Aa

HEAP...

Video-13

(Priority Queue)

codestorywithMIK

Leetcode - 767

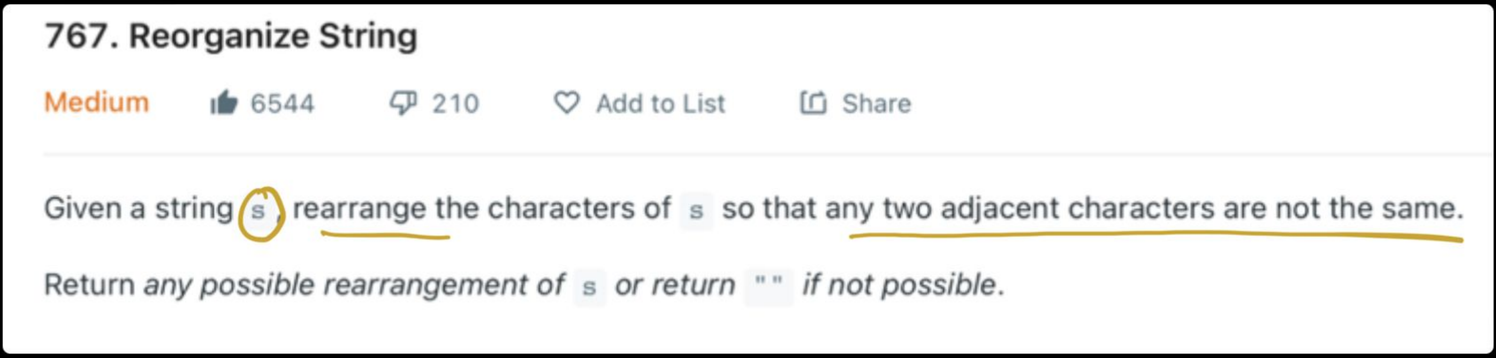
~~Medium~~

Easy

(2 Approaches)

Reorganise String...

Company :- amazon TESLA



767. Reorganize String

Given a string s, rearrange the characters of s so that any two adjacent characters are not the same.

Return *any possible rearrangement of* s *or return* "" *if not possible.*

Examples :- "aab" → aba ✓

"aaab" →

Approach-1

Why Heap ?

Why Heap ?

(•) The most I am worried about = char with max Freq.

aaabbbbcdd

a→ ~~3~~ ~~2~~ ~~1~~ 0
b→ ~~3~~ ~~2~~ ~~1~~ 0
c→ ~~1~~ 0
d→ ~~2~~ ~~1~~ 0

result = 'ababd
abcd'

(•) Store frequency. → (if freq. > (n+1)/2 return "");

(•) max-heap ({freq., char})

"aab" , n = 3

a b a

≤2

3/2 = ⌈1.5⌉ = 2

"$\underline{a}\underline{a}b$" , $\underline{n = 3}$

$\underline{a}$ $\underline{b}$ $\underline{a}$

$\underline{=2}$

$3/2 = \lceil 1.5 \rceil = \boxed{2}$

$\underline{\lceil n/2 \rceil = (n+1)/2}$

"aaa" $n = 3$

$\underline{a} = 3$ ✓ $(n+1)/2 = 2$

_ _ _ ✗

"aaab" $n = 4$

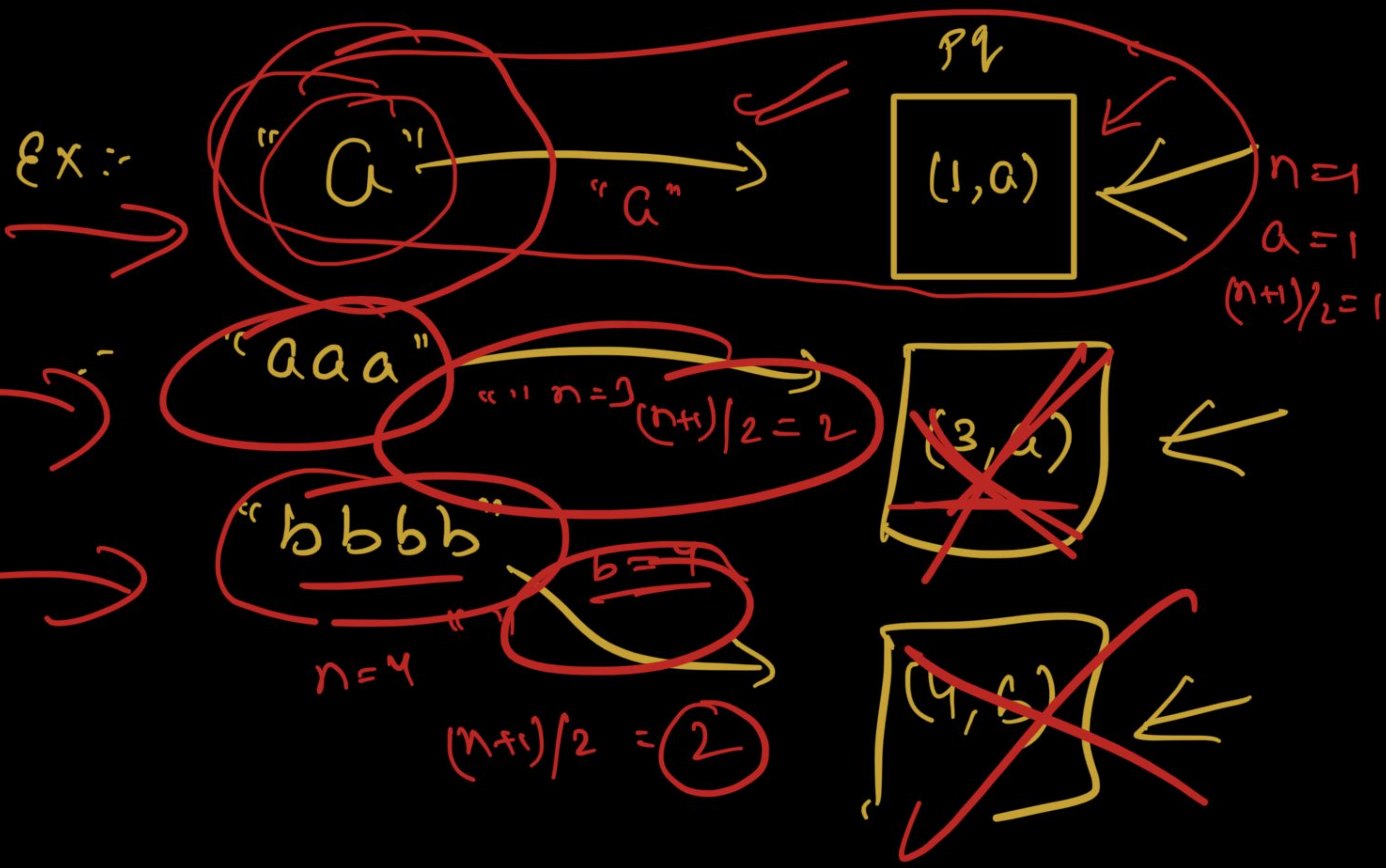
$(n+1)/2 = 5/2 = 2$

$(n+1)/2 =$

Aa

$(n+1)/2 = 5/2 = 2$

$(n+1)/2 =$



```
while (pq.size() >= 2) {
→
}
```

Ex:- "a"

{Ex:- "a"

```
if (!pq.empty()) {
    result = pq.top().second;
}
```

(1, a)

Story :-

① Store frequency of chars.

↳ if any char freq > (n+1)/2
return " " ; ←

② Max-heap {(freq , char)}

③ while (pq.size() >= 2)

⇒ form your string

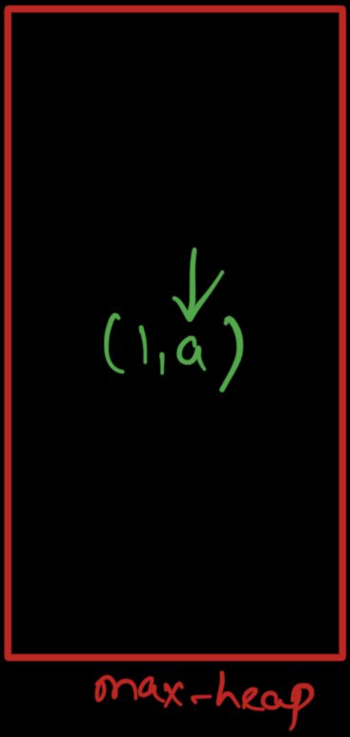
4

```
if (!pq.empty()) {
        res = pq.top().second ;
}
```

"aaabbbcdd" , n = 9

a → 3
b → 3
c → 1
d → 2

result = "badbadcba"



Approach-2

Approach - 2

max ↑

"aab"   a → ~~2~~ 1

a ___ a

i=0   i=2

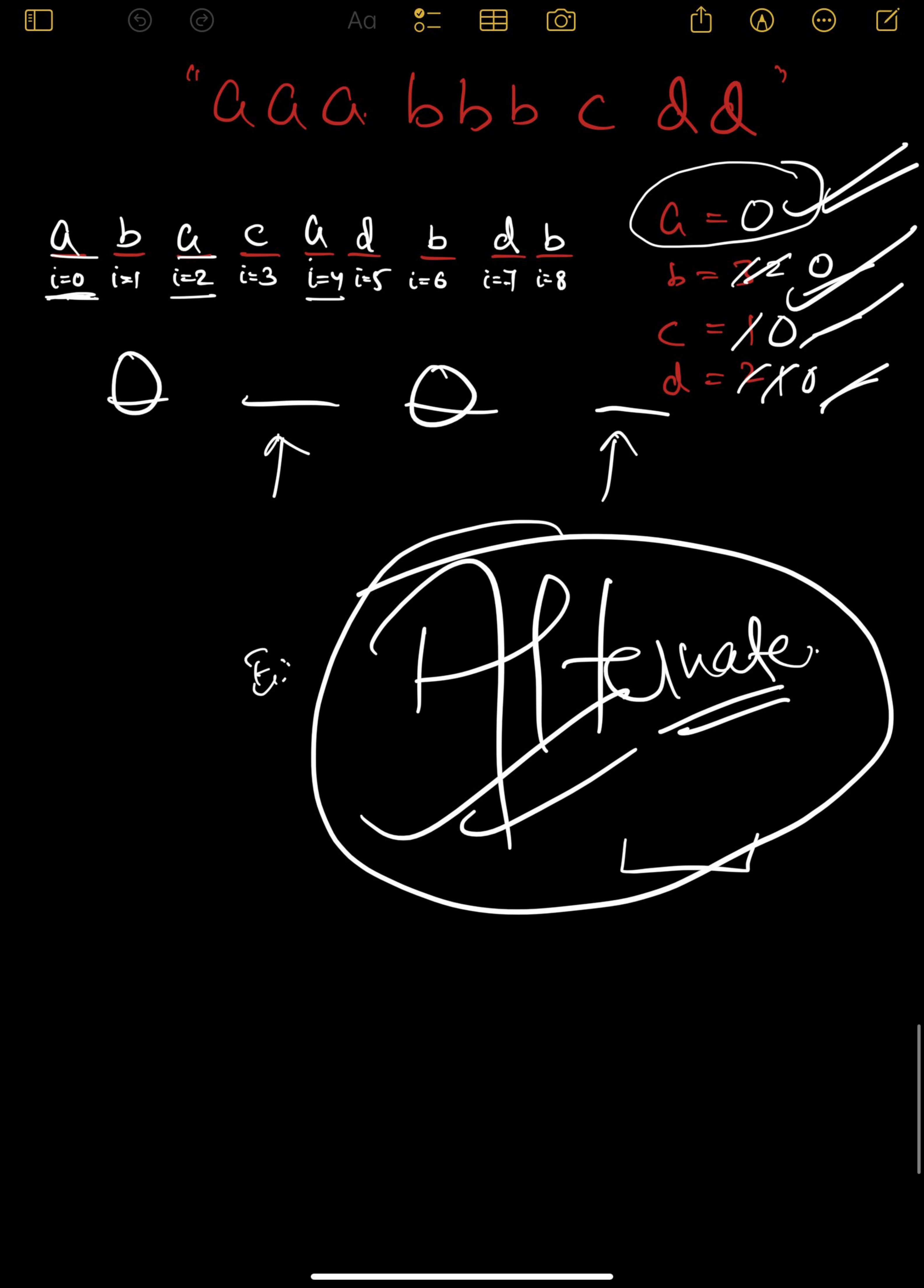
maxfre = 'a'
cn = 2

"aaabbbcdd"

a	b	a	c	a	d	b	d	b
i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8

a = 0
b = ~~3~~ ~~2~~ 0
c = ~~1~~ 0
d = ~~2~~ ~~1~~ 0

O ___ O ___



"aaa bbb c dd"

a	b	a	c	a	d	b	d	b
i=0	i=1	i=2	i=3	i=4	i=5	i=6	i=7	i=8

a = 0

b = ~~2~~ ~~1~~ 0

c = ~~1~~ 0

d = ~~2~~ ~~1~~ 0

0 _ 0 _

Eg: Alternate