Q → Given an integer array with N elements & Q queries.
For each query, calculate sum of all elements from index L to R.

some task on multiple input.

	0	1	2	3	4	5	6	7	8	9
A = [	-3	6	2	4	5	2	8	-9	3	1]

Query → L[], R[] (L[i], R[i]) ✓

Query[Q][2] → (Query[i][0], Query[i][1])

(4, 8) → 9
(3, 7) → 10
(1, 3) → 12

L = [4 3 1]
R = [8 7 3]

Bruteforce →

```
for i → 0 to (Q-1) {
    sum = 0
    for j → L[i] to R[i] {
        sum += A[j]
    }
    print (sum)
}
```

TC = O(Q * N)

scoreboard

Over →	1	2	3	4	5	6	7	8	9	10
runs after i^{th} over	0	2	8	14	29	31	49	65	79	88	97

runs in 7^{th} over → 65 − 49 = 16

runs from 6^{th} to 10^{th} over → score[10] − score[5]
= 97 − 31 = 66

runs scored in 10^{th} over = score[10] − score[9]

= 97 − 88 = 9

runs scored from 3^{rd} to 6^{th} over = score[6] − score[2]

= 49 − 8 = 41

runs from 4^{th} to 9^{th} over = score[9] − score[3]

= 88 − 14 = 74

Prefix Sum

	0	1	2	3	4
A =	-3	6	2	4	5
P =	-3	3	5	9	14

P[i] = A[0] + A[1] + A[2] + ... + A[i]

P[0] = A[0]

P[i] = P[i-1] + A[i] ✓

	0	1	2	3	4	5
A =	10	32	6	12	20	1
P =	10	42	48	60	80	81

A[0] + A[1] + A[2]

A[0] + A[1] + A[2] + A[3]

```
P[0] = A[0]
for i ⟶ 1 to (N-1) {
    P[i] = P[i-1] + A[i]
}
```

TC = O(N) SC = O(N)

	0	1	2	3	4
A =	-3	6	2	4	5
P =	-3	3	5	9	14

L = 2, R = 2 } P[2] − P[1] = 5 − 3 = 2

L = 1, R = 3 } P[3] − P[0] = 9 − (−3) = 12 ✓

L = 0, R = 3 } P[3] = 9

A[1] + A[2] + A[3]

P[3] → ~~A[0]~~ + A[1] + A[2] + A[3]

- P[0]

```
for i ⟶ 0 to (Q-1) {
    sum = 0
    l = L[i]      r = R[i]                    TC = O(Q)
    if (l == 0)   print (P[r])
    else          print (P[r] - P[l-1])
}
```

Total TC = O(N + Q) SC = O(N) P[N]

convert input array into prefix sum?

```
for i ⟶ 1 to (N-1) {
    A[i] = A[i-1] + A[i]
}
```

SC = O(1)

	0	1	2	3	4
A =	-3	~~6~~ 3	~~2~~ 5	~~4~~ 9	~~5~~ 14

Q → Given an integer array & Q queries.
For every query find the sum of all even index elements from L to R.

	0	1	2	3	4	5
A =	2	3	1	6	4	5

Query

(1, 3) ⟶ 1

(2, 5) ⟶ 1 + 4 = 5

```
          0   1   2   3   4   5
A = [ 2   3   1   6   4   5 ]
P = [ 2   2   3   3   7   7 ]
```

P[0] = A[0]

→ if (i % 2 == 0) P[i] = P[i-1] + A[i]
  else P[i] = P[i-1]

```
          0   1   2   3   4
A = [ 2   4   3   1   5 ]
P = [ 2   2   5   5   10 ]
```

```
for i → 0 to (Q-1) {
    sum = 0
    l = L[i]    r = R[i]
    if (l == 0)    print (P[r])
    else    print (P[r] - P[l-1])
}
```

Total TC = O(N + Q) SC = O(N) → O(1)

update A → P

Q → Given an integer array A, 10:40 PM

count the # special index in the array.

Special index are those index removing which, sum of all even index elements is equal to sum of all odd index elements.

```
i   A = [ 4   3   2   7   6   -2 ]
          0   1   2   3   4   5
```

i							odd	even	
0	→		3	2	7	6	-2	8 = 8	✓
1	→	4		2	7	6	-2	8 ≠ 9	×
2	→	4	3		7	6	-2	9 = 9	✓

3 → 4 3 2 6 -2 9 ≠ 4 X

4 → 4 3 2 7 -2 10 ≠ 4 X

5 → 4 3 2 7 6 10 ≠ 12 X

Ans = 2

A = [4 1 ~~3~~ 7 10]

7 10 sum of odd ind = 1 + 10 = 11

A = [2 3 1 ~~4~~ 0 -1 2 -2 10 8]

0 -1 2 -2 10 8 sum of odd = 15

sum of even = 8

sum of odd ind after removing A[i] =

sum of odd ind for index < i → PO[i-1]

+ sum of even ind for index > i → PE[N-1] − PE[i]

(i+1) — (N-1)

sum of even ind after removing A[i] =

sum of even ind for index < i → PE[i-1]

+ sum of odd ind for index > i → PO[N-1] − PO[i]

```
PE[0] = A[0]
for i → 1 to (N-1) {
    if (i % 2 == 0)   PE[i] = PE[i-1] + A[i]
    else              PE[i] = PE[i-1]
}
PO[0] = 0
for i → 1 to (N-1) {
    if (i % 2 == 1)   PO[i] = PO[i-1] + A[i]
    else              PO[i] = PO[i-1]
}
```

```
ans = 0
for i ⟶ 0 to (N-1) {
    if (i == 0) {
        so = PE[N-1] - PE[i]
        se = PO[N-1] - PO[i] ⟶ 0
    }
    else {
        so = PO[i-1] + PE[N-1] - PE[i]
        se = PE[i-1] + PO[N-1] - PO[i]
    }
    if (so == se)  ans++
}
return ans
```

TC = O(N + N + N) = O(N) SC = O(N + N) = O(N)