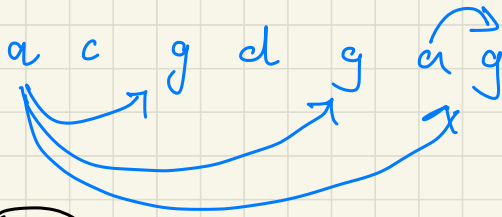


Lecture 5 : Carry Forward



find (i, j)
pairs s.t. $i < j$

(10)

→ 4 pairs

Brute Force

TC: $O(N^2)$

SC: $O(1)$

```
cnt = 0
for i in range(len(l)):
    for j in range(i+1, len(l)):
        if l[i] == 'a' and l[j] == 'g':
            cnt += 1
```

return cnt

Optimized

carries forward the num of "g"s
adds to the count for
subsequent "a"s

```
ans = 0, count = 0
for i in range(len(l)-1, -1, -1):
    if l[i] == 'g':
        count += 1
    if l[i] == 'a':
        ans += count
```

return ans

TC: $O(N)$

SC: $O(1)$

11

Given an array of N elements, count the no. of leaders in array if its strictly greater than all the element on its right

15 -1 7 2 5 4 2 3
 $\Rightarrow 5$

```
cnt = 1
```

```
max = A[-1]
```

```
for i in range(len(A)-2, -1, -1):
```

```
    if A[i] > max:
```

```
        cnt += 1
```

```
        max = A[i]
```

```
return cnt
```

⑫ Find the closest minmax

length of smallest subarray which has at least one occurrence of min & max element

```
ans = len(A)
min-index = -1
min-value = min(A)
max-value = max(A)
```

```
for i in range(len(A)-1, -1, -1):
```

```
    if A[i] == max-value and min-index != -1:
```

```
        ans = min(ans, min-index - i + 1)
```

```
    if A[i] == min-value
```

```
        min-index = i
```

```
max-index = -1
```

```
for i in range(len(A)-1, -1, -1):
```

```
    if A[i] == min-value and max-index != -1:
```

```
        ans = min(ans, max-index - i + 1)
```

```
    if A[i] == max-value:
```

```
        max-index = i
```

```
return ans
```

(13) Given N bulbs in an initial state, find minimum number of switches needed to turn ON all bulbs

a switch also changes state of all bulbs on the right
[0 1 0 1] \Rightarrow [1 0 1 0] \Rightarrow [1 1 0 1]
 \uparrow \uparrow

```
start_idx = None
for i in range(len(A)):
    if A[i] == 0:
        start_idx = i
        break
```

```
if start_idx == None:
    return 0
```

```
min_switches = len(A) - start_idx
```

```
for i in range(start_idx, len(A)-1):
    if A[i] == A[i+1]:
        min_switches -= 1
```

```
return min_switches
```

④ Given an integer element A.

Decide whether its possible to divide the array into one or more subarrays of even length st. the first and last element of all subarrays is even

$[2, 4, 8, 6] \Rightarrow [2, 4], [8, 6]$

$[2, 4, 8, 7, 6] \Rightarrow \times$

$n = \text{len}(A)$

if $n \% 2 == 1$:
return False

elif $A[0] \% 2 == 1$ or $A[n-1] \% 2 == 1$:
return False

else:
return True