Q Given N integers. Arrange the numbers such that

$$|a_1 - a_2| <= |a_2 - a_3| <= |a_3 - a_4| <= |a_4 - a_5| <= \dots |a_{n-1} - a_n|$$

A: 1 40 42

39 2

42 40 1

2 39

A: 5 −2 4 8 6 5

40 42 1

7 6 4 2 1

8 6 5 5 4 −2

2 1 0 1 6

5 5 4 6 8 −2

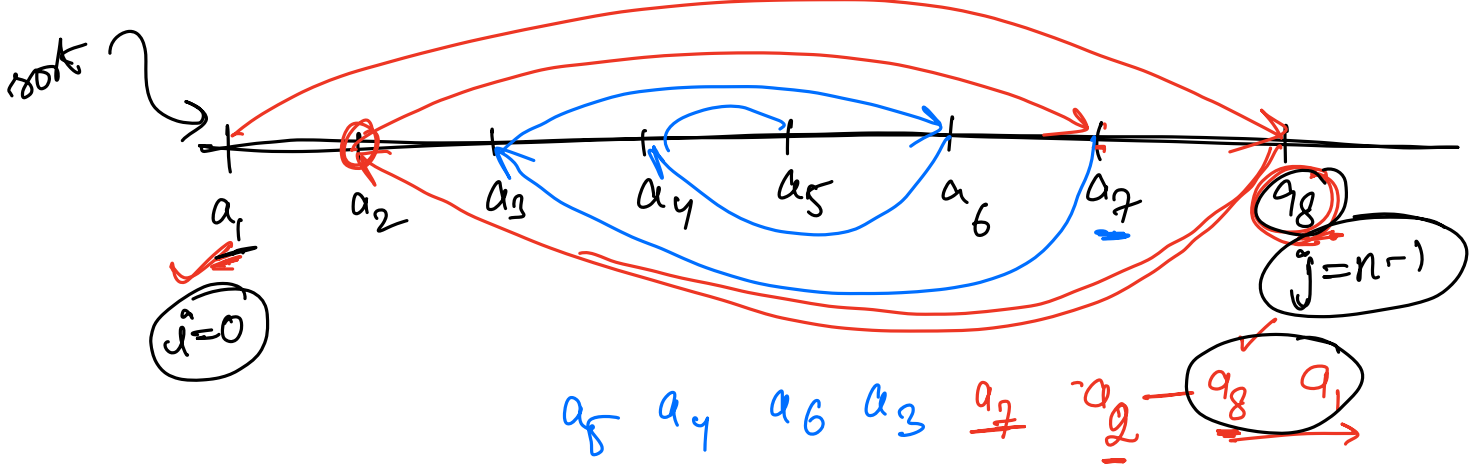
0 1 2 2 10

sorted

1 2 3 4 5 6 7 8

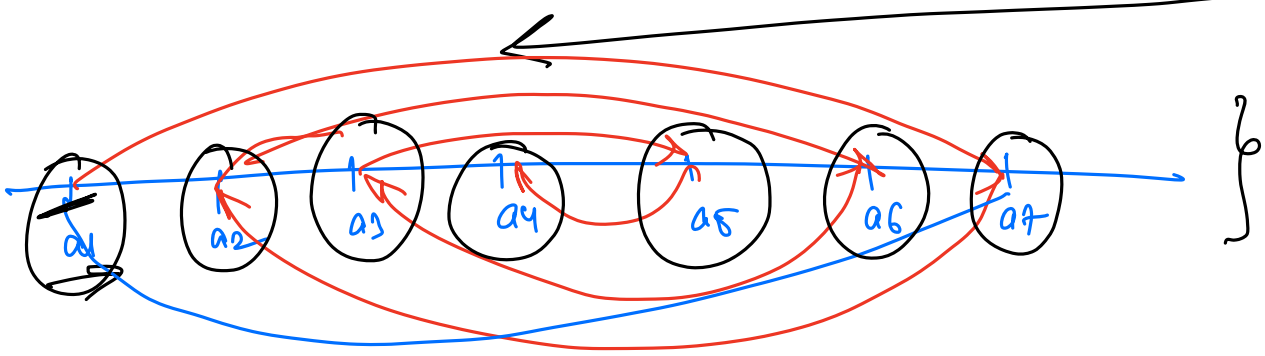
a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8

$a_5\ a_4\ a_6\ a_3\ a_7\ a_2\ a_8\ a_1$



a_5 a_4 a_6 a_3 a_7 a_2 a_8 a_1

arr[j=n-1] arr[i=0]
j--; i++;



a4 a5 a3 a6 a2 a7 a1

T.C: nlogn

S.C: O(n)

a4	a5	a3	a6	a2	a7	a1

5 -2 4 8 6 5

7 → 6

sort

-2 4 5 5 6 8

|8-(-2)|

10

5 5 6 4 8 -2

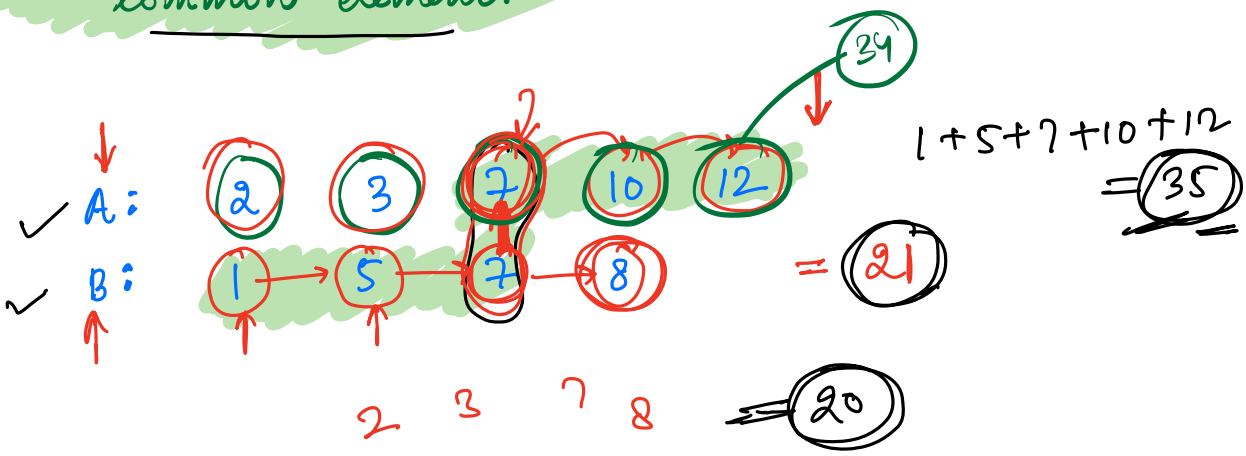
0 1 2 4 10

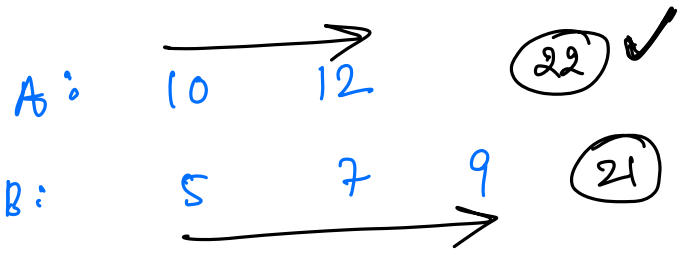
Q Two sorted arrays with distinct elements.

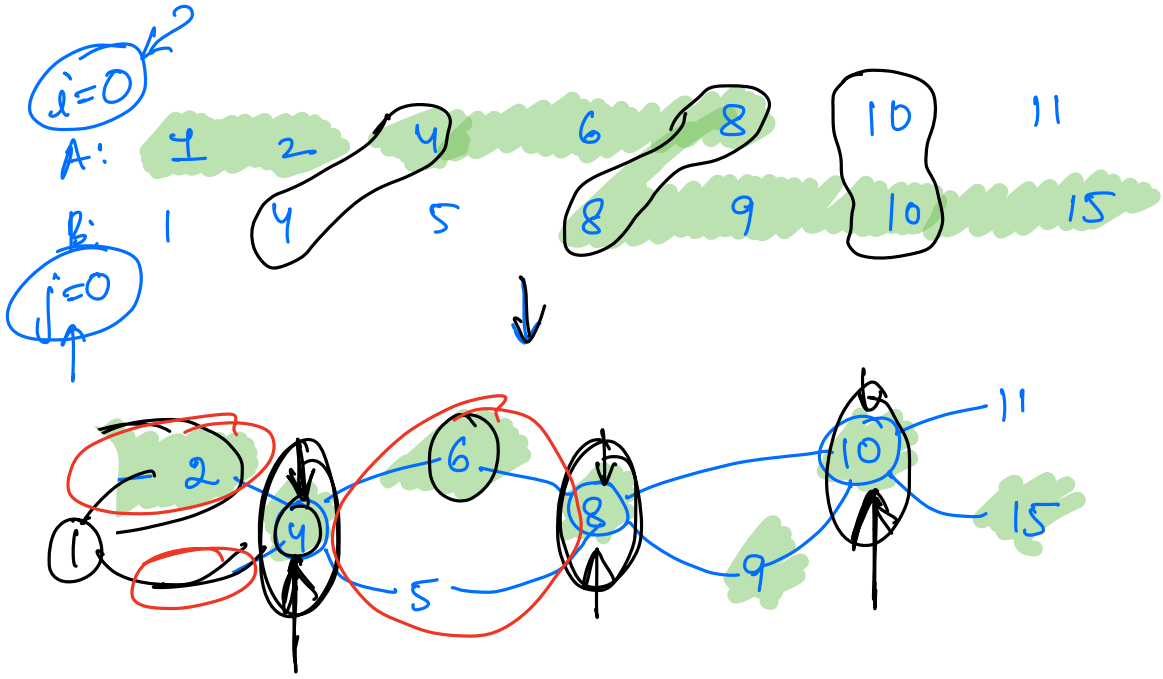
Arrays can have common elements.

Find the sum of maximum sum path to reach from start of any array to end of any array.

We can switch from one array to another at common element.





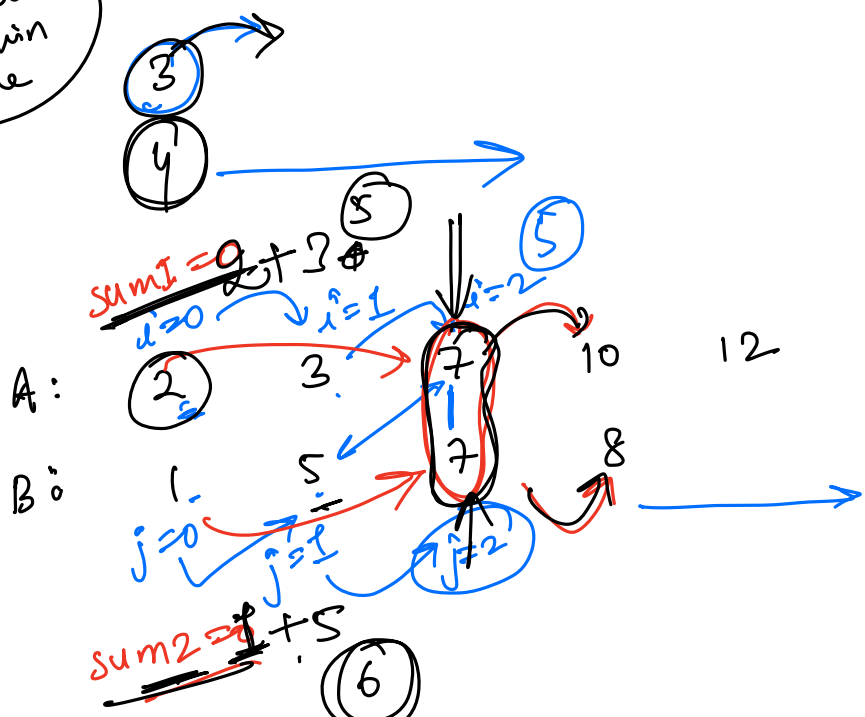


A[i]

B[j]

j=0

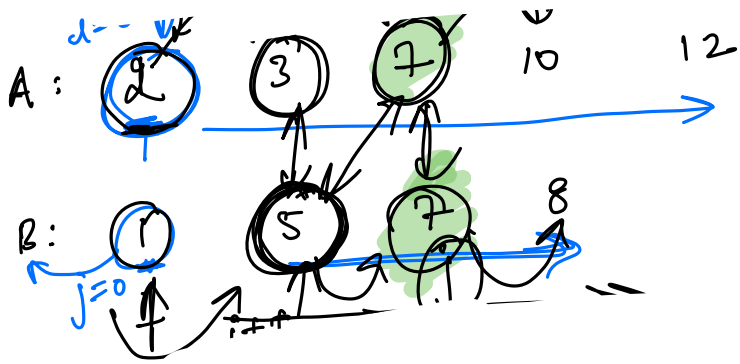
increment the min one



ans = 0

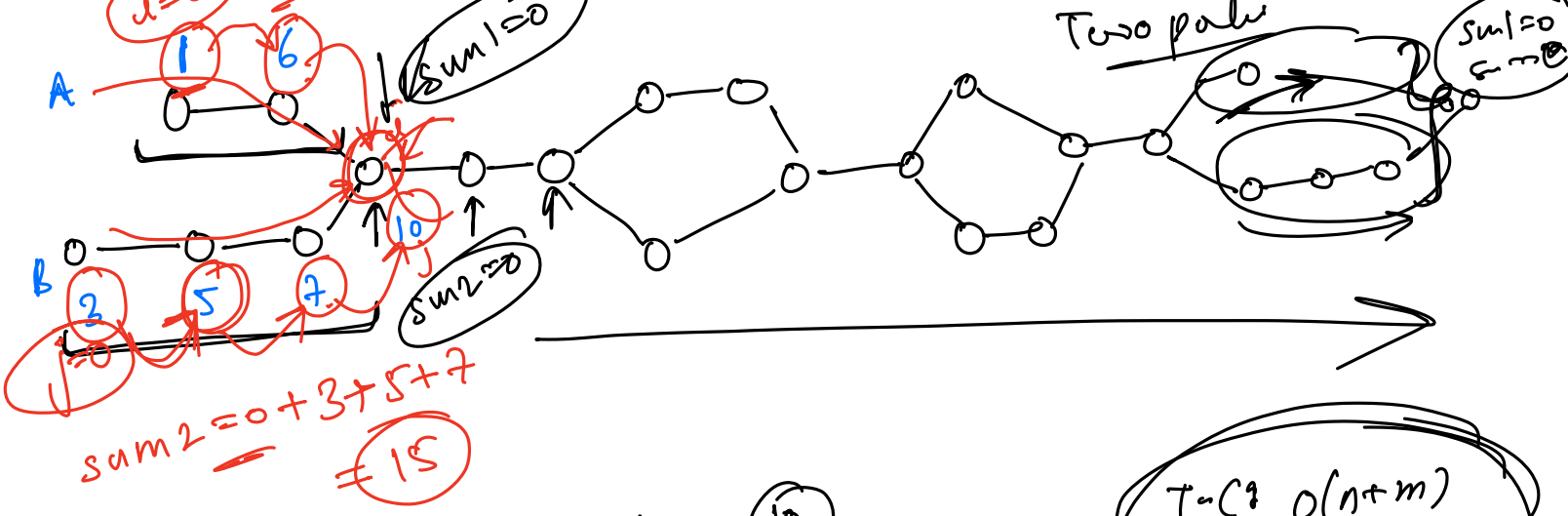
ans += max(sum1, sum2) + A[i]

sum1 = 0 sum2 = 0



i=0 sum1 = 0 + 1 + 6 = 7

merge two sorted



ans = 0

ans + = A[i]; (10)

ans + = max(sum1, sum2); (15)

ans = 25

T.C: O(n+m)

S.C: O(1)

(3) Q N miners working for you placed on y-axis.

N diamond mines placed on x-axis.

No mine or miner placed at origin.

Every miner has to mine one diamond.

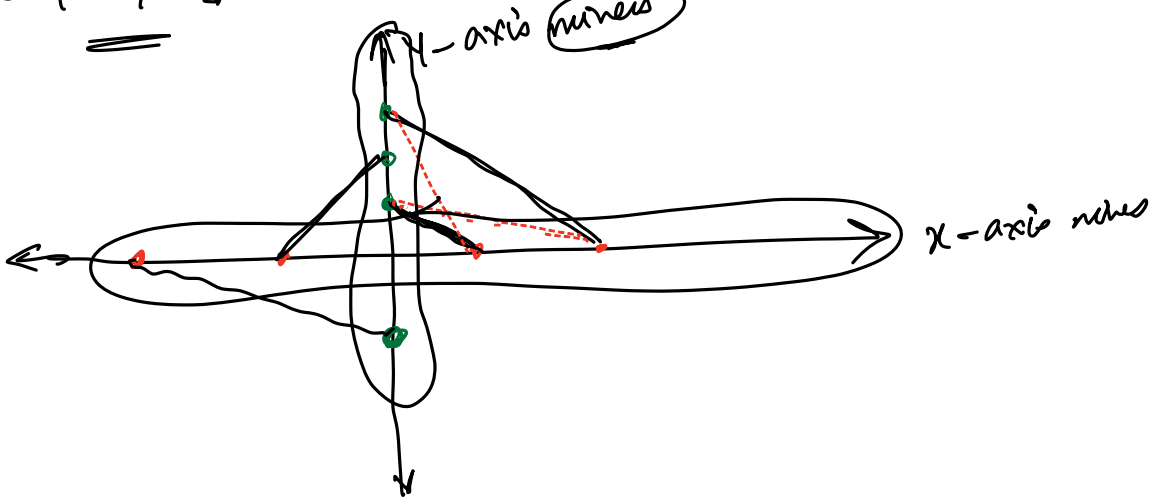
Miner (a,b)

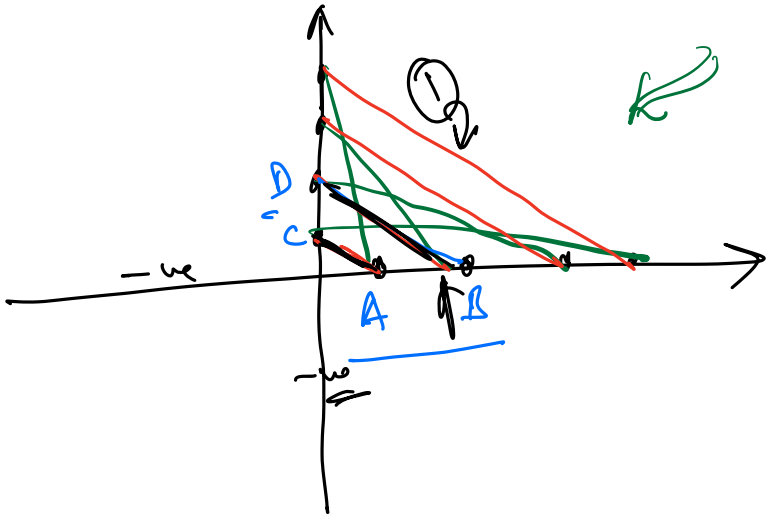
Mine (c,d)

$$\boxed{\text{Energy}: \sqrt{(a-c)^2+(b-d)^2}}$$

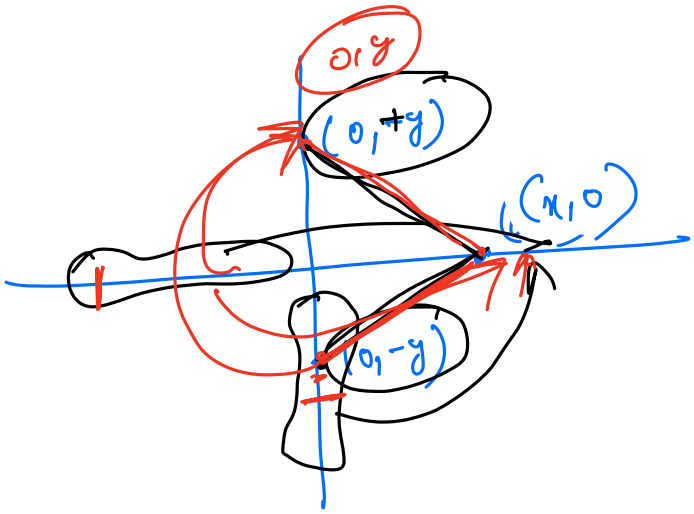
distance

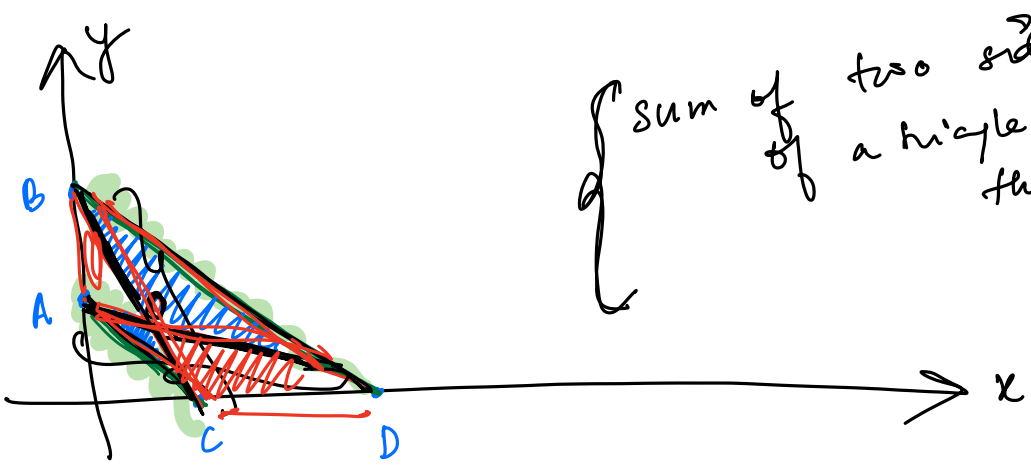
what can be the ans? minimize the total energy





positions can be negative





sum of two sides of a triangle > third side

$|AC| < |AO| + |OC|$ — (1)

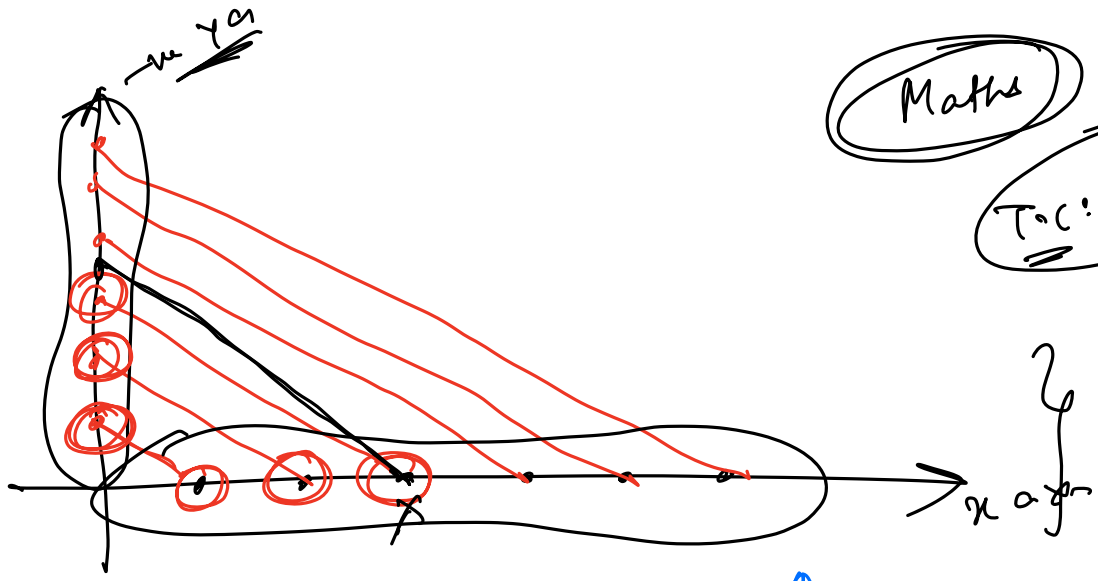
$|BD| < |BO| + |OD|$

$|AC| + |BD| < |AO| + |BO| + |OC| + |OD|$

$|BC| + |AD|$

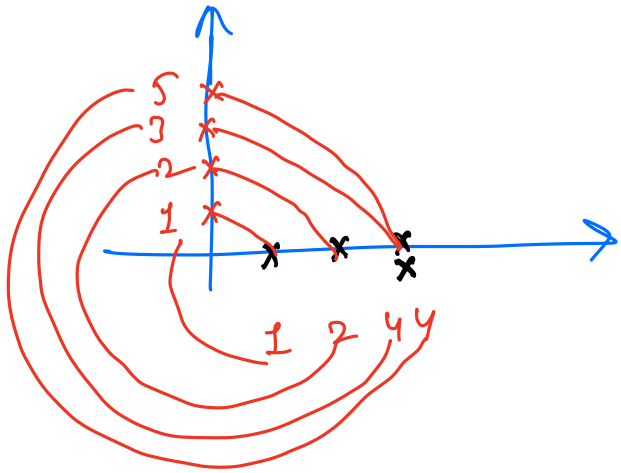
green approach

Red



Maths

T.C: nlogn



(4) Given with N array elements:

q queries → [L, R] → output = sum of elements in the range L to R.

rearrange the elements → such that total output of all queries is maximum possible.

[2 3 5]

5 Q

8

10

23

	0	1	2
	5	3	2

	0	1	2
	3	5	2

[0 1] → 8 → 8

[1 2] → 5 → 7

[0 2] → 10 → 10

23 → 25

- Assign your max no at an index which is occuring most no of queries ?

How many times every index is coming in queries?

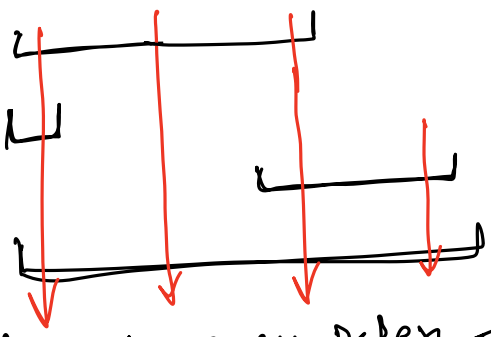
5 3 4 2

2 3 4 5

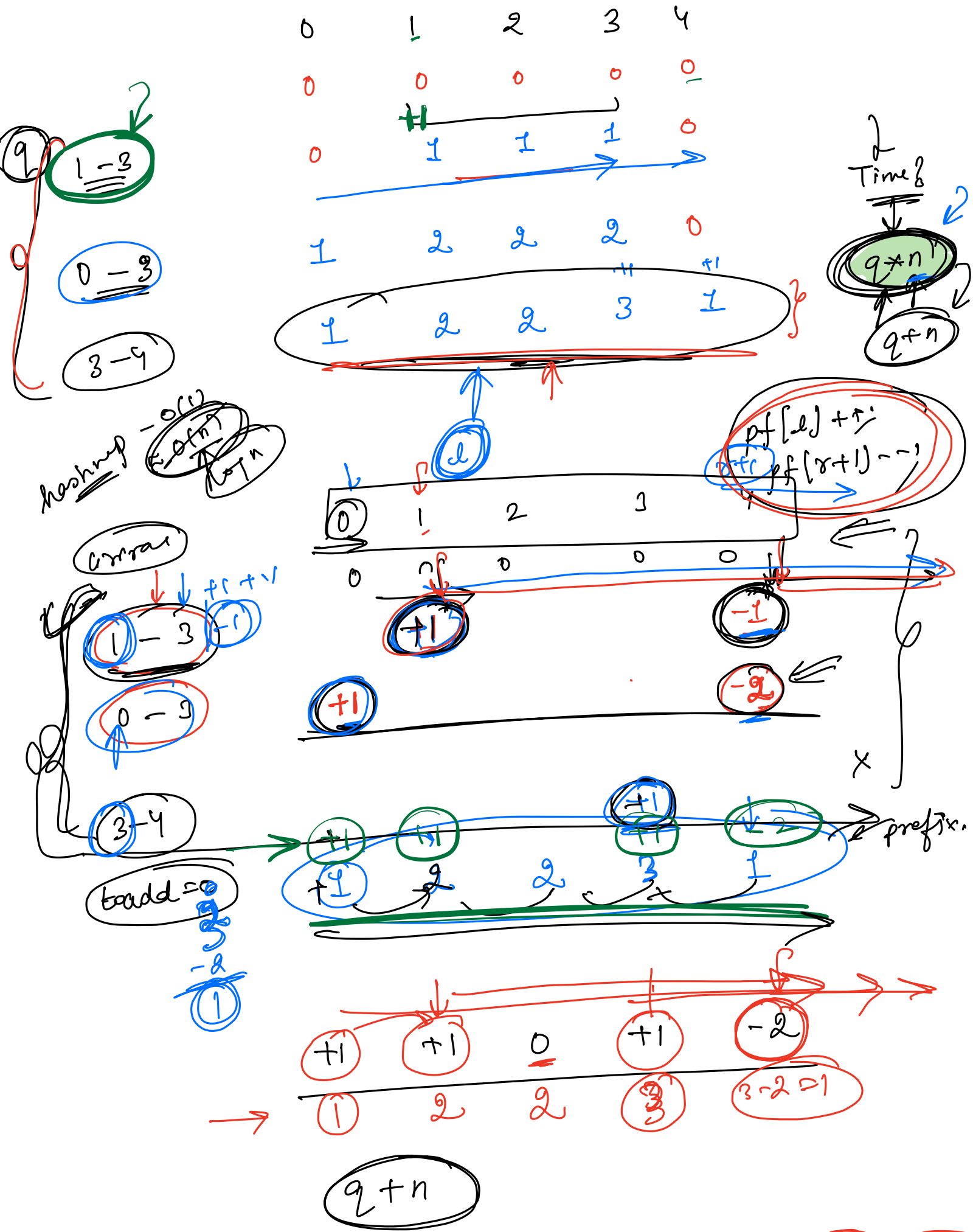
3 2 2 2

1 2 3 4

count arr



1. find the freq of every index — q+n
2. Assign values acc to larger value to larger freq index


0 1 2 3 4
q
1 - 3
0 - 3
3 - 4
Time?
q*n
q+n
hashmap - O(1)
O(n)
O(n)
array
1 - 3
0 - 3
3 - 4
+1 +1
-1
+1
-1
-2
+1
prefix.
toadd = 0
-2
1
+1 +1 0 +1 -2
1 2 2 3 3-2=1
q+n

freq array

0	1	2	3	4
2	3	1	0	2

pair & sort the

[ ]( )

max element = max freq one

sorted

sorted

4 7 1 6 9

3	2	0	4	1
0	1	2	2	3

imp → which freq is assoc with what index

4 12 14 27

T.C?

q+n
+
nlogN
+
nlogN
+
n

6 9 4 1 7

q+nlogN

BBST

Trees

log(n)
log(n)

O(1)