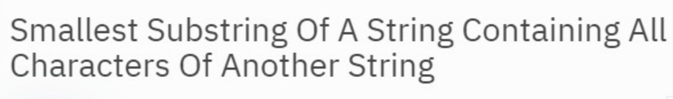


Smallest Substring Of A String Containing All Characters Of Another String

s1 = time to practice

s2 = toc

ans : → toprac
→ opract

(i) aquire & release.

dmtc = 3

(i) aquire till $\boxed{mtc == dmtc}$

(ii) release (better ans)

t -> 1
o -> 1
c -> 1

s2 = t o c

j

time to practice

i

m2

t -> 1

r -> 1
a -> 1
c -> 2
i -> 1
e -> 1

p -> 1

2
~~2~~ ~~3~~

mtc = ~~0~~ ~~1~~ ~~2~~ ~~3~~

ans = ~~timetoprac~~

~~imetoprac~~

~~metoprac~~

~~etoprac~~

toprac

dmtc = 6

5

~~6~~

~~8~~

mtc = ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~5~~ ~~6~~

j

S1 = e a m b c a f g d b a c n c b d e

i

i = -1

j = -1

ans = ~~eambcafgdbac~~

~~ambcafgdbac~~

~~mbcafgdbac~~

~~bcafgdbac~~

~~gdb~~acncb

dbacncb

a-1 d-1

b-1 n-1

c-2 e-1

a -> 1

b -> 2

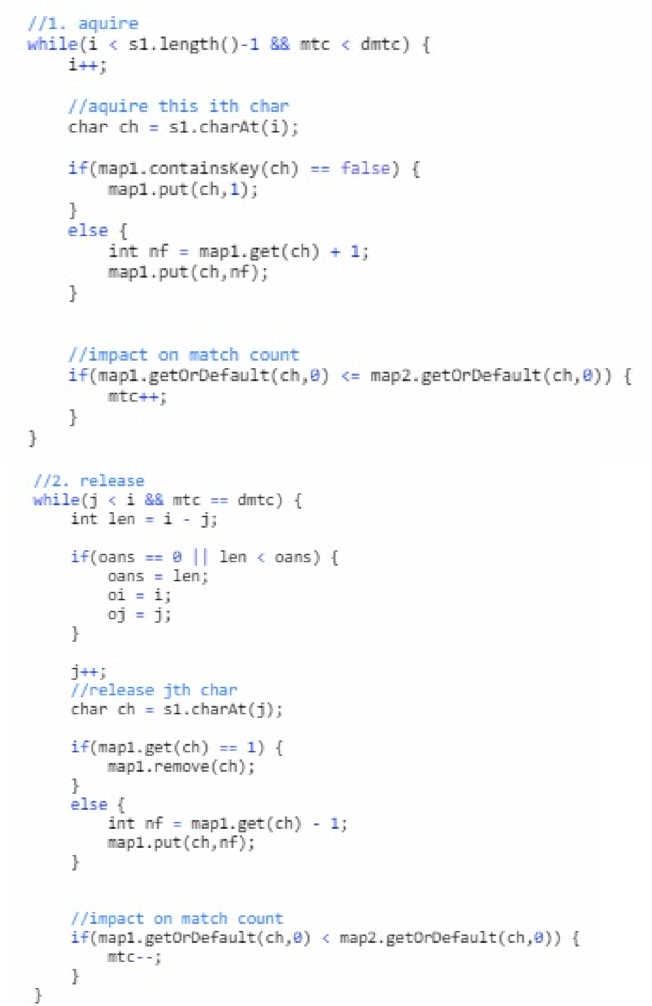
c -> 2

d -> 1

m2

S2 = a b c b d c

while (i < s1.length()-1) {



```
//1. aquire
while(i < s1.length()-1 && mtc < dmtc) {
    i++;

    //aquire this ith char
    char ch = s1.charAt(i);

    if(map1.containsKey(ch) == false) {
        map1.put(ch,1);
    }
    else {
        int nf = map1.get(ch) + 1;
        map1.put(ch,nf);
    }


    //impact on match count
    if(map1.getOrDefault(ch,0) <= map2.getOrDefault(ch,0)) {
        mtc++;
    }
}

 //2. release
 while(j < i && mtc == dmtc) {
    int len = i - j;

    if(oans == 0 || len < oans) {
        oans = len;
        oi = i;
        oj = j;
    }

    j++;
    //release jth char
    char ch = s1.charAt(j);

    if(map1.get(ch) == 1) {
        map1.remove(ch);
    }
    else {
        int nf = map1.get(ch) - 1;
        map1.put(ch,nf);
    }


    //impact on match count
    if(map1.getOrDefault(ch,0) < map2.getOrDefault(ch,0)) {
        mtc--;
    }
 }
```

}

mtc = ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~6~~ ~~6~~ 5

oj=7 oi = 14 oans = 7 abdcbc

oj j oi

e a m b c a j g d b a c n c b d e

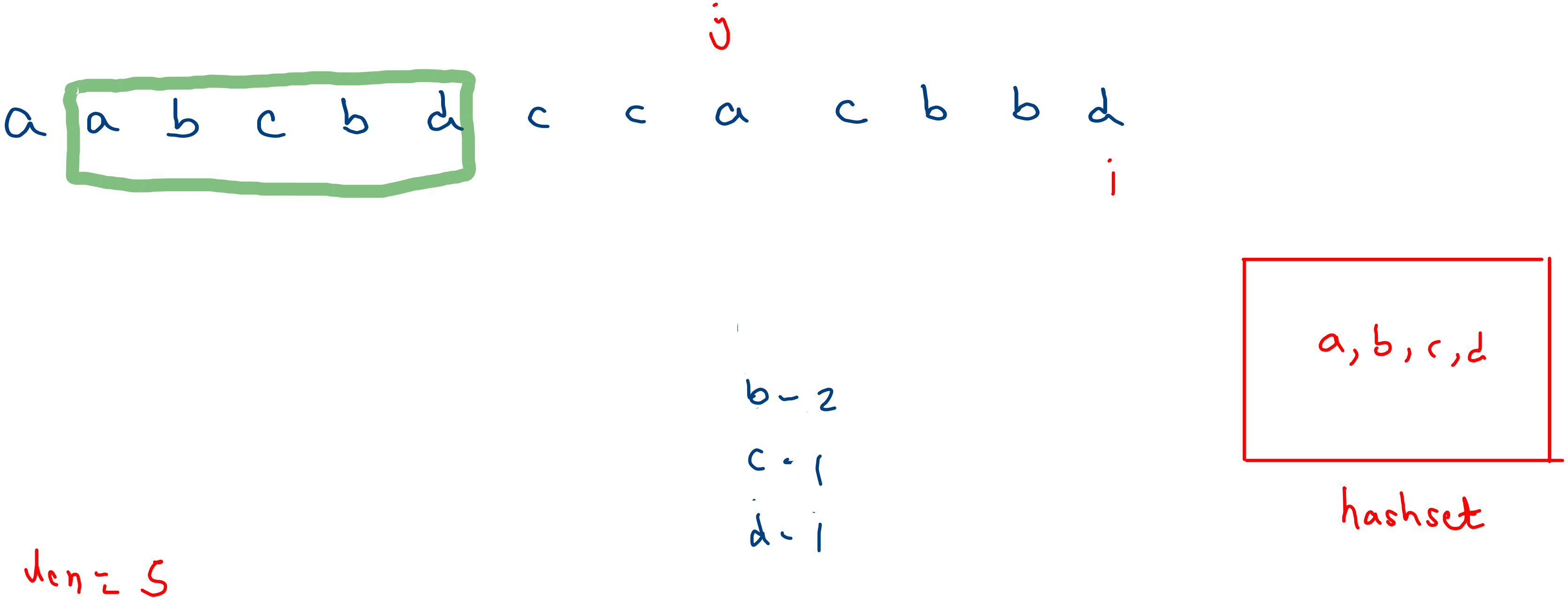
i

ans: dbacncb

a-1.	d-1	a- 1
		b-2
b-1.	c-1	c- 2
		d-1
c- 2	n-1	map2

dmtc = 6

Smallest Substring Of A String Containing All Unique Characters Of Itself



Longest Substring Without Repeating Characters

j

a a c c b a c c d e g s d

i

S

s-1

d-1

e-1

g-1

ans creation.

(i) aquire till invalid.

(ii) release to be valid

ans again.