

Q

array of size N and Q queries

s and e
(start) (end)

Find the sum of elements from s to e.
(both included)

0 1 2 3 4 5 6 7 8 9
A: -3 6 2 4 5 2 8 -9 3 1

Q: 4

Basic →

for every query

traverse the range

& find the sum.

s	e	
1	3	12
2	7	12
4	8	9
0	2	5

```
for (int j = 1; j <= Q; j++)  
{  
    // s, e    sum = 0
```

```
    for (i = s; i <= e; i++)  
    {  
        sum += arr[i];
```

```
    }  
    print(sum);
```

```
}
```

~~$O(N)$~~
 ~~$O(N)$~~
 ~~$O(N)$~~
T.C: $O(Q \times N)$
S.C: $O(1)$

$e - s + 1 \leq N$

$1 \leq N \leq 10^5$

$1 \leq Q \leq 10^4$

cricket score after every over from 41 to 50:

41	42	43	44	45	46	47	48	49	50
288	312	330	349	360	383	394	406	436	439

↓
cumulative
score

runs scored in last 5 overs

$$\cancel{439} - \cancel{383}$$

$$439 - 360 = 79$$

in 49th over

$$436 - 406 = 30$$

[42-45]

$$\text{score}[45] - \text{score}[41]$$

$$360 - 288$$

$$= 72$$

$$\text{prefix sum}[i] = \text{sum}(0-i)$$

At the
beginning

	0	1	2	3	4	5	6	7	8	9
A:	-3	6	2	4	5	2	8	-9	3	1

pfsum	-3	3	5	9	14	16	24	15	18	19
-------	----	---	---	---	----	----	----	----	----	----

$$\text{pf}[4] = A[0] + A[1] + A[2] + A[3] + A[4]$$

$$\text{pf}[5] = A[0] + A[1] + A[2] + A[3] + A[4] + A[5]$$

$$\text{pf}[5] = \text{pf}[4] + A[5]$$

$$pf[0] = A[0]$$

$$pf[1] = A[0] + A[1] = pf[0] + A[1]$$

$$pf[2] = A[0] + A[1] + A[2] = pf[1] + A[2]$$

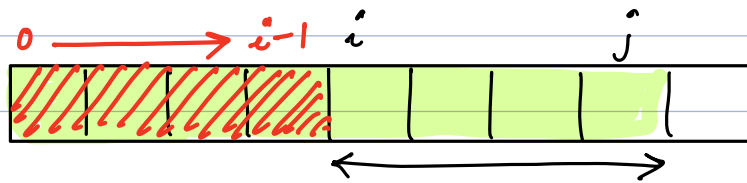
$$pf[i] = \text{sum}(0-i) = pf[i-1] + \underline{A[i]}$$

$\text{sum}(0-i-1)$

$$pf[0] = arr[0];$$

for (int i=1; i<N; i++)

$$pf[i] = pf[i-1] + arr[i];$$



$$\text{sum}(i-j) = pf[j] - pf[i-1]$$

$$\underline{i=0} \rightarrow pf[j]$$

0 1 2 3 4 5 6 7 8 9
A: -3 6 2 4 5 2 8 -9 3 1

pfsum -3 3 5 9 14 16 24 15 18 19
 ↑ ↑ ↑

s	e
1	3
2	7
4	8
0	2

$$= pf[3] - pf[0] = 9 - (-3) = 12$$

$$= pf[7] - pf[1] = 15 - 3 = 12$$

$$= pf[8] - pf[3] = 18 - 9 = 9$$

$$pf[2] = 5$$

①

$$pf[0] = arr[0];$$

for(int i=1; i<N; i++)

$$pf[i] = pf[i-1] + arr[i];$$

} N

②

for(int j=1; j<=Q; j++)

{ // s, e

if(s==0) sum = pf[e];

else sum = pf[e] - pf[s-1];

}
}

} Q

$$T.C: O(N + Q)$$

$$S.C: O(N)$$

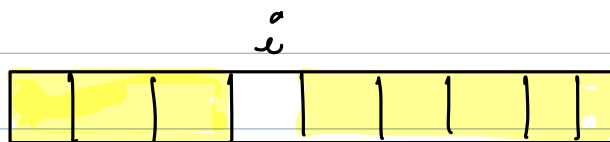
→ modify the input array itself
↓
 $O(1)$

Amazon, Adobe ...

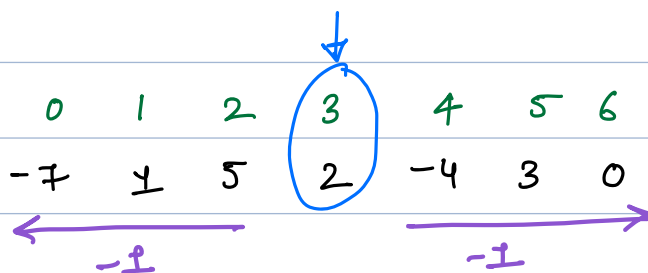
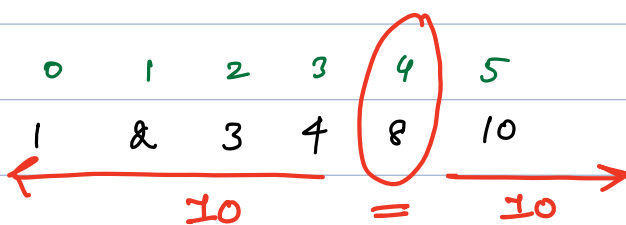
= array of size N . Find any equilibrium index.

index is said to eq^m index

sum of elements to the left = sum of elements to the right

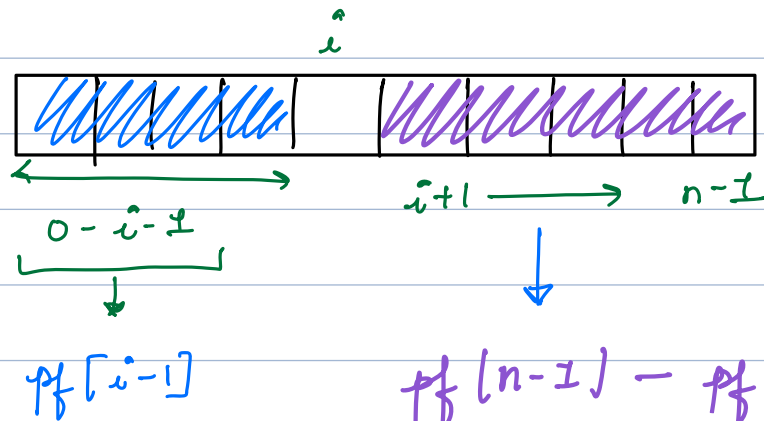


multiple ans, just need any one



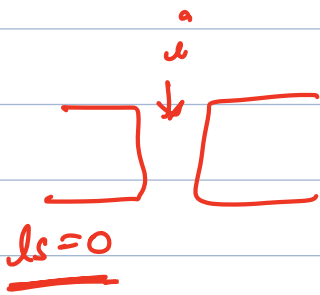
B.F. $O(N^2)$ { for every index i ✓
 $\text{sum}(0 \rightarrow i-1) == \text{sum}(i+1 \rightarrow n-1)$

Try pf sum



$$pf[n-1] - pf[i+1-1] = pf[n-1] - pf[i]$$

// build pf sum array



for($i=0; i < N; i++$)

// handle edge case.

if($pf[i-1] == pf[n-1] - pf[i]$)
return i ;

Actual / Possible

T.C: $O(N)$

S.C: $O(N)$

pfsum

Break → 10:37 pm

S.C: $O(1)$
↓
leftsum / rightsum
HW

Q

given an array of size N and Q queries

$[start, end]$

Find count of even numbers in the given range.

Ex:-

0	1	2	3	4	5	6	7	8	9
2	4	3	7	9	8	6	5	4	9

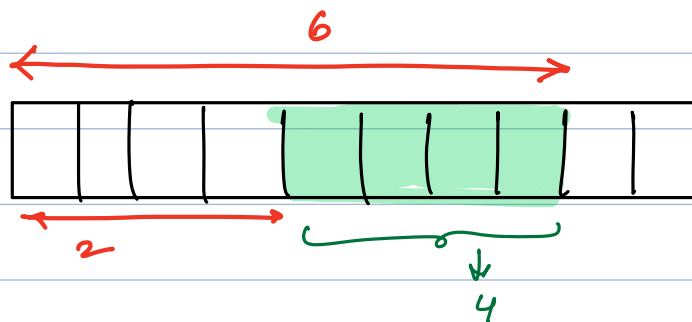
s	e	
3	7	$\rightarrow 2$
1	8	$\rightarrow 4$
0	6	$\rightarrow 4$ <u> </u>

Basic approach \rightarrow

\downarrow for every query
iterate the range &
check the answer

\downarrow
if $(a[i] \% 2 == 0)$
 $cnt++$

T.C: $O(Q * N)$



0	1	2	3	4	5	6	7	8	9
2	4	3	7	9	8	6	5	4	9

↓
pf even count 1 2 2 2 2 3 4 4 5 5

$pf_{even\ count}(i) = \text{total count of even } (0-i)$

$$even\ count(i-j) = pf_{even\ count}(j) - pf_{even\ count}(i-1)$$

$pf(0) = 0$

$if (arr[i] \% 2 == 0) pf(0) = 1;$

for (int i=1; i<N; i++)
 {

$pf_{even}(i) = pf_{even}(i-1);$

$if (arr[i] \% 2 == 0)$

$pf_{even}(i)++;$

}

Takeaway →

ranges

again & again (sum, count)

pfsum

