

Queries \Rightarrow Same task for different input.

Q Given an integer array of size N . &
no. of queries.

Answer

Query \Rightarrow s & e
 \downarrow \downarrow
start end

Print the sum of all elements from the index s to the index e . (Both included)

A: $[-3, 6, 2, 4, 5, 2, 8, -9, 3, 1]$ $N=10$

$Q = 4$

s	e	Sum	$s=e$ $[x, x] \Rightarrow A[x]$ 1 element
1	3	12	
2	7	12	
4	8	9	
0	0	-3	$s=0$ $e=N-1$ N

Solⁿ \Rightarrow Brute force

✓ Queries \Rightarrow Iterate from s to e & calc. the sum.

Code

1) Input (A)
2) Input (Q)

for ($i = 0$; $i < Q$; $i++$) $\rightarrow O(Q)$

Input (s, e);

sum = 0;

for ($j = s$; $j \leq e$; $j++$)

$O(N)$

sum += $A[j]$;

print (sum);

}

T.C. = $O(Q \times N)$

S.C. = $O(1)$

2) Optimise

Cumulative Score

Over \nearrow
Score : 0, 16, 22, 30, 45, 51, 70, 75, 90, 104, 120

$$\begin{aligned} \text{Runs scored in last Over} &\Rightarrow [10, 10] &= \text{Runs}[10] - \text{Runs}[9] \\ & &= 120 - 104 = 16 \end{aligned}$$

$$\begin{aligned} \text{Runs scored in last 6 overs} & &= \text{Runs}[10] - \text{Runs}[4] \\ & &= 120 - 45 = 75 \\ & [5, 10] \end{aligned}$$

$$\begin{aligned} \text{Runs scored in 5th Over} & &= \text{Runs}[5] - \text{Runs}[4] \\ & &= 51 - 45 = 6 \\ & [5, 5] \end{aligned}$$

Score Board \Rightarrow Prefix Sum of runs scored in every over.

$$P[i] = A[0] + A[1] + A[2] \dots A[i]$$

$$A = [0, 1, 2, 3, 4, 5, -2, 1, 9]$$

$$P = [-6, -3, -1, 3, 8, 6, 7, 16]$$

$$P[0] = \text{Sum}[0, 0] = A[0]$$

$$P[1] = \text{Sum}[0, 1] = A[0] + A[1]$$

$$P[2] = \text{Sum}[0, 2] = A[0] + A[1] + A[2]$$

$$P[3] = \text{Sum}[0, 3] = A[0] + A[1] + A[2] + A[3]$$

$$P[3] = P[2] + A[3]$$

⋮

~~for~~

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

Code

$P[N]$

$P[0] = A[0];$

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

$P[i] = P[i-1] + A[i];$

}

T.C. = $O(N)$

$A = [-6, 3, 2, 4, 5, -2, 1, 9]$

$P = [-6, -3, -1, 3, 8, 6, 7, 16]$

$$Q = 3$$

s	e	Sum
1	4	$= P[4] - P[0] = 14$
3	6	$= P[6] - P[2] = 8$
0	5	$= P[5]$

$$\text{Sum}[s, e] = P[e] - P[s-1]$$

\Downarrow
 if $(s == 0) \wedge$
 $\quad \text{Sum} = P[e];$
 $\quad \wedge$

for $(i = 0; i < Q; i++) \wedge \quad // O(Q)$

$\text{Input}(s, e);$
 if $(s == 0) \wedge$
 $\quad \text{print}(P[e]);$
 $\quad \wedge$
 else \wedge
 $\quad \text{print}(P[e] - P[s-1]);$
 $\quad \wedge$
 \wedge

O(1)

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

↓
Prefix
Sum

↓
Answer Q
queries.

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

↓
Prefix Sum.

~~range~~

Prefix Sum \Rightarrow Range Sum

Q

Given an integer array of size N.

Directly

Return true if there exists an equilibrium index in the array.

Equilibrium Index = An index for which

Sum of all elements on left = Sum of all elements on right.

$$A = [0, 1, 3, -6, 5, -2, -1]$$

Sum Left = 0

Sum = 0

$$A: [1, 2, 3, 4, 8, 10]$$

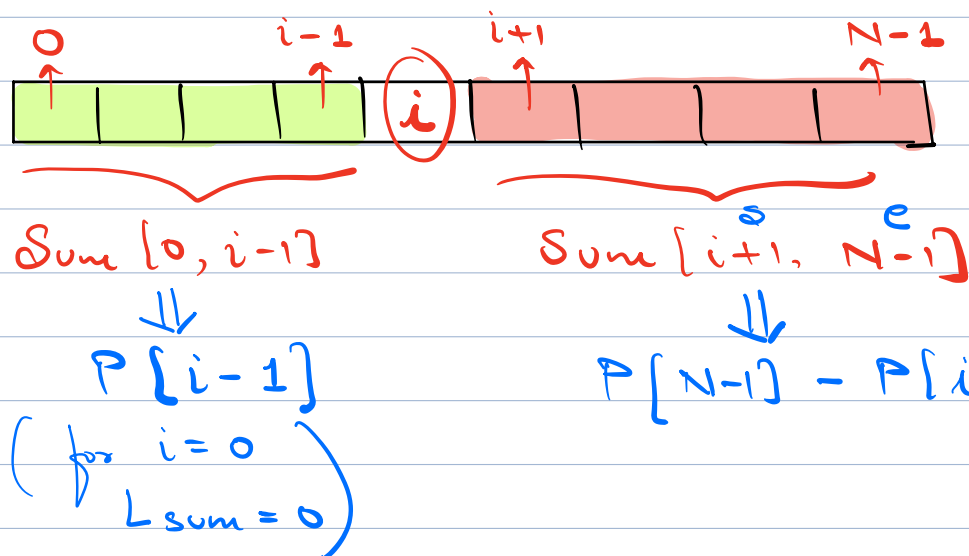
10

10

FI

Solⁿ > Brute force

Index \Rightarrow Check if it is an FI



// $i=0 \Rightarrow Lsum = 0$

$Rsum = Sum[1, N-1]$

if $((P[N-1] - P[0]) == 0) \& \text{return } 0; \&$

for $(i=1; i < N; i++) \&$

if $(P[i-1] == (P[N-1] - P[i])) \&$

return $i;$

$\&$

$\&$

return $-1;$

T.C. = $O(N + N)$

= $O(N)$

S.C. = $O(N)$

H.W.

Solve in S.C. = $O(1)$

w/o modifying I/P array.

T.C. = $O(N)$

Hint

Use sum of all elements.

Q Given an array of size $N \in \mathbb{Q}$ queries.

Query $\Rightarrow s, c$

✓ query \rightarrow Count of even elements from index s to e

Diagram illustrating the merge sort process. The array A is shown with indices 0 to 9. Red arrows indicate the merging of sorted sub-arrays: 0-1, 2-3, 4-5, 6-7, 8-9, and then the final merge of the two halves [0-5] and [6-9].

Q: 3

251

c
7
9
1

count
4
3
0

Solⁿ \Rightarrow Bunte Farbe

\forall Query \Rightarrow iterate from s to e
 \neq count

Code

Input (Q)

```
for (i = 0; i < Q; i++) { // O(Q)
```

```
    Input (s, e);
```

```
    count = 0;
```

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

```
        if (A[j] % 2 == 0) {
```

```
            count++;
```

```
        }
```

```
    }
```

```
    print(count);
```

```
}
```

O(N)

T.C. = $O(Q \times N)$

2) Optimise ??

Prefix Even Count Array (PEC)

PEC[i] \Rightarrow Count of even elements from index 0 to i.

PEC[N];

```
if (A[0] % 2 == 0) {  
    PEC[0] = 1;  
}
```

```
}
```

```
else {
```

```
    PEC[0] = 0;  
}
```

```
}
```

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

```
    if (A[i] % 2 == 0) {
```

```
        PEC[i] = PEC[i-1] + 1;
```

```
    }
```

```
    else {
```

```
        PEC[i] = PEC[i-1];  
    }
```

```
}
```

```
}
```

Count of [s, e] = PEC[e] - PEC[s-1]
Even

\Downarrow

s = 0 \Rightarrow PEC[e]

T.C. = $O(N + Q \times 1)$
= $O(N + Q)$

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

A: $\left[\overset{0}{\underline{5}}, \overset{1}{7}, \overset{2}{\underline{2}}, \overset{3}{\underline{3}}, \overset{4}{\underline{4}}, \overset{5}{\underline{6}}, \overset{6}{\underline{7}}, \overset{7}{\underline{2}}, \overset{8}{\underline{1}}, \overset{9}{\underline{4}} \right]$

P: $\left[0, \underline{0}, 1, 1, 2, 3, 3, 4, 4, 5 \right]$

A: $\left[\overset{0}{\underline{3}}, \overset{1}{\underline{-2}}, \overset{2}{\underline{3}}, \overset{3}{\underline{6}}, \overset{4}{\underline{4}}, \overset{5}{\underline{6}}, \overset{6}{\underline{8}}, \overset{7}{\underline{4}}, \overset{8}{\underline{8}}, \overset{9}{\underline{5}} \right]$

max = ~~3~~ ~~6~~ 8

count = ~~1~~ ~~2~~ ~~1~~ ~~2~~ ~~1~~ 2

A = $\left[2, 4, 6, 8, 1, 3, 5, 7, 9 \right]$

K = 1 $\left[9, 2, 4, 6, 8, 1, 3, 5, 7 \right]$

temp = A[N-1];

QW

for ($i = N-1$; $i > 0$; $i--$) {

$A[i] = A[i-1];$

}

$A[0] = temp;$

Hint : Compare original & final rotated arrays.

$A = [2, 4, 6, 8, 1, 3, 5, 7, 9]$
 $K = 3 [5, 7, 9, 2, 4, 6, 8, 1, 3]$

Hint

Reverse the array