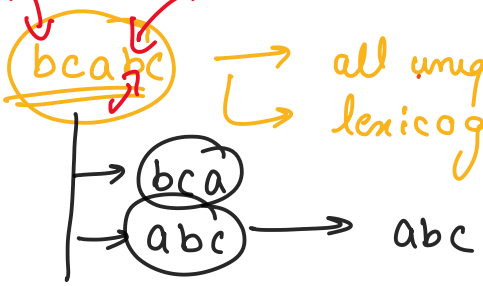
Friday, 10 January 2020 9:07 PM

Stacks

Q: You are given a string → bcabc

→ all unique characters

→ lexicographically smallest



visited

- cbacdcb

$c_1\ c_2 \dots\dots\ c_i\ c_{i+1}\ c_{i+2} \dots\dots$

→ for any index i, let's say the char is c_i

if the char at $(i+1)^{th}$ index is c_{i+1}

if $c_{i+1} > c_i$ & c_{i+1} is still not visited

→ Now for any char c_i, if c_{i+1} is smaller than c_i

$c_i > c_{i+1}$

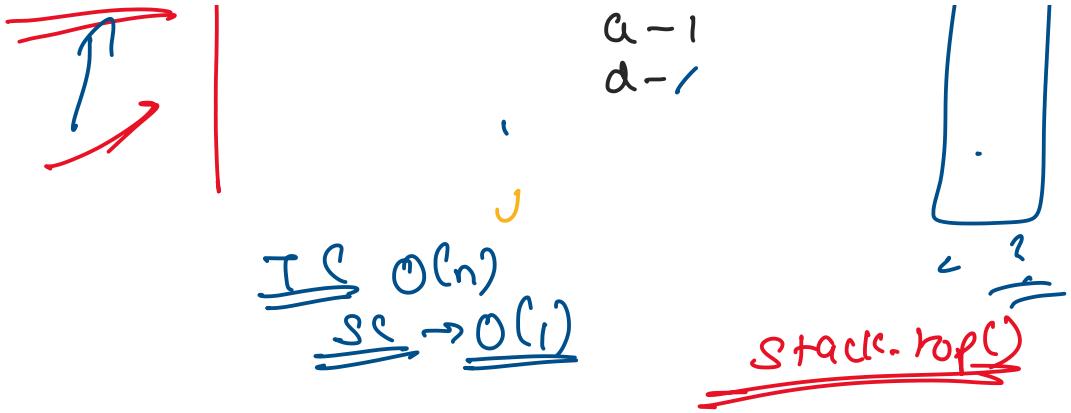
we will check that if c_i is going to be encountered later in my string after c_i or not

Can we use hashing?

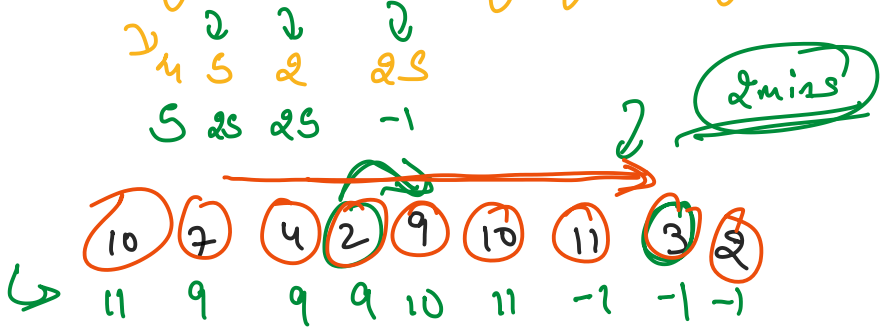
ex cbacdcbcd freq map

↳ c - 4
b - 2

visited



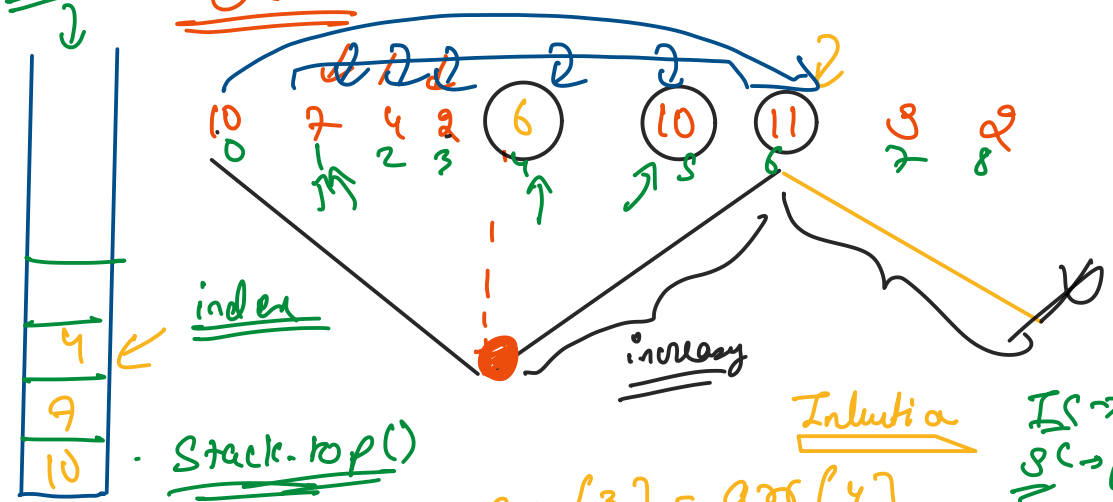
Q: You are given an array of integers



Just larger → first greater number on the right

Brute force

$O(n^2)$

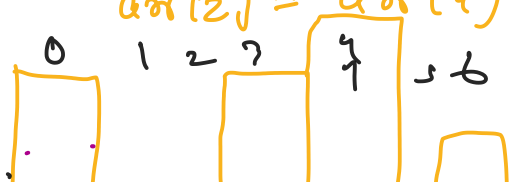


Intuition

TC → O(n)
SC → O(n)

ans[3] = arr[4]

arr[2] = arr[4]



$O(n^2)$

Q: histogram

find the largest rectangular area in the histogram