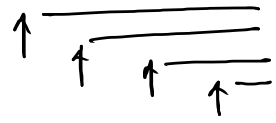
Sum of All Substrings

str = "1234"



Substring

- "1" → 1
- "12" → 12
- "123" → 123
- "1234" → 1234

"2" → 2 "3" → 3 "4" → 4

"23" → 23 "34" → 34

"234" → 234

⇒ How to convert a String to Integer

1) `Integer.parseInt(str);`

2) `Integer.valueOf(str);`

psudo code

sum = 0

1) loop i from 0 to (n-1)

1.1) loop j from (i+1) to (n)

1.1.1) find substring and convert into int

sum += int

return sum;

Code

T.C = $O(N^3)$

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    int ans = sumOfSub(str);
    System.out.println(ans);
}
public static int sumOfSub(String str) {
    int sum = 0;
    for (int i = 0; i < str.length(); i++) {
        for (int j = i + 1; j <= str.length(); j++) {
            String sub = str.substring(i, j);
            int num = Integer.parseInt(sub);
            sum += num;
        }
    }
    return sum;
}
```

Desired String

- count the substrings that start and end with 'A'
- print length of longest substring with above cond^n
- print that longest substring

pseudo code

```
count = 0
1) loop i from 0 to (n-1)
    1.1) loop j from (i+1) to (n)
        1.1.1) find substring and check if start & end
               with 'A'
               count++;
```

str = "ABADA"

Sub

A	B	A	D	A
AB	BA	AD	DA	
✓ ABA	BAD	✓ ADA		
ABAD	BADA			
✓ ABADA				

Count = 3

len = 5

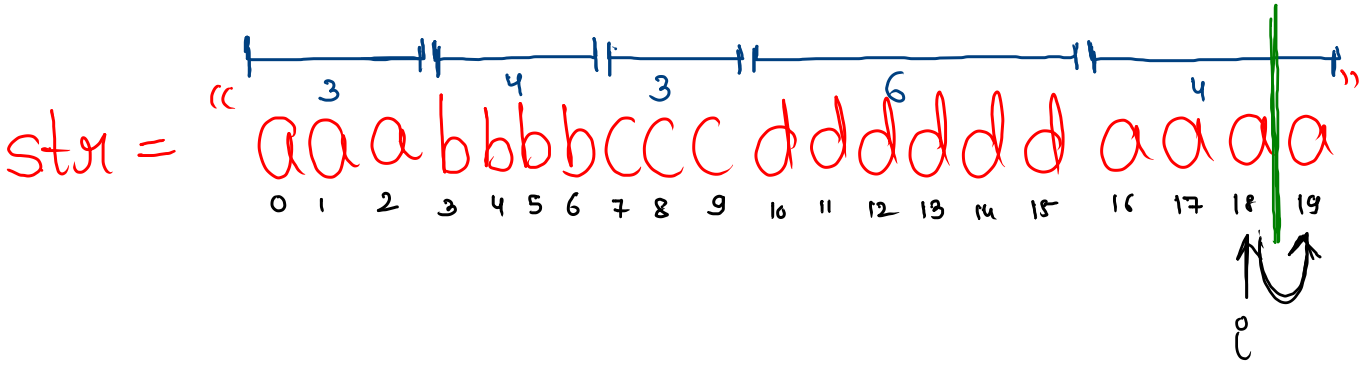
sub = ABADA

Code

T.C = $O(N^3)$

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    desiredString(str);
}
public static void desiredString(String str) {
    int count = 0;
    int len = 0;
    String maxSub = "";
    for (int i = 0; i < str.length(); i++) {
        for (int j = i + 1; j <= str.length(); j++) {
            String sub = str.substring(i, j);
            if ( sub.length() > 1 && sub.charAt(0) == 'A' && sub.charAt(sub.length() - 1) == 'A' ) {
                count++;
                if ( sub.length() > len ) {
                    len = sub.length();
                    maxSub = sub;
                }
            }
        }
    }
    if ( count == 0 ) {
        System.out.println(-1);
    } else {
        System.out.println(count);
        System.out.println(len);
        System.out.println(maxSub);
    }
}
```

Power of a String (Imp)



ans = 6

count = ~~1 2 3 1 2 3 4 1 2 3 1 2 3 4 5 6 1 2 3~~ 4

ans = ~~-∞ 3 4~~ 6

psudo code

count = 1 , ans = $-\infty$

1) traverse i loop from 0 to (n-2)

1.1) check if char at i == char at (i+1)

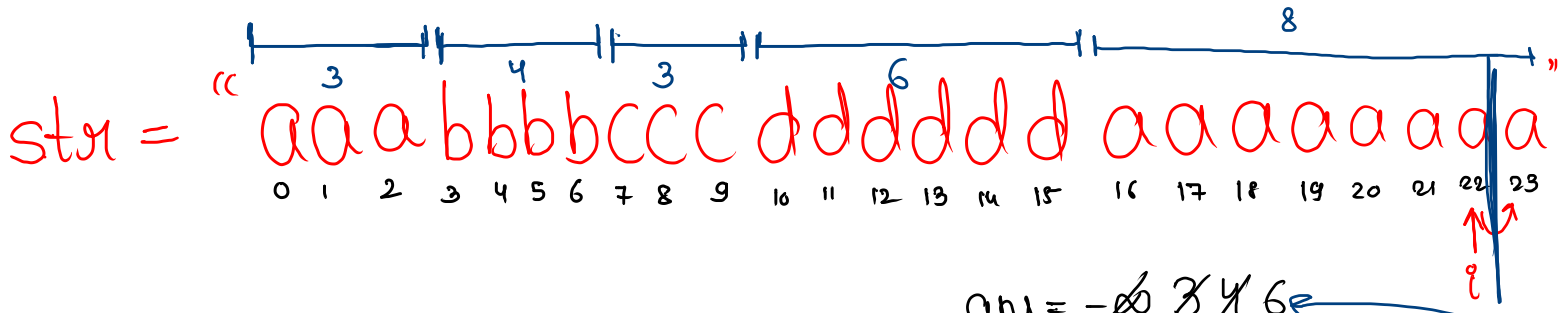
count++;

1.2) check if char at i != char at (i+1)

ans = max (ans , count) ;

count = 1

2) return ans ;



ans = ~~-∞~~ ~~3~~ ~~4~~ 6

Count = ~~1~~ ~~2~~ ~~3~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~1~~ ~~2~~ ~~3~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~

~~5~~ ~~6~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ 8

code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    int ans = powerOfString(str);
    System.out.println(ans);
}
public static int powerOfString(String str) {
    int count = 1;
    int ans = Integer.MIN_VALUE;
    for (int i = 0; i < str.length() - 1; i++) {
        if ( str.charAt(i) == str.charAt(i + 1) ) {
            count++;
        } else {
            ans = Math.max(ans, count);
            count = 1;
        }
    }
    ans = Math.max(ans, count);
    return ans;
}
```

gmp

T.C = O(N)

S.C = O(1)