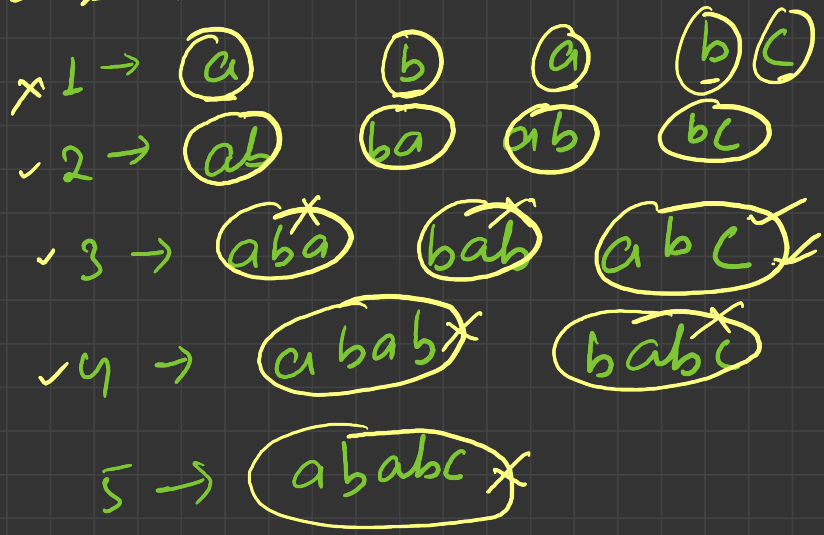
# Longest Substring Without Repeating Characters

① abcdecbed

② ababc

O(N²)

len = ~~1~~ ~~2~~ 3 a b a bc



$O(N^3) \rightarrow O(N^2)$

len = ~~0~~ ~~X~~ ~~X~~ 3

a b a b c

a b a ×

b a b ×

a b c ✓

b c

c

$O(N^2)$

$\Downarrow$

$O(N)$

a b a b c

len : ~~0~~ ~~1~~ ~~2~~ 3

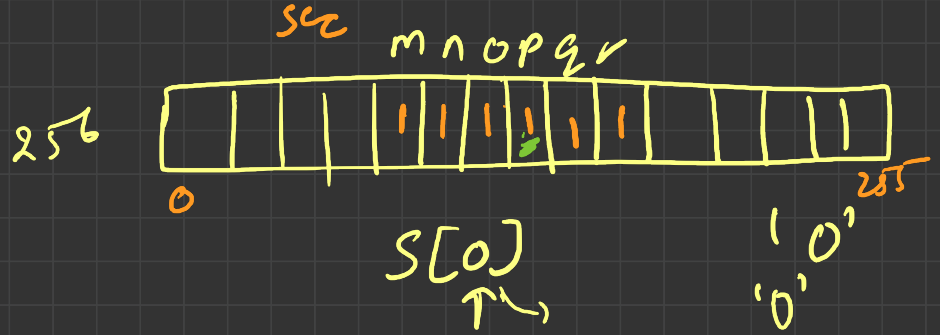
---

str : — m n o p q o n q m p r

first

len : ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ 6

256




O(N)

256

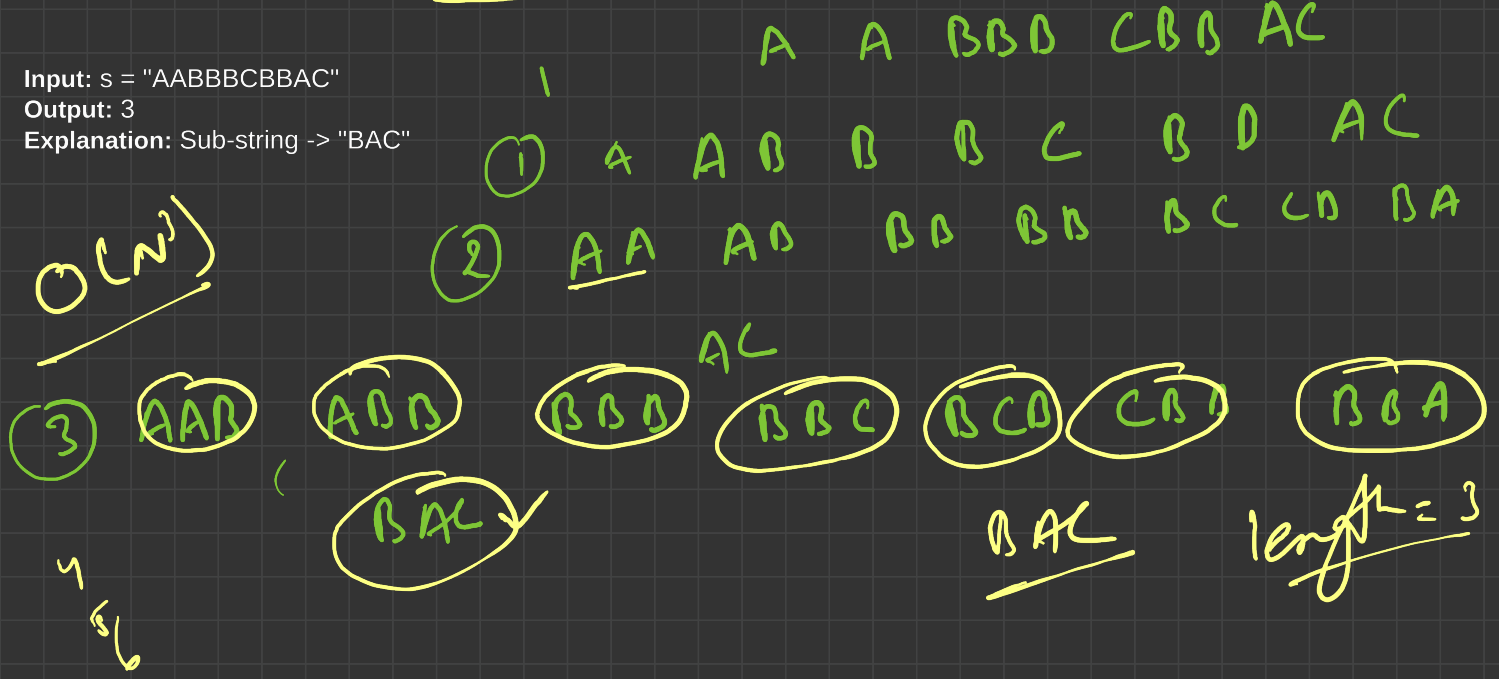
```
vector <bool> count (256, 0)
int first = 0, second = 0 , length = 0;
while(second < s.size()) {
    while (count [s[second]]) {
        count [s[ first]] = 0;
        first++;
    }
    count [s[second]] = 1;
    len = max(len, sec - first + 1)
    second++;
    }
    return length;
```

sec - first

O(N)
O(256)
O(1)

# Smallest distinct window

**Input:** s = "AABBBCBBAC"
**Output:** 3
**Explanation:** Sub-string -> "BAC"



first

Str - A A B B C B R A C

sec

O(256)
O(1)

✓O(N)

diff = 3

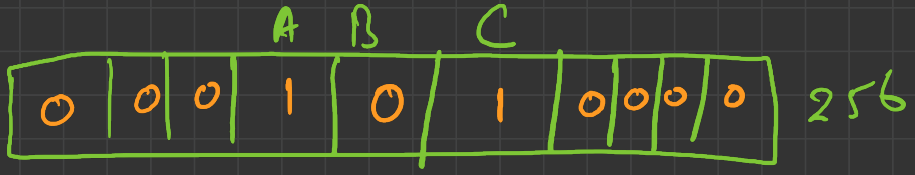
= 2

= 1

= 0

A
B
C

len = 10 6 8 4 3




count =