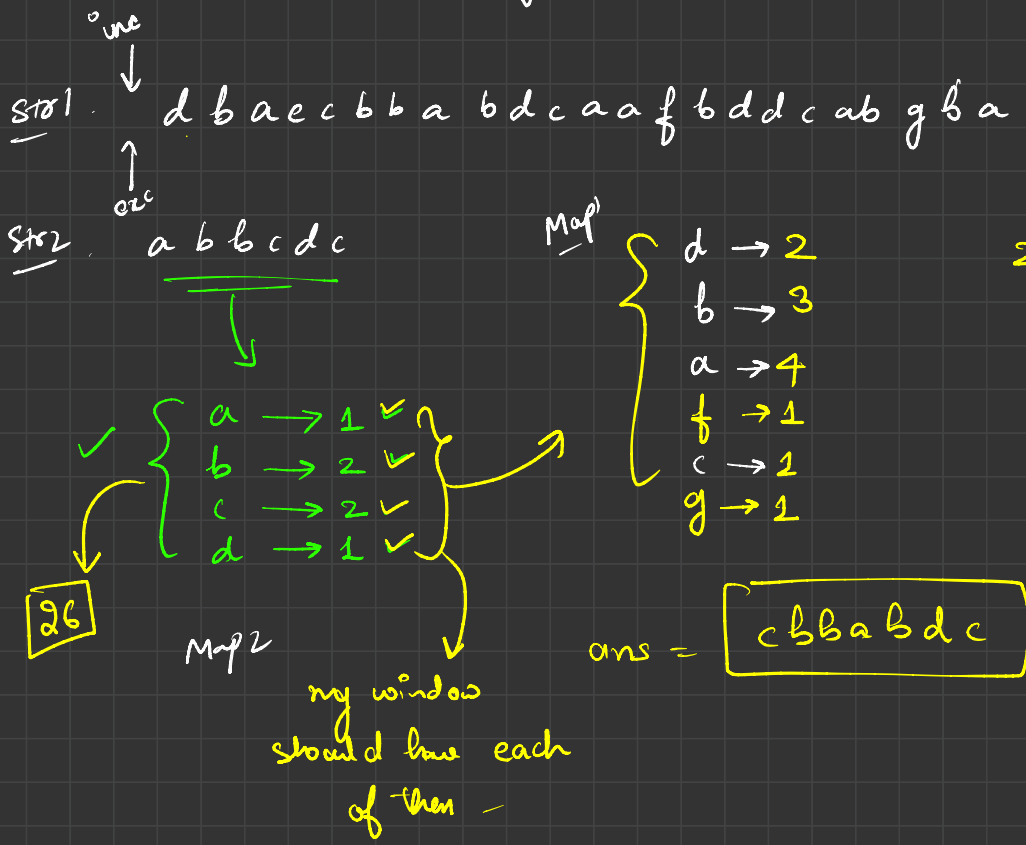
Minimum Window Substring



Str.substring(exc+1, mcnt)

inc

Str1 : d b a e c b b a b d c a a f b d d c a b g b a

exc

Str2 : a b b c d c

a → 1
b → 2
c → 2
d → 1

Map2

d → 1
b → 1 ≠ 3
a → 1

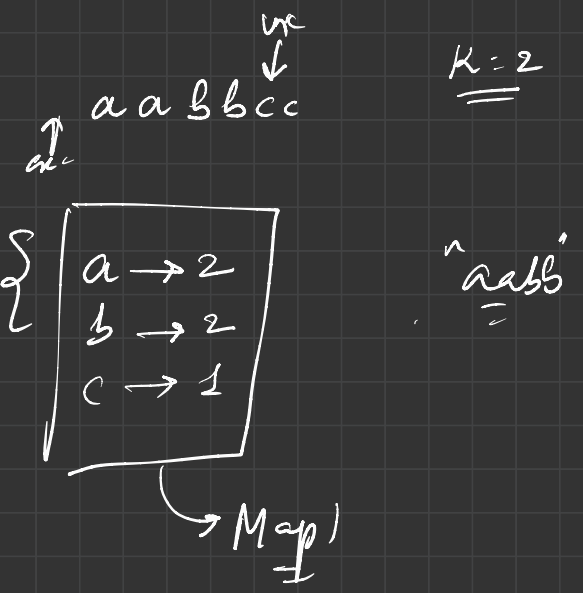
c → 1

dmcnt = 6

mcnt = ∅ ≠ 1 ≠ 3 ≠ 5

O(2N) ≈ O(N)

```java
while (true) {
    boolean f1 = false;
    // include people
    while (inc + 1 < s.length() && map1.size() <= k) {
        inc  = inc + 1;
        char ch = s.charAt(inc);
        map1.put(ch, map1.getOrDefault(ch, 0) + 1);
        if (map1.size() == k) {
            String pans = s.substring(exc + 1, inc + 1);
            if (ans.length() == 0 || pans.length() > ans.length()) {
                ans = pans;
            }
        }
        f1 = true;
    }
    boolean f2 = false;
    // exclude people
    while (exc < inc && map1.size() > k) {
        exc = exc + 1;
        char ch = s.charAt(exc);
        map1.put(ch, map1.get(ch) - 1);
        if (map1.get(ch) == 0) {
            map1.remove(ch);
        }
        f2 = true;
    }
    if (f1 == false && f2 == false) {
        break;
    }
}
```



{ Break 2:10 pm }