Strings

Ques Power of string

abbcccddddeeeeeffgghheecccc

a b c d e f h e c

c → 3, d → 4, e → 5, f → 2, h → 2, e → 2, c → 4

↓ unique

e-power

Substring

a b b c c c d d d d e e e e e f f g g h h e e c c c c

count = 1

```
if( str.charAt(i) == str.charAt(i-1)){
    count ++.
}
else {
    ans = Math.max( ans, count);
    count to reset = 1
}
```

ans = 3 4

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();

    int count =1;
    int ans =1;
    for(int i=1;i<str.length();i++){
        if(str.charAt(i)== str.charAt(i-1)){
            count++;
        }else{
            ans = Math.max(ans,count);
            count=1;
        }
    }
    ans = Math.max(ans,count);
    System.out.println(ans);

    /* Enter your code here. Read input from STDIN. P
}
```

abbcccddddeeeeeffgghheecccc

TC = O(n)

count = 1 2 1 2 3 1 2 3 4 1 2 3 4 5 1 2 1 2 1 2 3 4

ans = 1 2 3 4 5

Qus Consecutive 0's & 1's

00110011

count =

All substrings

# of 0's and # of 1's are same

Brute force

1. Find all substring
2. count substrings having equal # of 0's & 1's
3. count++;

Optimized

[00111111] 0011 1100

[01 0011 000111]

(0011)

count = Math.min(count0's, count1's);

3 4

count = 3

ans += Math.max(count0's, count1's)

ans = 6

i = 0

count0's = 0

count1's = 2

```
int ans =0;

int i=0;

while(i<str.length()){
    int count0=0;
    int count1=0;
    if(str.charAt(i)=='1'){ // if starts from 1
        while(i<str.length() && str.charAt(i)=='1'){ // count no of consecutive 1's
            count1++;
            i++;
        }
        int j =i;
        while(j<str.length() && str.charAt(j)=='0'){ // count no of consecutive followed by 0's
            count0++;
            j++;
        }
    }else{ // if starts ffrom 0
        while(i<str.length() && str.charAt(i)=='0'){ // count no of consecutive 0's
            count0++;
            i++;
        }
        int j =i;
        while(j<str.length() && str.charAt(j)=='1'){ // count no of consecutive followed by 1's
            count1++;
            j++;
        }
    }
    ans += Math.min(count0, count1);
}

System.out.println(ans);
/* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be name
```

Qus — Long pressed keyboard.

string str1 = "alex"

string str2 = "aaleex"

false

true

alex | True | false

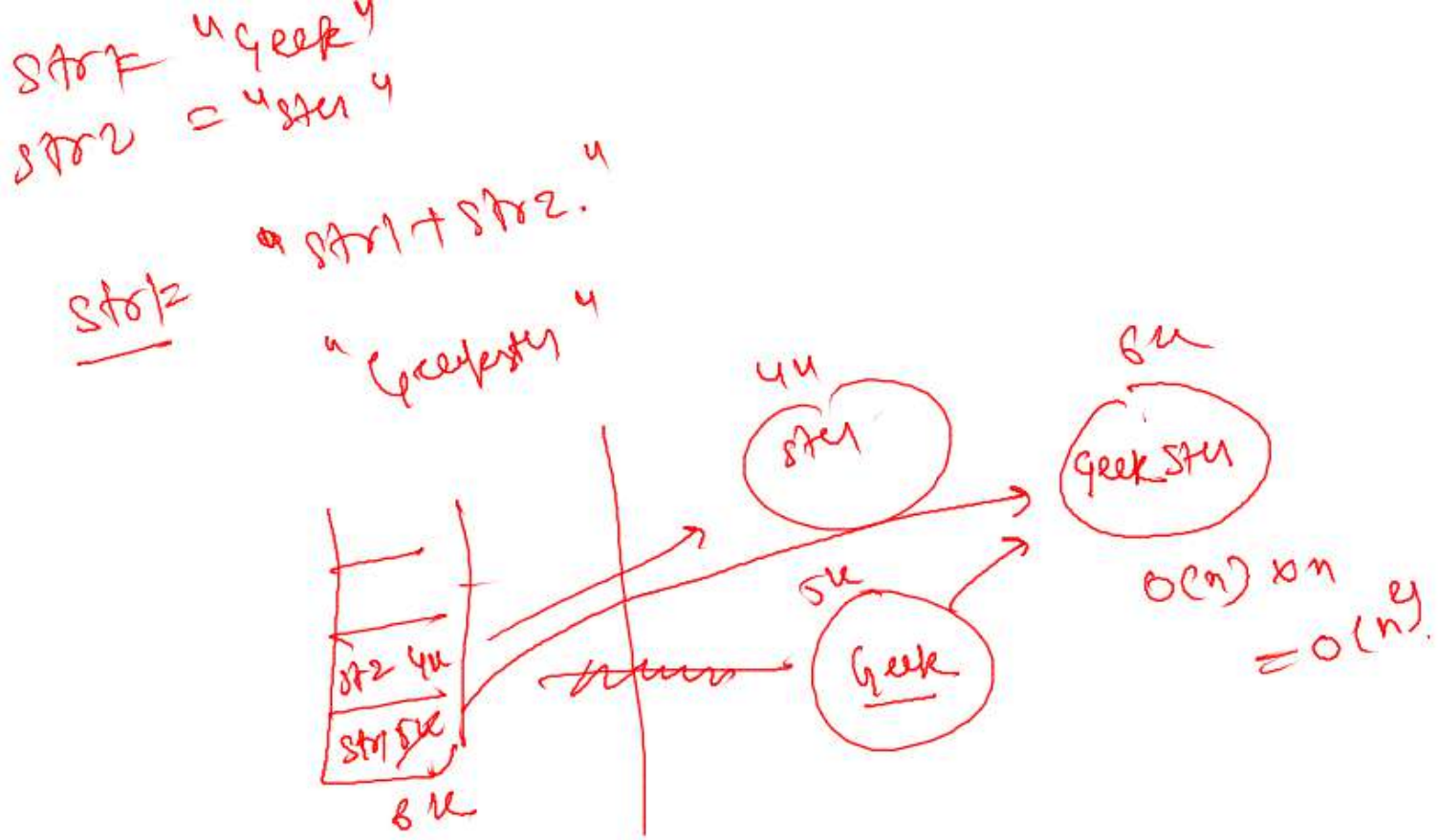
conditional → True / false

(==)

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str1= scn.nextLine();
    String str2= scn.nextLine();

    int i=0;
    int j=0;
    while(i<str1.length()){
        if(str1.charAt(i)== str2.charAt(j)){
            i++;
            j++;
        }else if(j>0 && j<str2.length() && str2.charAt(j)== str2.charAt(j-1)){
            j++;
        }else{
            System.out.println("false");
            return;
        }
    }

    System.out.println(i==str1.length());
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your
}
```




```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str1 = scn.nextLine();
    String str2 = scn.nextLine();

    StringBuilder sb = new StringBuilder();
    int i=0;
    while(i<str1.length()){
        sb.append(str1.charAt(i));
        sb.append(str2.charAt(i));
        i++;
    }

    System.out.println(sb.toString());
    /* Enter your code here. Read input from STDIN. Print output to STDOUT.
}
```