

Q Given an array of size N

and Q queries of the format s & e .
↑↑ start index
↑↑ end index

Return the sum of elements from index s to e .
(including both)

0 1 2 3 4 5 6 7 8 9

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

$Q: 4$

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

Basic: for every
query, traverse
& calculate the sum

```
for (int i=1; i<=q; i++)
```

```
    // s, e    sum=0;
```

```
    for (j=s; j<=e; j++)
```

```
        sum += arr[j];
```

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

```
}
```

T.C: $O(q*N)$
↑ input

given with total score after every over of last 10 overs!

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

↓
cumulative data

after finish of 48th

total runs were scored in last 5 overs
↓
[46-50]

$$439 - 360 = 79$$

last over [50-50]
50th over =

$$439 - 436 = 3$$

49th over [49-49]

$$= 436 - 406 = 30$$

42th to 45th over [42-45]

$$= 360 - 288 = 72$$

prefix sum -
 i^{th}

sum of all elements
from $0 \rightarrow i^{th}$ index

	0	1	2	3	4	5	6	7	8	9
A:	-3	6	2	4	5	2	8	-9	3	1
	↓									
pf:	-3	3	5	9	14	16	24	15	18	19

$pf[i] = \text{sum from } 0 \text{ to } i$

$$pf[5] = A[0] + A[1] + A[2] + A[3] + A[4] + A[5]$$

$$pf[6] = A[0] + A[1] + A[2] + A[3] + A[4] + A[5] + A[6]$$

$$pf[6] = pf[5] + A[6]$$

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

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

\uparrow \uparrow
 $0 \text{ to } i$ $0 \text{ to } (i-1)$

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

	0	1	2	3	4	5	6	7	8	9
A:	-3	6	2	4	5	2	8	-9	3	1
		↙	↗	↗	↗	↗	↗	↗	↗	↗
pf	-3	3	5	9	14	16	24	15	18	19

$pf[N];$

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

for ($i=1; i \leq N; i++$)

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

T.C: $O(N)$

S.C: $O(N)$

=

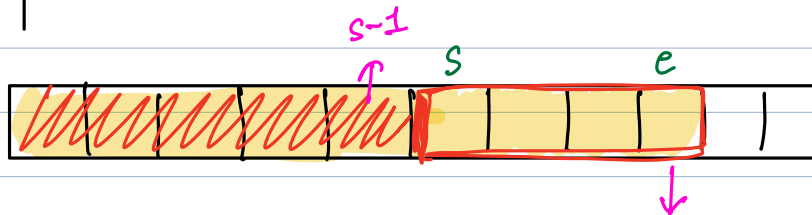
	0	1	2	3	4	5	6	7	8	9
A:	-3	6	2	4	5	2	8	-9	3	1
	-3	3	5	9	14	16	24	15	18	19



s	e
1	3
2	7
4	8
0	2

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

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



$$\left. \begin{array}{l} pf[end] - pf[start-1] \\ \text{if } start == 0 \text{ ans} = pf[end] \\ \text{0-end} \end{array} \right\}$$

// Build pf sum - N iter

q iterations

```

for (int i=1; i<=q; i++)
{
    // start, end
    if (start == 0) sum = pf[end];
    else sum = pf[end] - pf[start-1];
}

```

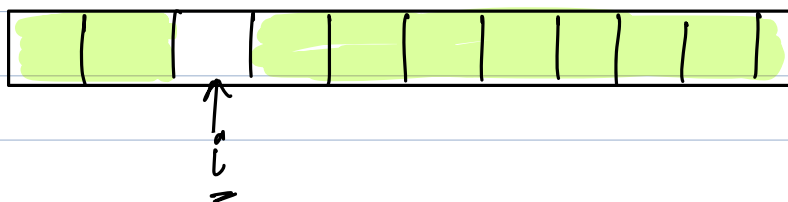
T.C: $O(N+q)$ *no of queries*
 S.C: $O(N)$

Q Given an array of size N. Find equilibrium index.

index is called eqⁿ if

sum of elements on left of the index = sum of element on right of the index

"do not" include i



0	1	2	3	4	5
1	2	3	4	8	10

Sum of elements from index 0 to 3 is 10. Sum of elements from index 5 to 5 is 10.

0	1	2	3	4	5	6
-7	1	5	2	-4	3	0

pf(end)
- pf(start-1)

$$\text{sum}(0 \text{ to } i-1) = \text{sum}(i+1 \text{ to } n-1)$$

pf(i-1)

pf(n-1) - pf(i+1-1)

$$= \text{pf}(n-1) - \text{pf}(i)$$

// pf array

for (i=1; i<n-1; i++)

if (pf(i-1) == pf(n-1) - pf(i))

return i;

}

Corner cases
if (i=0)
if (i=n-1)

N+N

T.C: O(N)

S.C: O(N)

O(1)

HW

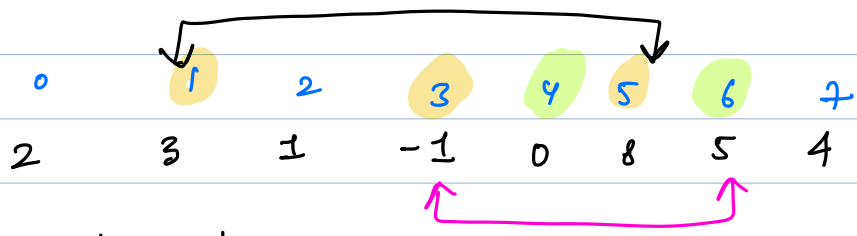
don't modify original array

Q Given an array of size N . You are asked

q queries
 start end type

1 \rightarrow sum of all even indices elements

2 \rightarrow sum of all odd indices elements



②

s	e	type
3	6	1
1	5	2

$0 + 5 = 5$

$3 + -1 + 8 = 10$

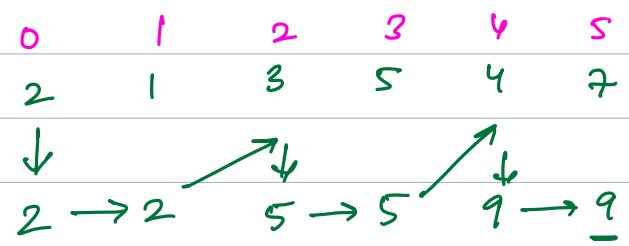
Basic ?

②

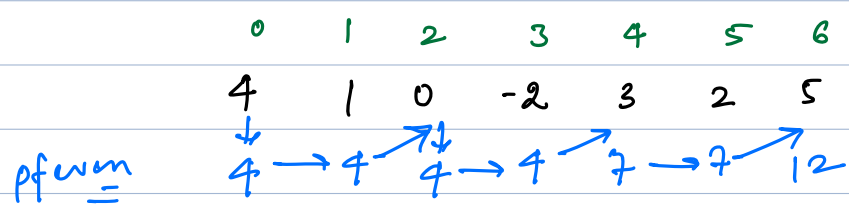
T.C: $n \times q$

oddpsum
 evenpsum

pf_{even}



$pf_{even}(i) = \text{sum of even index from } (0 \sim i)$



pfodd

0	1	2	3	4	5
2	1	3	5	4	7
↓	↗	↓	↗	↓	↗
0	1	→ 1	6	→ 6	13

pfodd

0	1	2	3	4
2	4	3	1	5
0	4	4	5	5

	0	✓ 1	2	3	4	5	6	7
	2	3	1	-1	0	8	5	4
pfeven	2	2	3	3	3	3	8	8
pfodd	0	3	3	2	2	10	10	14

s	c	type
3	6	1
1	5	2

$$pfeven[6] - pfeven[2] = 8 - 3 = 5$$

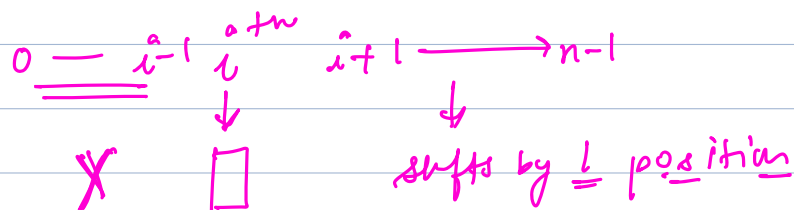
$$pfodd[5] - pfodd[0] = 10 - 0 = 10$$

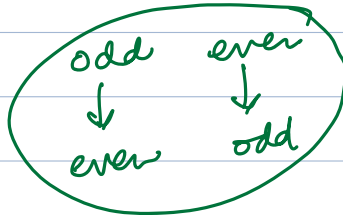
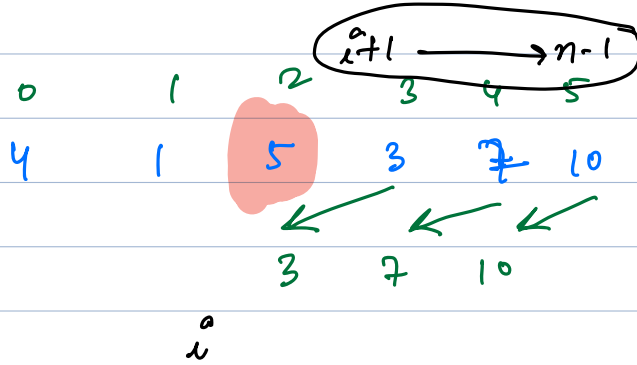
Q Given an array of size N . Count number of special index

If we delete this index element

$$\text{sum of odd indices} = \text{sum of even indices}$$

	0	1	2	3	4	5	even	odd
	4	3	2	7	6	-2		
$i=0$	3	2	7	6	-2		8	8
$i=1$	4	2	7	6	-2		9	8
$i=2$	4	3	7	6	-2		9	9
$i=3$	4	3	2	6	-2		4	9
$i=4$	4	3	2	7	-2		4	10
$i=5$	4	3	2	7	6		12	10





0 \rightarrow $i-1$ $i+1 \rightarrow n-1$

even indices
sum

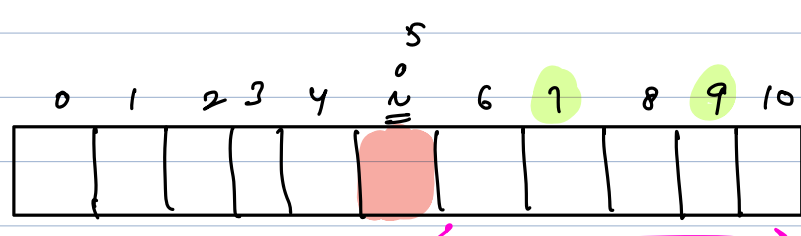
$sum_{even}(0 \rightarrow i-1)$
 \downarrow
 $pf_{even}(i-1)$

$pf_{odd}[n-1] - pf_{odd}[i]$

odd indices
sum

$pf_{odd}(i-1)$

$pf_{even}[n-1] - pf_{even}[i]$



even
indices
sum

$pf_{even}[4]$

pf_{odd}

// $pf_{even} \rightarrow O(N)$
// $pf_{odd} \rightarrow O(N)$
 $ans = 0;$

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

handle the
base case
for $i = 0$

$total_{even} = pf_{even}(i-1) + pf_{odd}(n-1) - pf_{odd}(i);$

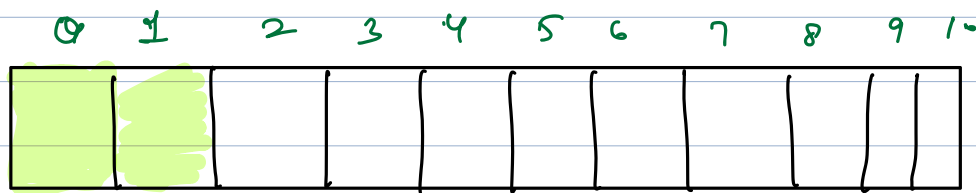
$total_{odd} = pf_{odd}(i-1) + pf_{even}(n-1) - pf_{even}(i);$

if ($total_{even} == total_{odd}$)
 $ans++;$

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

$O(1)$
HW - take
the value
from eqn

	0	1	2	3	4	5	6	7	8	9
	2	3	1	4	0	-1	2	-2	10	8
even	2	2	3	3	3	3	5	5	15	15
odd	0	3	3	7	7	6	6	4	4	12

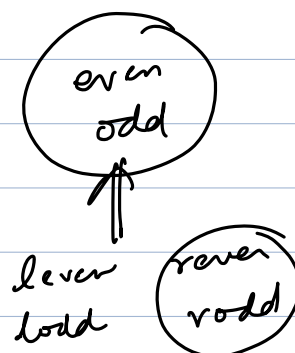


leftsum = 0
rightsum = totalsum

ls = 1 (rs)
0-0 ~~2-9~~
0-1 ~~3-9~~
0-2 ~~4-9~~
0-3 ~~5-9~~
0-4 6-9

for (i = 0 → i < n)
{ rightsum -= arr[i]
if (ls == rs)
leftsum += arr[i];
}

0 1 2 3 4 5
1 2 3 4 8 10
ls = 0 ls = 1 ls = 3
rs = 28 rs = 27 rs = 25



eq — T.C : $O(N)$
spa — S.C : $O(1)$