10-Dec-2021

10 December 2021 20:59

String:

=) "abcde"

=) Immutable

```
String n = "abcd";
n = n + "fgh";
```

```
String a = "a";
       b = "a";
```

String is a Class:

StringBuilder

IsPalindrome?

1331 (=) 1331

123 ≠ 321

```
int n = 5
n += 10   n = n + 10
```

```
isPalindromeRev(String s){
temp = "";
for(i = length-1; i ≥ 0; i--){
temp += s.charAt(i);
}
```

```
temp = temp + s.charAt(i);
```

```
isPalindrome2(String s){
temp = "4321";
for(i = n-1 to 0){
... = s.charAt(i);
```

1 2 3 4 3 2

1 2 3 4

temp += s.charAt(i);

}

i = 0

Toggle Case :-

kRish = KrISH

'a' 'A'

1 + 80 — 'a' A 1 = 80

4 + 80 — 1

83

84 + 1 − 80 = 5

'a' != 'A'

1   80 — 106

'z' = 4   A − Z

1 − 26

a − z

E = 84

e = 5

a B c D

a = 97

B = 66

c = 99

D = 68

A = 65

Z = 91

a = 97

z = 123

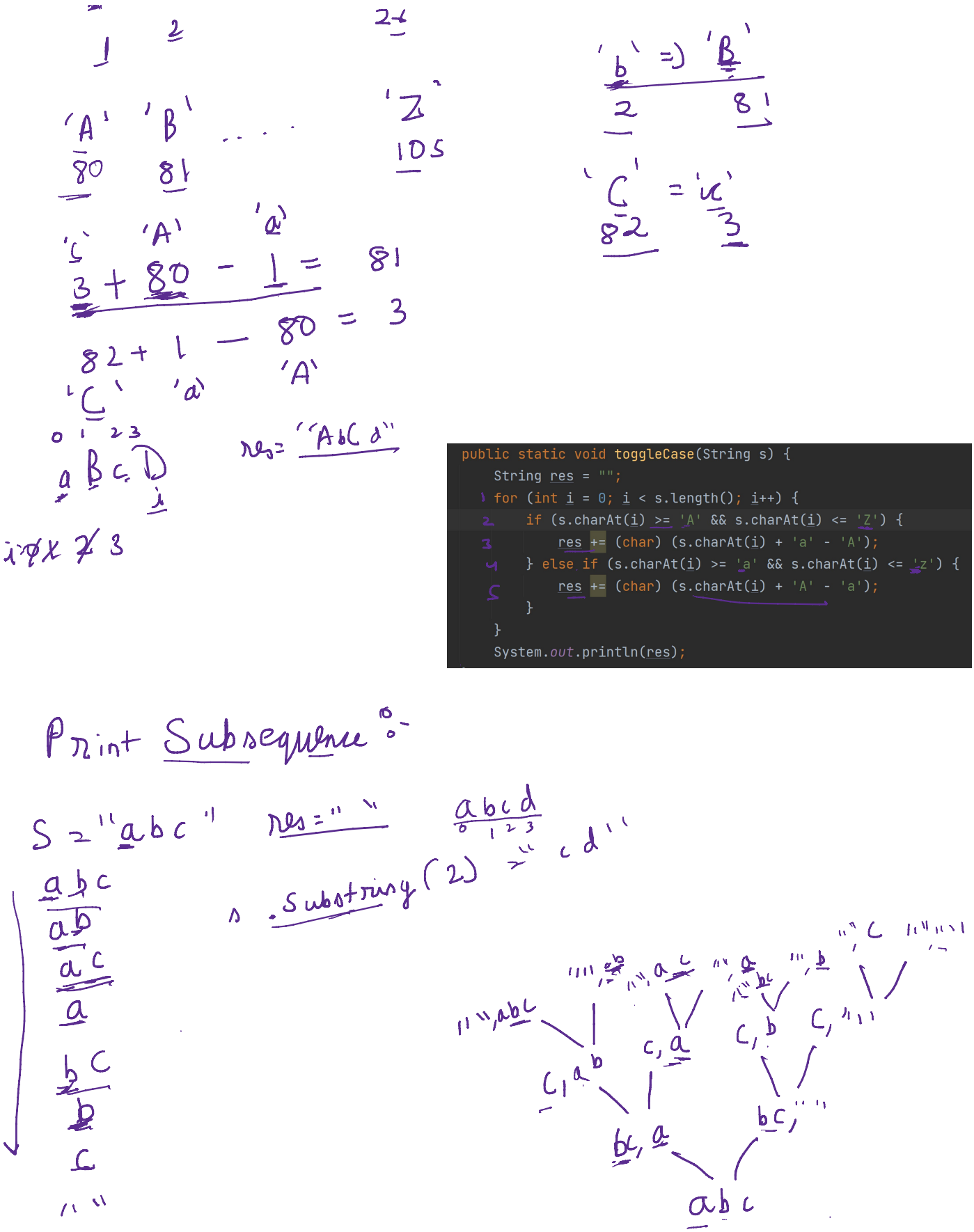
```java
public static void toggleCase(String s) {
    String res = "";
    for (int i = 0; i < s.length(); i++) {
        if (s.charAt(i) >= 'A' && s.charAt(i) <= 'Z') {
            res += (char) (s.charAt(i) + 'a' - 'A');
        } else if (s.charAt(i) >= 'a' && s.charAt(i) <= 'z') {
            res += (char) (s.charAt(i) + 'A' - 'a');
        }
    }
    System.out.println(res);
```

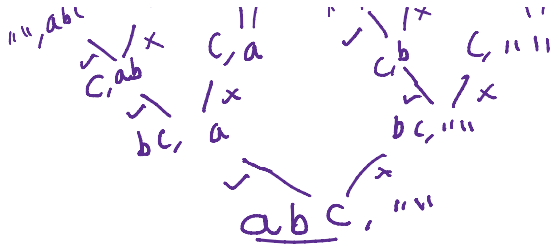
'a' 'b' . . . . 'z'

1   2   26

'a' != 'A'

'b' => 'B'

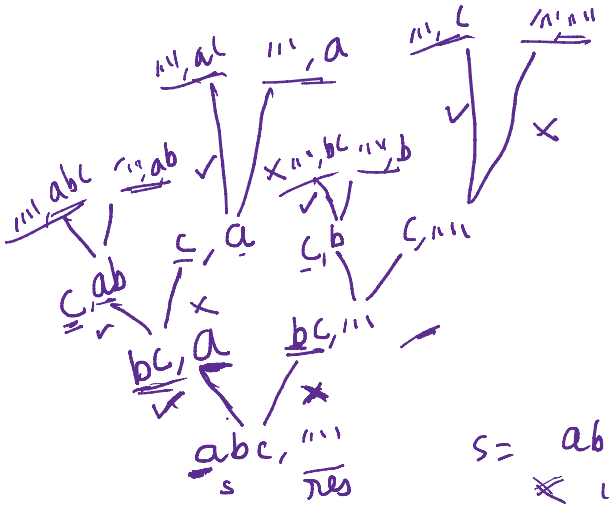

'A' 'B' ... 'Z'
80 81 105
'b' => 'B'
2 81
'C' = 'c'
82 3
'C' 'A' 'a'
3 + 80 - 1 = 81
82 + 1 - 80 = 3
'C' 'a' 'A'
0 1 2 3
a B c D
i
res = "AbCd"
i % 2 != 3
public static void toggleCase(String s) {
String res = "";
for (int i = 0; i < s.length(); i++) {
if (s.charAt(i) >= 'A' && s.charAt(i) <= 'Z') {
res += (char) (s.charAt(i) + 'a' - 'A');
} else if (s.charAt(i) >= 'a' && s.charAt(i) <= 'z') {
res += (char) (s.charAt(i) + 'A' - 'a');
}
}
System.out.println(res);
Print Subsequence :-
S = "abc" res = ""
abcd
0 1 2 3
s.substring(2) => "cd"
abc
ab
ac
a
bc
b
c
""
"abc"
"ab" "ac" "a" "bc" "b" "c" ""
"abc"
"ab" "ac" "a" "bc" "b" "c" ""
"abc"
"ab" "ac" "a" "bc" "b" "c" ""
abc



```java
public static void printSubSeqRec(String s, String res) {
    if (s.length() == 0) {
        System.out.println(res);
        return;
    }
    char curr = s.charAt(0);
    //Accepting the curr character
    printSubSeqRec(s.substring(1), res: res + curr);
    //Rejecting the curr character.
    printSubSeqRec(s.substring(1), res);
}
```

abcde

abc



s = "abc"

res = ""

s = abc

✗ "bc"

abc

ab

a c

a

bc

b

c

""