7:05AM

Interview tip:
fialing an interveiw session
is part of process

Topics:- 1 Sum Query
2 Prefix Sum example
break? - 3 Prefix Sum
4 Eq. index
5 Count of even numbers

Prefix sum
↓
Dynamic Programming
last three weeks
of Advanced DSA

why multiple queries?
why not one? ← [L, R]

P1 Given an array of N elements and Q queries, for each query calculate sum of all elements in range [L, R]

Constraints: L≤R $1 \leq N, Q \leq 10^5$

ex a: {-3, 6, 2, 4, 5, 2, 8, -9, 3, 1}
0 1 2 3 4 5 6 7 8 9

index ← Queries

L	R	ans
4	8	9
3	7	10
1	3	12
0	4	14
7	7	-9

2

L ← 4	8 → R
3	7
:	:
7	7

Q×2

Code:

```
void querySum(int a[], int q[][]){
    Q = q.Len // # of queries
    for(i=0; i<Q; i++){
        L = q[i][0]
        R = q[i][1]
        sum = 0
        for(j=L; j<=R; j++){
            n sum += a[j]
        }
        print(sum)
    }
}
```

Q

1000 sec

1- O(Q×n)
2- Q n
$\downarrow 10^5$ $\searrow 10^5$
3- 100 ops
4- 1GHz
5- 1sec

$10^5 \times 10^5 = 10^{10}$
10^2

$\frac{1\times10^{12}}{10^9} = 10^3$

10^3

$\left[\begin{array}{l} 10^{12} \rightarrow x \\ 10^9 \rightarrow 1sec \end{array}\right]$

10^9 ← 1GHz

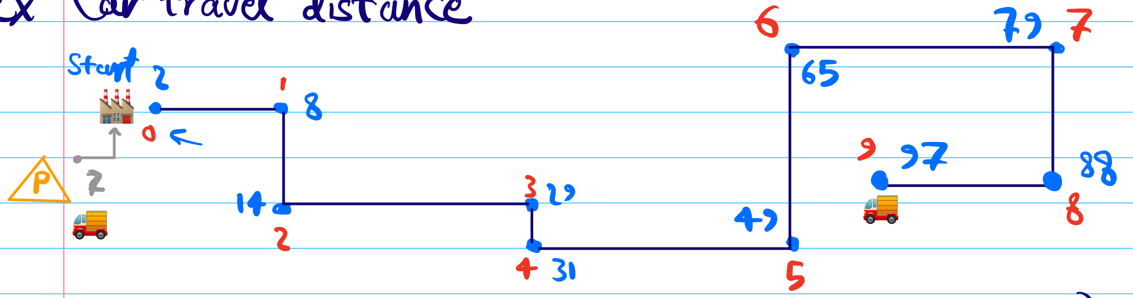
10^{12} opretion

Quiz
TC: O(Q×N)
SC: O(1)

$10^5 \times 10^5 = 10^{10}$
10-100 → 10^2

detour

ex Car travel distance



trip Computer = {2, 8, 14, 29, 31, 49, 65, 79, 88, 97} ← a
(odometer)

2	8	14	29	31	49	65	79	88	97
0	1	2	3	4	5	6	7	8	9

query: ① Last trip distanse: 97-88=9 km O(1)

Quiz

dest.
② 6th delivery, trip distance: 65-49=16km ✓
5-to-6 O(1)

dest.
③ 5th to 9th delivery, total distance:
97-49=48km ✓
O(1)

dest.
④ 1st to 5th delivery, total distance:
49-8=41km ✓
O(1)

- total distance from dest ith to dest jth ?
a[j]-a[i]

optimized idea:

$$a: \{-3, 6, 2, 4, \overbrace{5, 2, 8, -9, 3}^{9}, 1\}$$

	0	1	2	3	4	5	6	7	8	9
a	-3	6	2	4	5	2	8	-9	3	1
PS	-3	3	5	9	14	16	24	15	18	19

0 + 1 + 2 + 3 + 4 ...

$$PS: \{-3, 3, 5, 9, 14, 16, 24, 15, 18, 19\}$$

🟩 ⊖ 🟥 = 🟪

Queries

L	R	ans		
4	8	9	9✓	PS[8]−PS[3]=9
3	7	10	10✓	PS[7]−PS[3−1] = 15−5=10
1	3	12	12✓	PS[3]−PS[1−1] = 9−(−3)=12
0	4	14	14✓	~~PS[4]−PS[0−1]~~ = P[4] = 14
7	7	−9	−9✓	PS[7]−PS[7−1]=15−24=−9

PS[−1]
↓
exception
✗

O(1)

$$\circledast \quad sum(L,R)\begin{cases} PS[R]-PS[L-1] & \text{if } L>0 \\ PS[R] & \text{if } L==0 \end{cases}$$

TC: O(1)

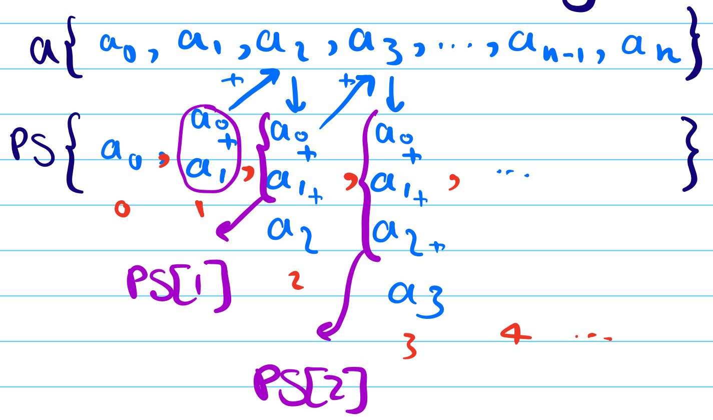
one more time :)

How to Construct prefix sum array?

Quiz ✓

14

14



a{ a0, a1, a2, a3, ..., an-1, an }
+
+
PS{ a0, a0 + a1, a0 + a1 + a2, a0 + a1 + a2 + a3, ... }
0
1
2
3
4
...
PS[1]
PS[2]

$$PS[i] = PS[i-1] + a[i]$$

```
PS[0] = a[0]
for (i = 1; i < n; i++){
    PS[i] = PS[i-1] + a[i]
}
```

TC: $O(N)$

$$Q \times O(1) \approx O(Q)$$

total $O(N+Q)$ $10^5 + 10^5 = 2 \times 10^5$

$O(Q \times N)$ 10^{10}

Optimized Code for P1

SC : O(n)

Quiz

TC : O(n) + Q x O(1)

O(n+Q)

Can we optimize SC?

if we can change the original array you may use input a[] as PS[]

```
void querySum2(int a[], int q[][]){
    n = a.Len          PS = new int[n]
    PS[0] = a[0]
    for(i=1; i<n; i++){                      Calculate PS
        PS[i] = PS[i-1] + a[i]               O(n)
    }
                                             be aware of overflow
    Q = q.Len
    for(i=0; i<Q; i++){
 Q      L = q[i][0]
        R = q[i][1]
        if(L==0) print(PS[R])                formula (*)
        else print(PS[R]-PS[L-1])            O(1)
    }
}
```

P2 Equilibrium Index:

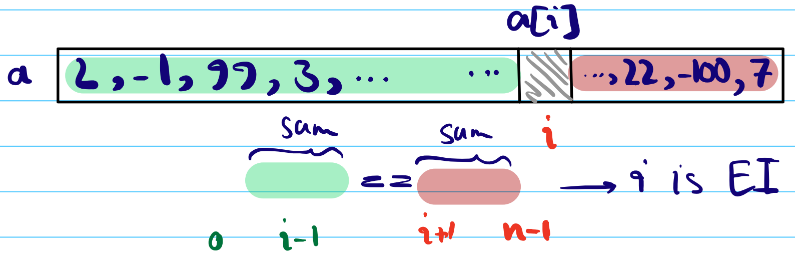
ret int

Given an array of N elements, count the number of equilibrium indexes.

what is?

equilibrium index(EI):

⚠ Sum of all elements on left of ith index == Sum of all elements on right of ith index



edge case

→ Abhishek mentioned

→ when 🟩 is 0 len

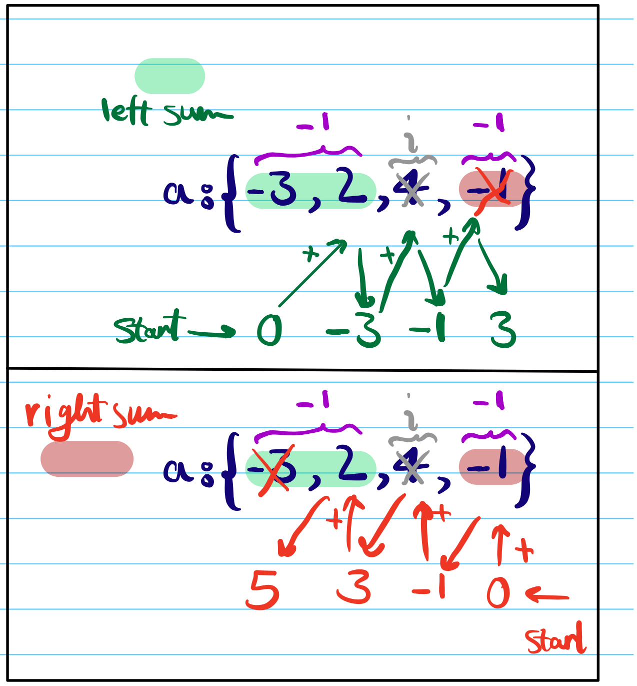
or 🟥 is 0 len

ex a: {-3, 2, 4, -1} ans = 1

indices: 0 1 2 3 → 2 is EI

→ 🟩 left sum (+ →)	0	-3	-1	3
← 🟥 right sum (← +)	5	3	-1	0

how to get left/right sum



(*) we do not use left sum & right in optimized solution; just to understand EI

(*) Left sum & right sum are not exactly equal to prefix sum & postfix sum heads up: EI is an index, not the element itself.

Quiz

$$a:\{\underbrace{-7,1,5}_{}^{-1},\ \cancel{2},\ \overbrace{-4,3,\cancel{0}}^{-1}\}$$

indices: 0 1 2 3 4 5 6

0

ans=2

Quiz

0

{prefix sum
{postfix sum

Codes:

Quiz

SC:O(n)

TC:O(n+n)
=O(n)

```
int CountEI(int a[]){
    n=a.Len        ans=0
    PS=new int[n]
    PS[0]=a[0]
    for(i=1;i<n;i++){
        PS[i]=PS[i-1]+a[i]
    }                                   } Prefix sum

    for(i=0;i<n;i++){
      if(PS[i-1]==P[n-1]-P[i]){
          ans+=1
      }
    }
    ret ans
}
```

from [0,i-1]

[i+1,n-1]

P[n-1]-P[(i+1)-1]

bug

HW

P3 Given an array of N elements and Q queries, for each query [L, R], find count of even numbers in a given range.

keep the original array intact.

ex

$$a: \{\overset{0}{2}, \overset{1}{4}, \overset{2}{3}, \overset{3}{7}, \overset{4}{9}, \overset{5}{8}, \overset{6}{6}, \overset{7}{5}, \overset{8}{4}, \overset{9}{9}\}$$

$$e: \{1, 1, 0, 0, 0, 1, 1, 0, 1, 0\}$$

	L	R	ans
x	4	8	3
x	3	9	3
x	0	4	2

Code

```
     n+n+q      void CountEven( int a[], int q[][]){
TC≈O(n+q)            e = new int[a.Len]      n = a.Len
SC≈O(2n)=O(n)        for( i=0; i<n; i++){
                        if( a[i] %2 ==0) e[i] =1
                        else e[i]=0
                     }
                  PS = new int[n]
                  PS[0] = a[0]
                  for (i=1; i<n; i++){                       Calculate PS
                        PS[i] = PS[i-1] + e[i]                  O(n)
                  }
                                                  be aware of overflow
                  Q = q.Len
                  for (i=0; i<Q; i++){
          Q          L = q[i][0]
                     R = q[i][1]
                     if( L==0) print( PS[R])            } formula (*)
                     else print( PS[R]-PS[L-1])         } O(1)
                  }
                }
```