Amazon | Paypal

Q1. Given a string s, count no. of pairs s.t

1) i<j
2) s[i] = 'a'
3) s[j] = 'g'

s =

0	1	2	3	4	5
a	l	h	g	a	g

(0,3) ✓
(0,5) ✓
(4,5) ✓

(3,0) ✗ (g = a)
(4,3) ✗ (a = g)

0	1	2	3	4	5	6
c	a	g	d	g	a	g

(1,2) ✓
(1,4) ✓
(1,6) ✓
(5,6) ✓

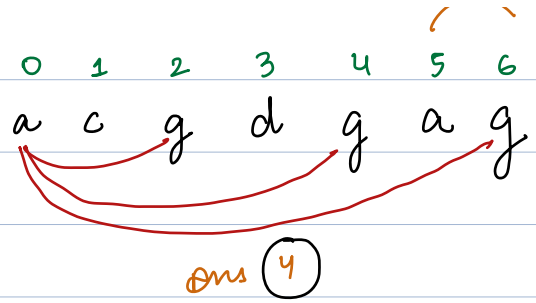
ans: 4

0	1	2	3	4	5	6	7	8
a	d	g	a	g	a	g	f	g

(0,2)
(0,4)
(0,6)
(0,8)
(3,4)
(3,6)
(3,8)
(5,6)
(5,8)

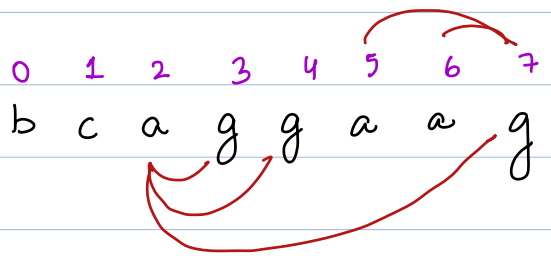
Total (9) pairs

QUIZ 1



Ans (4)

QUIZ 2



Ans : (5) =

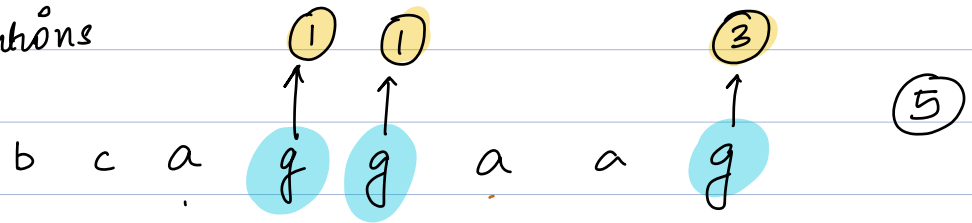
Brute Force

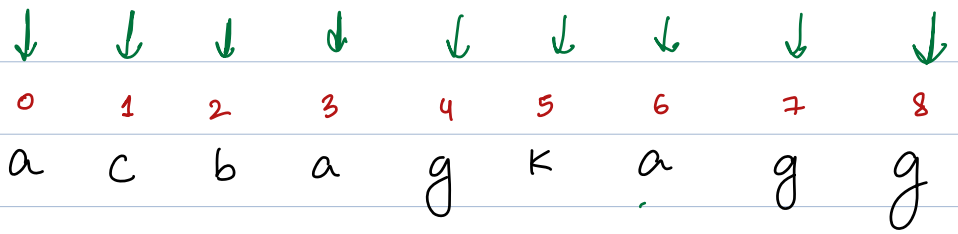
```
count = 0
for( i = 0 ; i < N ; i++ ) {
    if( s[i] == 'a' ) {
        for( j = i + 1 ; j < N ; j++ ) {
            if( s[j] == 'g' ) {
                count++
            }
        }
    }
}
return count
```

TC → $O(N^2)$

SC → $O(1)$

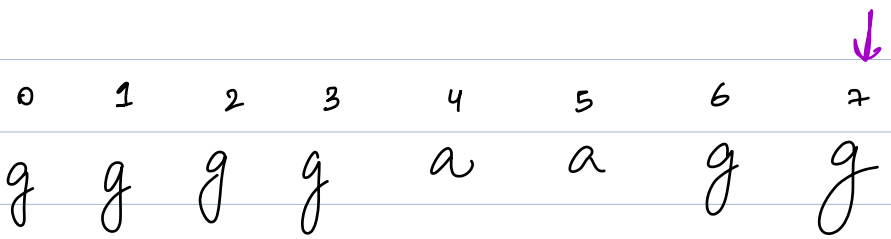
Observations





count
= ~~0~~
~~1~~
~~2~~
3

ans = 2 + 3 + 3
= 8



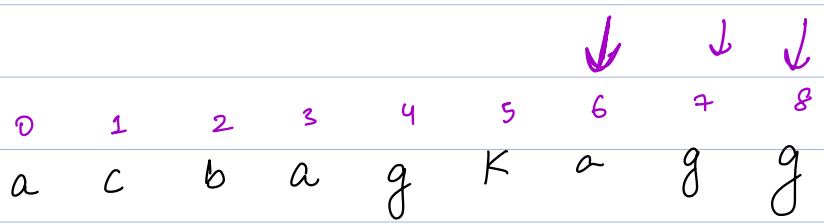
count
= ~~0~~
~~1~~
2

ans = ~~0~~ ~~2~~ 4

```
count_a = 0
ans = 0
for( i = 0 ; i < N ; i++ ){
    if ( s[i] == 'a' ){
        count_a++
    }
    else if ( s[i] == 'g' ){
        ans = ans + count_a
    }
}
return ans.
```

TC → O(N)
SC → O(1)

TODO



count_g = ~~0~~

~~1~~

2

ans = ~~0~~

ans = ans + count

2

Q2. Leaders in the array.

Given an array A of N integers. Find all the leaders.

A leader is strictly greater than all ele to its right

A: [1, 2]

leaders = 1

A: [16, 17, 17, 4, 3, 5, 2]

leaders = ~~1~~ ~~2~~ 3

QUIZ

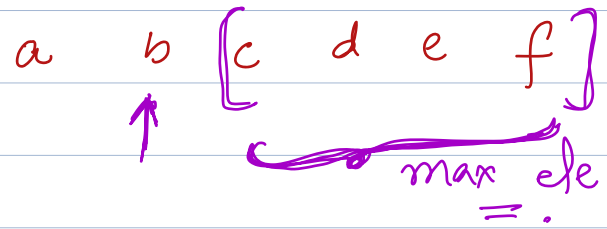
ar[7]: {8, -2, 4, 7, 6, 5, 1}

leaders = ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5

ar[6]: {10, 7, 9, 3, 2, 4}

leaders = ~~X~~ ~~X~~ 3

Observation



$b > e$ ✓

$e > c, d \& f$

{16, 17, 17, 4, 3, 5, 2}

leaders = ~~1~~ ~~2~~ (3) =

max_ele = ~~2~~ ~~5~~ 17

Write the code by yourself.

Que: Closest Min Max.

Given an array A of N integers.
Find smallest continuous part of array s.t it contains both the min & max value of the array.

	0	1	2	3	4	5	6	7	8	9
arr[]:	1	2	3	1	3	4	6	4	6	3

max → 6
min → 1

len = 4

{2, 2, 6, 4, 5, 1, 5, 2, 6, 4, 1}

Max → 6
Min → 1

len = 3

{1, 6, 4, 2, 7, 7, 5, 1, 3, 1, 1, 5}

max → 7
min → 1

Ans: 3

9:30 Break + thinking on solution

Observations :

(1) min & max will be the corner ele of that segment.

(2) There will be exactly one MIN & MAX

. max _ _ max _ min _ min _ _ max

// for every min, look for closest max on right

// for every max, look for closest min on right

(1) Find Amin & Amax before brand

len = +∞ / INT_MAX

(2) for (i=0; i<N; i++) {

    if (A[i] == Amin) {

```
        for ( j = i+1 ; j<N ; j++) {
            if ( A[j] == Amax ) {
                len = min ( len, j-i+1 )
                break
            }
        }
    }
    else if( A[i] == Amin ) {
        for ( j = i+1 ; j<N ; j++) {
            if ( A[j] == Amin ) {
                len = min ( len, j-i+1 )
                break
            }
        }
    }
}
return len
```

max → 7

min → 1

0	1	2	3	4	5	6	7	8	9	10	11
1	6	4	2	7	7	5	1	3	1	1	5

len = ~~INT_MAX~~
~~5~~
~~4~~
3

i j

QUIZ

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

min → 8
max → 8
len → 1

len = 2

j-i+1
0-0+1
= 1

min → 1
max → 5

TC

1	1	1	1	1	1	5

$N + (N-1) + (N-2) + (N-3) + \dots 1$

$1 + 2 + 3 + \dots N$

$\frac{N(N+1)}{2} \rightarrow O(N^2)$

Observation for optimized Solution

Amin Amax

Amin = 1 Amax = 7

0	1	2	3	4	5	6	7	8	9	10
7	2	1	1	2	7	5	3	1	7	6

last_max = ~~-1~~ ~~0~~ ~~5~~ 9

last_min = ~~-1~~ ~~2~~ ~~3~~ 8

len = 2-0+1

= ~~3~~

2

YES↑ NO↓.

Amin = 1 Amax = 6

0	1	2	3	4	5	6	7	8	9	10
2	1	6	6	2	1	5	2	6	4	1

last_min = ~~-1~~ ~~1~~ ~~5~~ 10

last_max = ~~-1~~ ~~2~~ ~~3~~ 8

len = 2

steps

1. Find Min → O(N)
2. Find Max → O(N)
3. Iterated again to find the ans → O(N)

Code

```
Amax = A[0]
Amin = A[0]
for(i=0; i<N; i++){
    if (A[i] > Amax){
        Amax = A[i]
    }
    else if (A[i] < Amin){
        Amin = A[i]
    }
}
last_min = -1
last_max = -1
for (i=0; i<N; i++){
```

```
if ( A[i] == Amin ) {
    last_min = i
    if (last_max != -1){
    |    len = min (len, i - last_max + 1)
    }
}
if ( A[i] == Amax) {
    last_max = i
    if (last_min != -1){
    |    len = min (len, i - last_min + 1)
    }
}
}
return len
```

can also be written as

```
if(A[i] = Amin){
|   last_min = i
}
if(A[i] = Amax){
|   last_max = i
}
if( last_min != -1 && last_max != -1){
```

```
    len = min(len, abs(last_min - last_max) + 1)
}
```