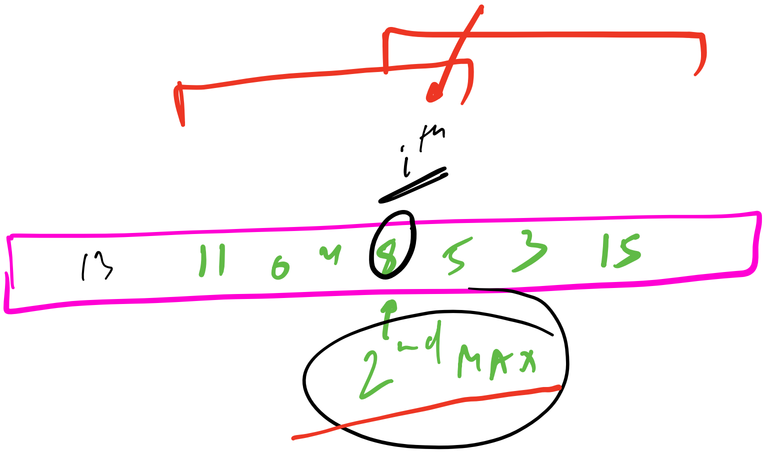
Q All S.A

find the MAX of XOR of MAX & 2nd MAX over all S.As

NOTE: S.As size ≥ 2

Idea:



Consider ith element as 2nd MAX

find all possible MAX values on left & right.

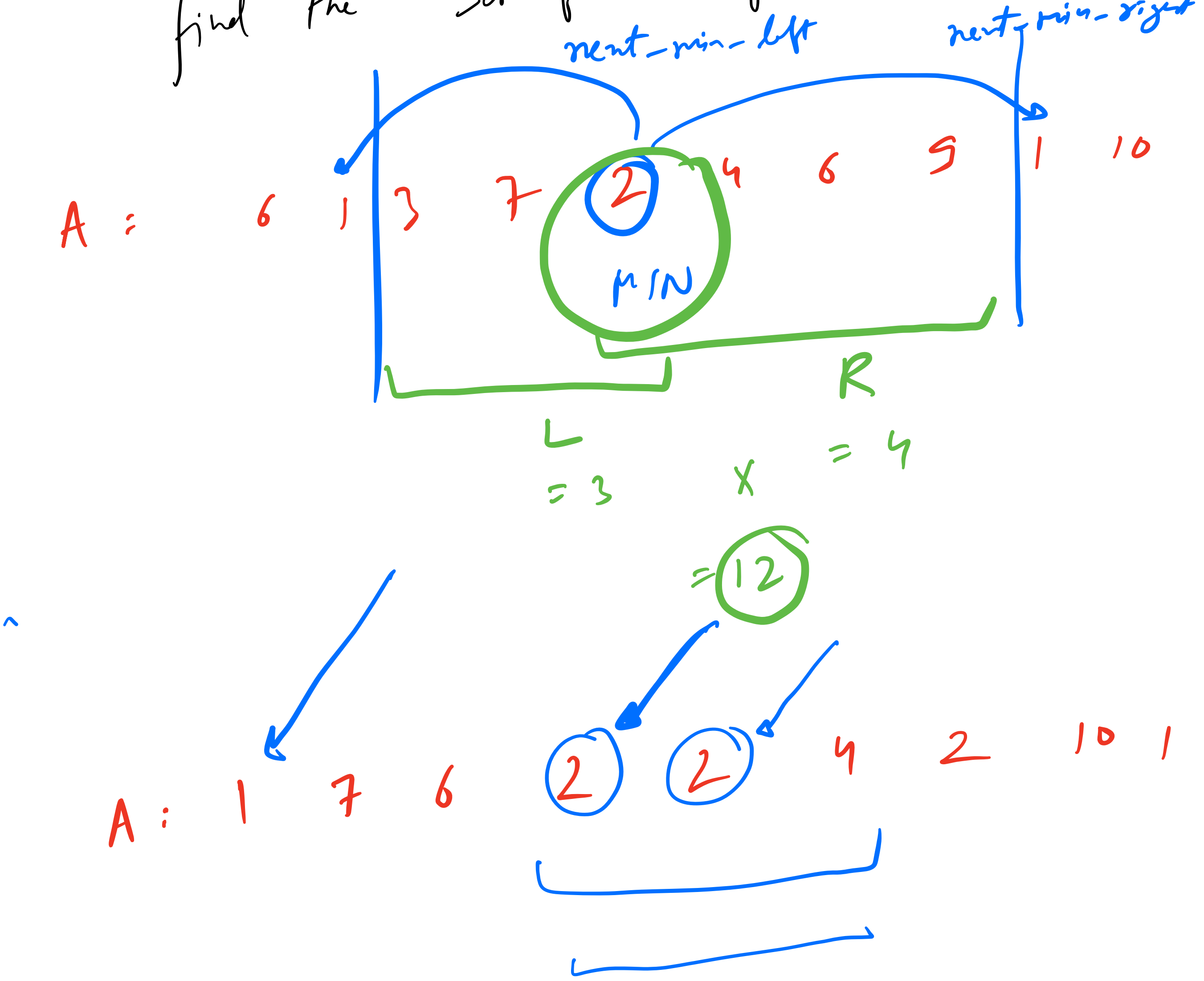
stack!

O(N)

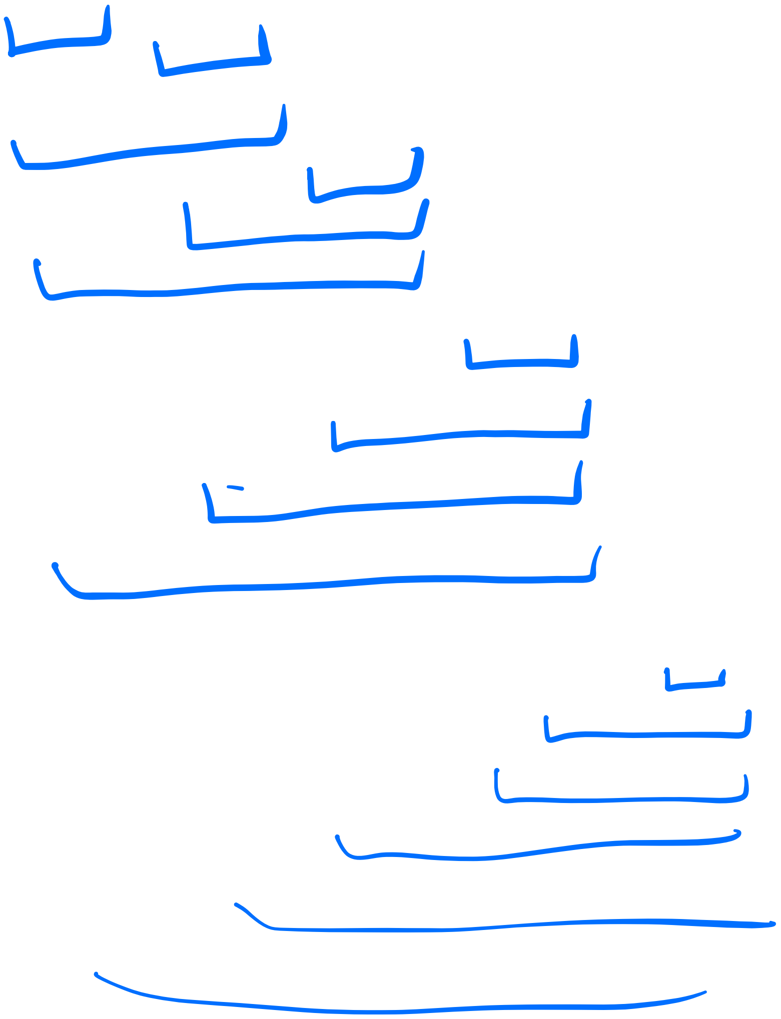
Q MAX & MIN

Given an Array A[].
value of a S.A is max(S.A) − min(S.A)

find the Sum of values of all SAs.



A : 1 2 2 2 2 2 1



strict

not so strict

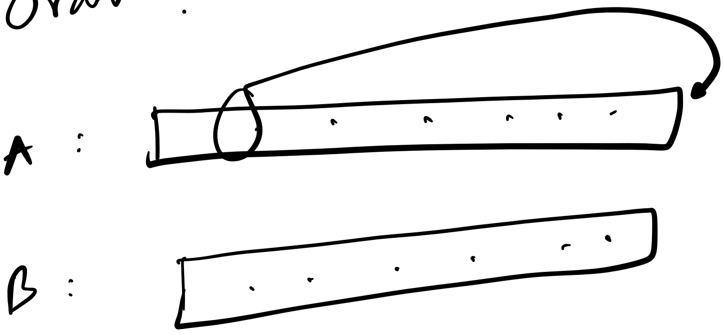
< 2 MIN ≤ 2

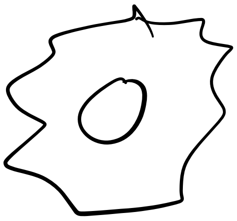
> 2 MAX ≥ 2

Task Scheduling

CPU , N tasks

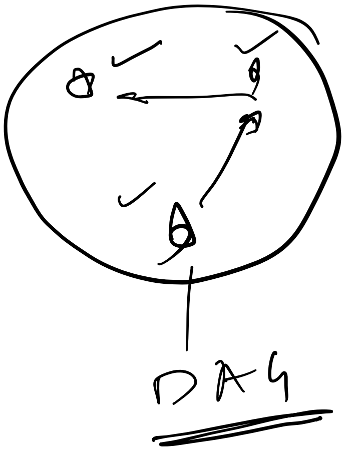
Order !





A : [2 3 1 5 4]

B : ~~1~~ ~~3~~ ~~5~~ ~~4~~ ~~2~~



Queue :

~~2~~ ~~3~~ ~~1~~ ~~5~~ ~~4~~ ~~2~~ ~~3~~ ~~5~~ ~~4~~ ~~2~~

cnt ++

$1 <= N <= 1000$

N^2

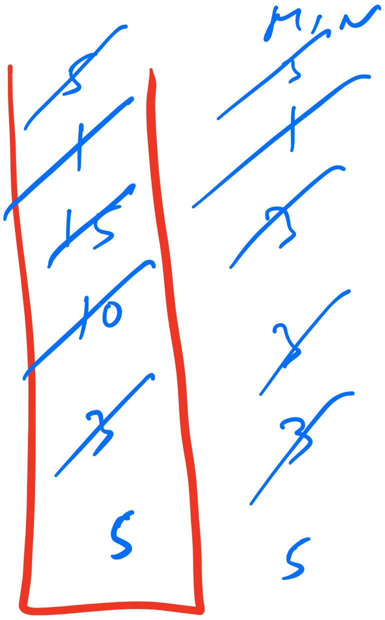
Q Implement a MIN stack!

1) push(x)

2) pop()

3) findMin()

⋮



Idea: Maintain EXTRA stack for MIN till now!

st, minSt

```
getMin() {
    ret minSt.top();
}
```

```
push(x) {
    st.push(x);
    minSt.push(min(x,
                   minSt.top()))
}
```

```
pop() {
    st.pop();
    minSt.pop();
}
```

$\boxed{TC/op \to O(1)}$

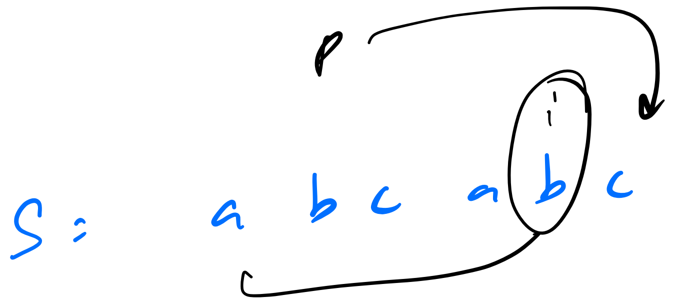
$\boxed{SC \to O(N)}$

first Non-repeating char

Q Given a string s.
find the first non-repeating char for all prefixes!

S: a b c a b c g a

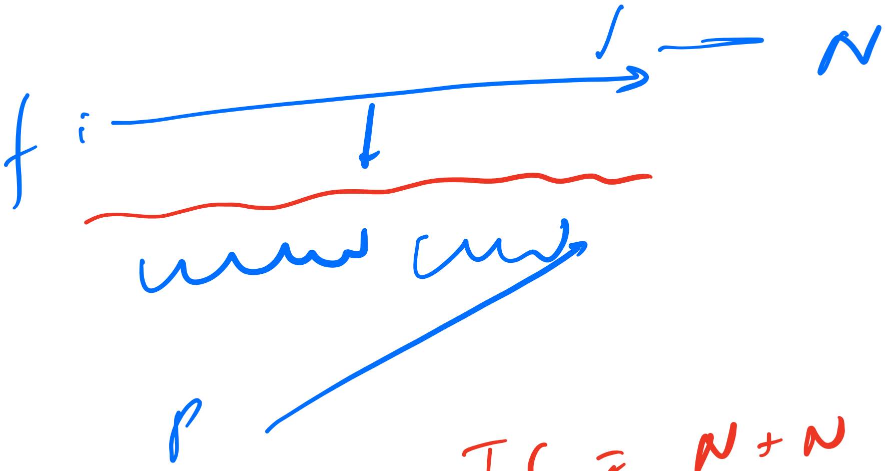
a a a b c # g g



HM <char, int>

```
for(i=0; i<n; i++) {  → N
  hm[s[i]]++;
  while( p <= i &&
             hm[s[p]] > 1)  → N
        p++;
  if( p <= i) print (s[p]);
  else print('#');
```

H.M	
a	3
b	2
c	2
g	1



K → # of chars

$TC = N + N$

$\boxed{TC = O(N)}$

$\boxed{SC = O(\cancel{K})}$

(1)