

Agenda

- 1) Prefix sum intro
- 2) Questions based on prefix sum
- 3) Carry forward
- 4) Ag pairs
- 5) Subarray intro

Q. Given $A[]$ and Q queries (every query has L & R). Find sum of elements of array from L to R .

$$A[] = [-3 \ 6 \ 2 \ 4 \ 5 \ 2 \ 8 \ -9 \ 3 \ 1]$$

constraints
 $1 \leq n \leq 10^5$
 $1 \leq q \leq 10^5$

L	R	Sum
0	3	9
1	5	19
4	4	5
5	8	4

0	1
0	3
1	5
2	4
3	8

Q

Brute Force

```
void solve (int []A, int [][]Q) {
    for (int i=0; i<Q.length; i++) {
        int L= Q[i][0];
        int R= Q[i][1];
        int sum=0;
        for (int j=L; j<=R; j++) {
            sum+= A[j];
        }
        System.out.println(sum);
    }
}
```

$$A = [6 \ -3 \ 1 \ 4 \ 2]$$

0	1
1	3
2	4

Q

$$i=0, \ L=1 \ R=3, \ sum=2$$

$$i=1, \ L=2 \ R=4, \ sum=7$$

constraints
 $1 \leq n \leq 10^5$
 $1 \leq q \leq 10^5$

TC: $O(q \cdot n) \longrightarrow \text{TLE}$

SC: $O(1)$

Optimised Idea

7th Over = 65 - 49

$$= 16$$

Given the scores of the 10 overs of a cricket match

2, 8, 14, 29, 31, 49, 65, 79, 88, 97

How many runs were scored in just 7th over?

2	8	14	29	31	49	65	79	88	97
1	2	3	4	5	6	7	8	9	10

How many runs were scored from 6th to 10th over(both included)?

2	8	14	29	31	49	65	79	88	97
1	2	3	4	5	6	7	8	9	10

runs till 10th over -

runs till 5th over

$$= 97 - 31 = 66$$

How many runs were scored in just 10th over?

$$97 - 88 = 9$$

How many runs were scored from 3rd to 6th over(both included)?

2	8	14	29	31	49	65	79	88	97
1	2	3	4	5	6	7	8	9	10

runs scored till 6th -

run scored till 2nd

$$= 49 - 8 = 41$$

Intro to prefix sum

$$A[] = [-3 \ 6 \ 2 \ 4 \ 5 \ 2 \ 8 \ -9 \ 3 \ 1]$$

$$PS[] = [-3 \ 3 \ 5 \ 9 \ 14 \ 16 \ 24 \ 15 \ 18 \ 19]$$

$$\downarrow$$

$PS[i]$ = sum of elements from 0 to i or
sum of elements till i^{th} index

$$* PS[i] = PS[i-1] + A[i]$$

$$A[] = [-3 \ 6 \ 2 \ 4 \ 5 \ 2 \ 8 \ -9 \ 3 \ 1]$$

$$PS[] = [-3 \ 3 \ 5 \ 9 \ 14 \ 16 \ 24 \ 15 \ 18 \ 19]$$

L R Sum

$$1 \quad 5 \quad PS[5] - PS[0] = 16 - (-3) = 19$$

if ($L == 0$) {

$$sum = PS[R];$$

$$5 \quad 8 \quad PS[8] - PS[4] = 18 - 14 = 4$$

3

$$0 \quad 3 \quad PS[3] = 9$$

edge {

$$4 \quad 4 \quad PS[4] - PS[3] = 14 - 9 = 5$$

$$sum = PS[R] - PS[L-1];$$

5

```
void solve ( int [ ] A, int [ ] [ ] Q ) {
```

TC: $O(n+q)$

|| construct prefix sum array

```
int [ ] PS = new int [ A.length ];
```

SC: $O(n)$

```
PS [ 0 ] = A [ 0 ] ;
```

```
for ( int i = 1; i < PS.length; i ++ ) {  
    PS [ i ] = PS [ i - 1 ] + A [ i ];  
}
```

↳ creation of
PS []

|| find the ans for each query

```
for ( int i = 0; i < Q.length; i ++ ) {  
    int L = Q [ i ] [ 0 ];  
    int R = Q [ i ] [ 1 ];  
    int sum = 0;  
    if ( L == 0 ) {  
        sum = PS [ R ];  
    }  
    else {  
        sum = PS [ R ] - PS [ L - 1 ];  
    }  
    System.out.println ( sum );  
}
```

A: [6 5 1 -4 3 7]
0 1 2 3 4 5

PS: [6 11 12 8 11 18]
0 1 2 3 4 5

Q: [[0, 1], [2, 5], [0, 4]]
0 1
2 5
0 4
→ 7
→ 11

i = 0, L = 2 R = 5 $PS[5] - PS[1]$
 $18 - 11 = 7$

i = 1, L = 0 R = 4 $PS[4] = 11$

3

Q. Given an $A[]$ and Q queries with L, R . For every query, return sum of all even indexed elements from L to R .

$$A[] = [2_0 \ 3_1 \ 1_2 \ 6_3 \ 4_4 \ 5_5]$$

1) Brute force :

L	R	sum
0	4	7
1	5	5
2	4	5
2	2	1

TC: $O(q \times n)$

SC: $O(1)$

2) Optimised logic

$$A[] = [2_0 \ 3_1 \ 1_2 \ 6_3 \ 4_4 \ 5_5]$$

$$PS[] = [2_0 \ 2_1 \ 3_2 \ 3_3 \ 7_4 \ 7_5]$$

↑

$PS[i] \Rightarrow$ sum of even indexed values from 0 to i

```

if (i < 2) {
    PS[i] = PS[i-1];
}
else {
    PS[i] = PS[i-1] + A[i];
}

```

L	R	sum
0	4	$PS[4] = 7$
1	5	$PS[5] - PS[0] = 5$
2	4	$PS[4] - PS[1] = 5$
2	2	$PS[2] - PS[1] = 1$

```
void fun ( int [ ] A, int [ ] [ ] Q ) {
```

TC: $O(n+q)$

```
    || construct prefix sum array
```

SC: $O(n)$

```
    int [ ] PS = new int [ A.length ];
```

```
    PS [ 0 ] = A [ 0 ] ;
```

↳ creation of
PS []

```
    for ( int i = 1; i < PS.length; i++ ) {
```

```
        if ( i - 1 != 0 ) {
```

```
            PS [ i ] = PS [ i - 1 ] ;
```

```
        }
```

```
        else {
```

```
            PS [ i ] = PS [ i - 1 ] + A [ i ] ;
```

```
        }
```

```
    }
```

```
    || find the ans for each query
```

```
    for ( int i = 0; i < Q.length; i++ ) {
```

```
        int L = Q [ i ] [ 0 ] ;
```

```
        int R = Q [ i ] [ 1 ] ;
```

```
        int sum = 0 ;
```

```
        if ( L == 0 ) {
```

```
            sum = PS [ R ] ;
```

```
        }
```

```
        else {
```

```
            sum = PS [ R ] - PS [ L - 1 ] ;
```

```
        }
```

```
        System.out.println ( sum ) ;
```

```
}
```

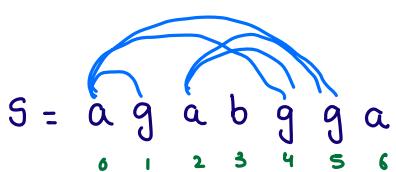
```
}
```

Q. Given a string s of lowercase characters, return the count of pairs (i, j) such that $i < j$ and $s[i]$ is 'a' and $s[j]$ is 'g'.



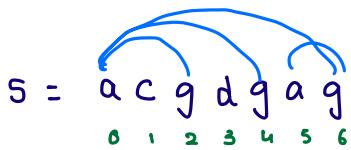
ans = 3

const. $1 \leq n \leq 10^7$

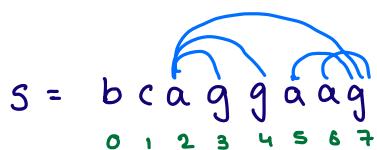


ans = 5

Find count of pairs in which
'a' is before 'g'



ans = 4



ans = 4

```
int countagPair (String str) {
```

```
    int n = str.length();
```

```
    int ans = 0;
```

TC: $O(n^2)$ TLE

```
    for (int i = 0; i < n; i++) {
```

SC: $O(1)$

```
        if (str.charAt(i) == 'a') {
```

```
            int temp = 0;
```

```
            for (int j = i + 1; j < n; j++) {
```

```
                if (str.charAt(j) == 'g') {
```

```
                    temp++;
```

```
                }
```

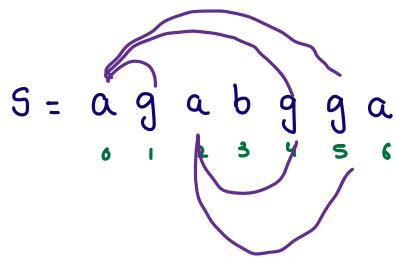
```
            }  
            ans += temp;
```

```
        }
```

```
    }
```

3

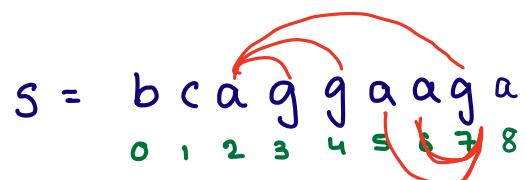
→ Optimised



$$cg = 0 \times 2 \times 3$$

$$ans = 0 + 2 + 3$$

```
int countagpairs (string str) {  
    int n = str.length();  
    int cg = 0, ans = 0;  
    for (int i = n - 1; i >= 0; i--) {  
        if (str.charAt(i) == 'g') {  
            cg++;  
        }  
        else if (str.charAt(i) == 'a') {  
            ans += cg;  
        }  
    }  
    return ans;  
}
```



$$cg = 0 \times 2 \times 3$$

$$ans = 0 + 0 + 1 + 1 + 3 = 5$$

Intro to subarrays

↳ continuous part of an Array

i) continuous part of an Array

ii) single element is also a subarray & complete array is also a subarray

iii) Order matters

$$A = \{ 2 \ 3 \ 8 \ 5 \ 7 \ 10 \}$$

valid subarray or not?

{ 2 3 5 10 } X

{ 3 8 5 7 } ✓

{ 7 } ✓

{ 3 8 7 5 } X

$A[] = \{ 2, 4, 1, 6, -3, 7, 8, 4 \}$

Which of the following is a valid subarray?

20 users have participated

- | | | | |
|---|--------------|-----------|----|
| A | {1, 6, 8} | X | 5% |
| B | {1, 4} | X | 0% |
| C | {6, 1, 4, 2} | X | 0% |
| ✓ | D | {7, 8, 4} | ✓ |

$A = \{ 2, 3, 8, 5, 7, 10 \}$ s, e

1, 4 $\rightarrow \{ 3, 8, 5, 7 \}$

2, 2 $\rightarrow \{ 8 \}$