Compression

Given a string, str, of length n composed of English letters (a-z,A-Z), a *compression operation* on S is described as follows:

1. Find a substring having some repeated character, c. The substring *must* contain every consecutive repetition of c for that section of str (i.e.: the left and right boundaries of the substring must fall at the first non-c characters).
2. Find the length, k, of said substring.
3. For every substring having $k > 1$, replace the substring with a single occurrence of the character, c, followed by the integer value of k.

The above steps should be repeated until str no longer contains any substring of repeated characters having $k > 1$. For single occurrences of a character ($k = 1$), the character cannot be compressed and should be left as-is. For example, if str = *"aaaaabbbbbbbbbccccdeeeeeee"*, a compression operation would reduce str to *"a5b9c4de7"*. Observe that d is not repeated, so no compression number (k) is displayed after it.

Complete the *compress* function which takes a string, str as a parameter, performs a compression operation on str, and returns the compressed string.

Input Format

A single line containing string str of length n.

Constraints

- $0 < n < 1024$

Output Format

You are not responsible for printing anything to stdout. Your *compress* function must return the compressed string.

Sample Input 1

```
aaaaabbbbbbbbbccccpqrstuv
```

Sample Output 1

```
a5b9c4pqrstuv
```

1

2

For "*aaaah*", *k=5*, so this substring is replaced with "*a5*".
For "*bbbbbbbbb*", *k=9*, so this substring is replaced with "*b9*".
For "*cccc*", *k=4*, so this substring is replaced with "*c4*".
As *p, q, r, s, t, u,* and *v* are all non-repeating characters, they are left as-is. Thus, the string returned by the *compress* function is `a5b9c4pqrstuv`.

YOUR ANSWER

We recommend you take a quick tour of our editor before you proceed. The timer will pause up to 90 seconds for the tour. Start tour

✖

Original code | C# ⌄ | ⚙

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
class Solution {

/*
 * Complete the function below.
 */
    static string compress(string str) {


    }


    static void Main(String[] args) {↔}
}
```

Line: 11 Col: 1

☐ **Test against custom input**

Run Code | Submit code & Continue

(You can submit any number of times)

1

2