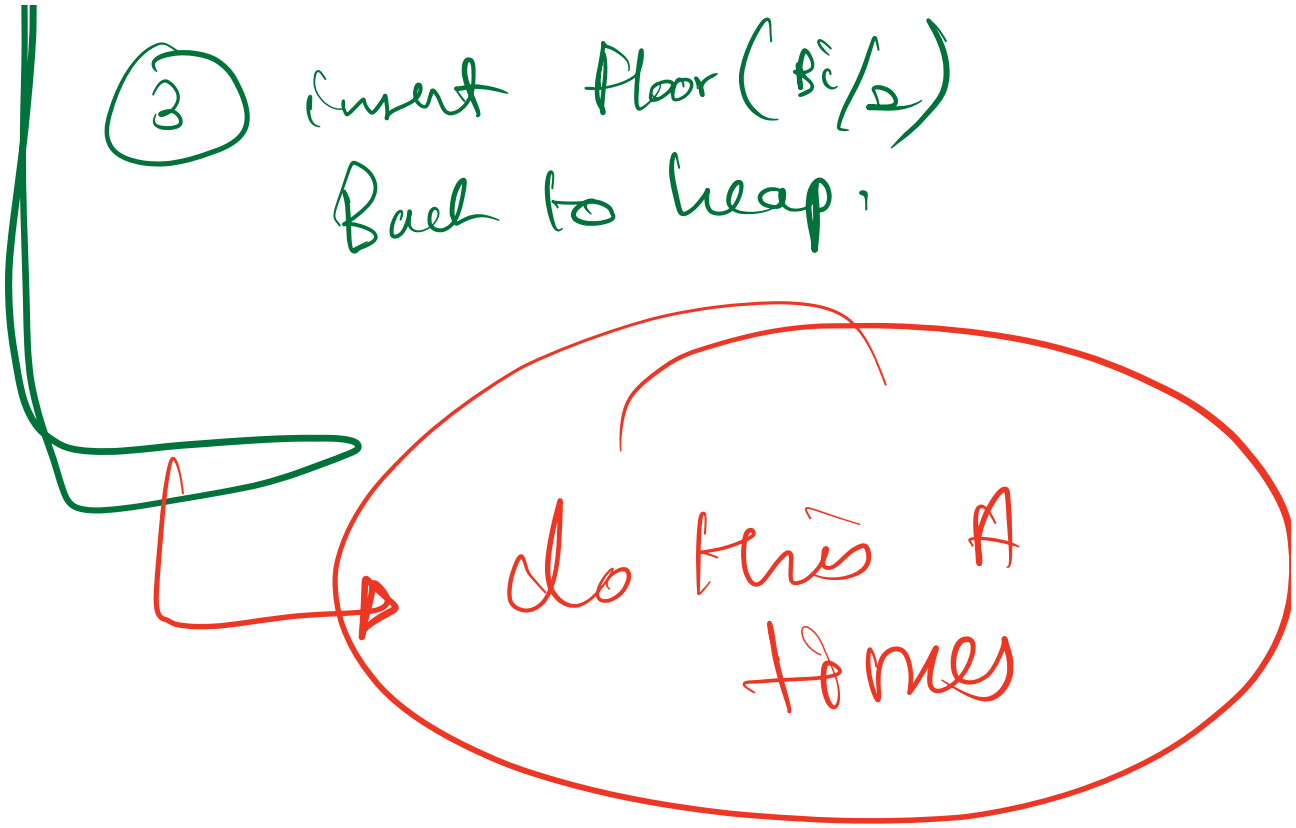
Max heap

Imp:- each time you pick Bi out, insert $floor(Bi/2)$ back into heap

Logic → ① Maintain a Max heap

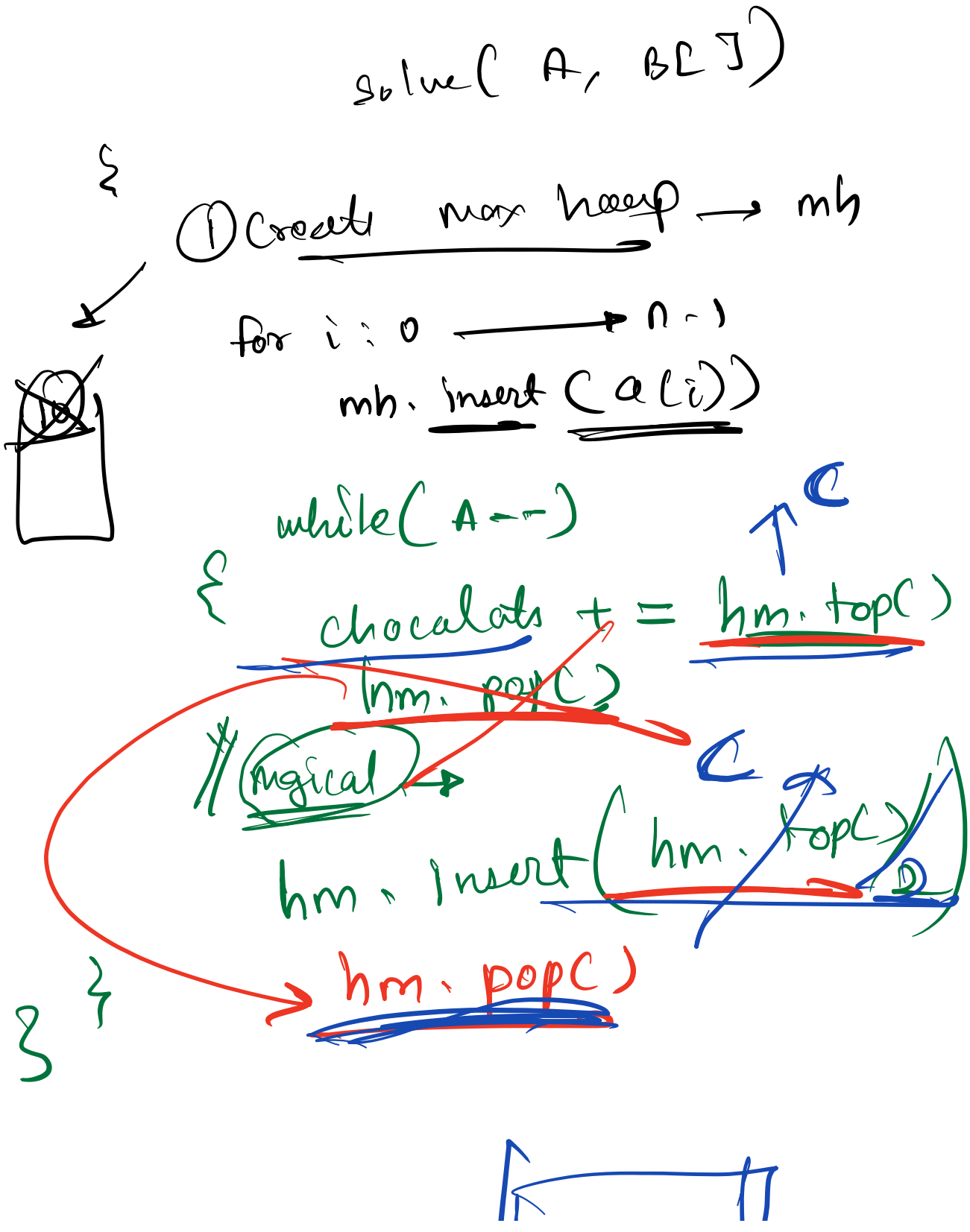
↳ top gives max among all boys chocolate at top

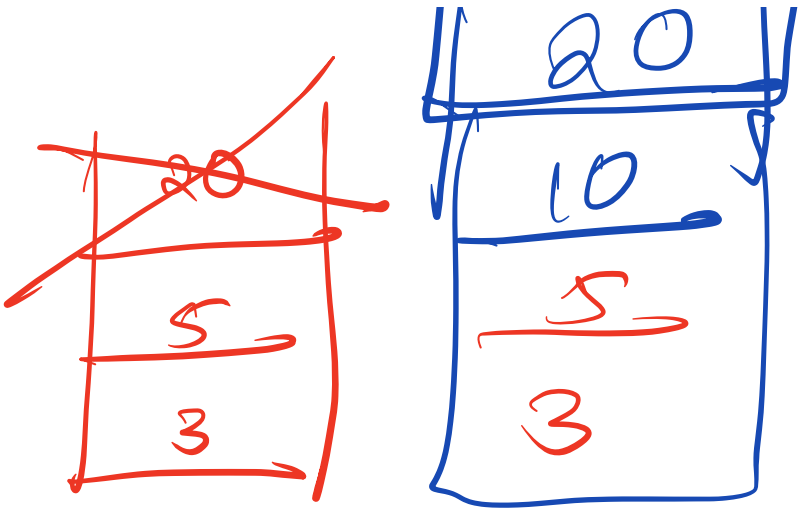
② Simply pick top of heap and add to ans.

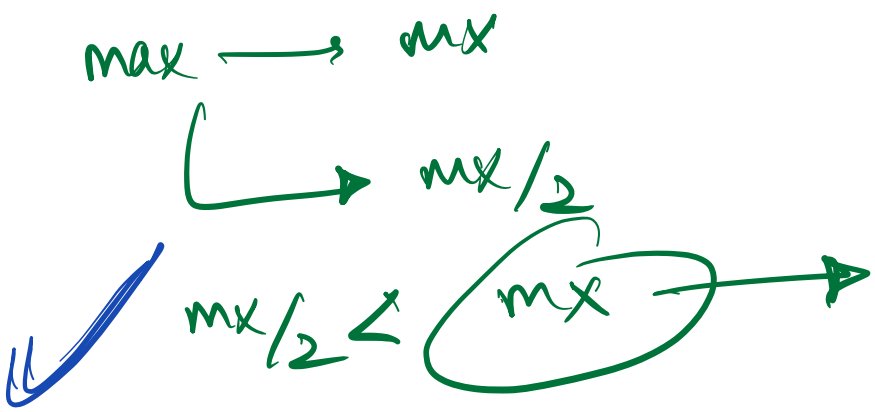


Pseudocode

Pseudocode

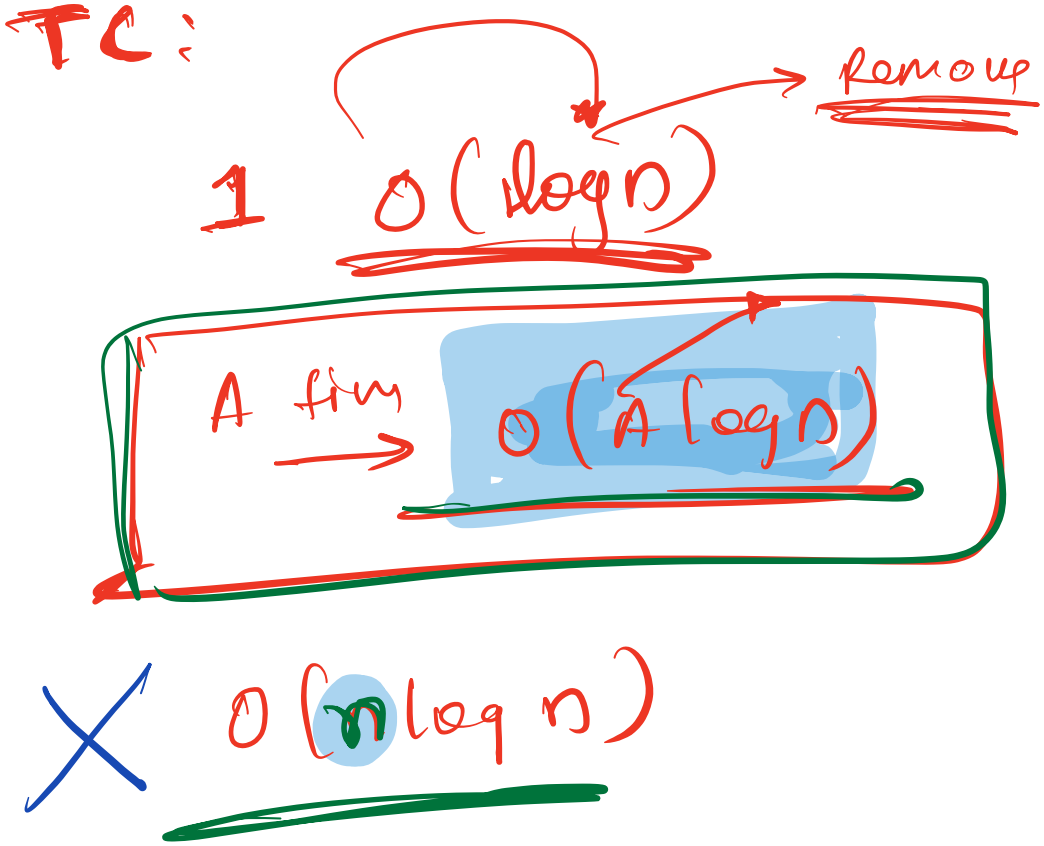

solve(A, B[ ])
{
① Create max heap → mh
for i : 0 → n-1
mh. insert (a[i])
while(A--)
{
chocolats += hm. top( )
hm. pop( )
// logical
hm. insert(hm. top( ) / 2)
hm. pop( )
}
}



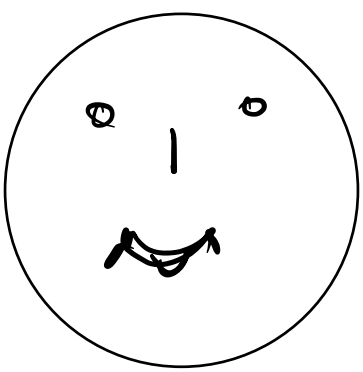


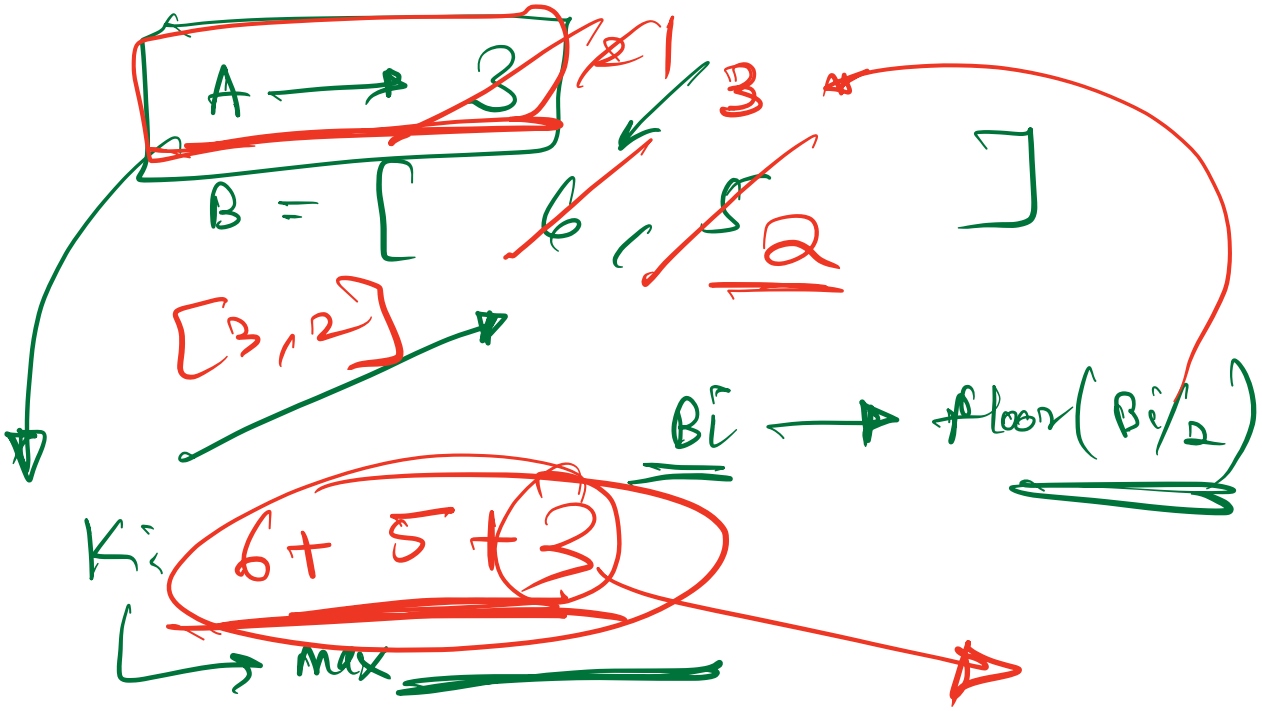

```
while(A--)
{
    c = hm.top()
    choco += c
    hm.pop()
    hm.insert(c/2)
}
return choco
```

TC:



✗ O(n log n)




A → 3
B = [6, 5, 2]
3
[3, 2]
Bi → floor(Bi / 2)
Ki
6 + 5 + 3
max