You're given a [substring](keyword://substring) s of some [cyclic string](keyword://cyclic-string). What's the length of the smallest possible string that can be concatenated to itself many times to obtain this cyclic string?

**Example**

For s = "cabca", the output should be  
cyclicString(s) = 3.

"cabca" is a substring of a cycle string "abcabcabcabc..." that can be obtained by concatenating "abc" to itself. Thus, the answer is 3.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] string s**

*Constraints:*  
3 ≤ s.length ≤ 15.

* **[output] integer**

<https://codefights.com/arcade/code-arcade/spring-of-integration/tKwbrKAQhoCZFqg33>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int cyclicString(string s)

{

string s1 = "";

int indiceultimo = s.IndexOf(s[s.Length-1]);

s1 = s + "" + s.Substring(indiceultimo+1, s.Length - indiceultimo-1);

Console.WriteLine(s1);

//string hs = "";

int ans = 0;

string p1 = "";

for (int i = 0; i < s1.Length; i++)

{

p1 = s1.Substring(0, i + 1);

string p2 = "";

if (i + 1 + p1.Length < s1.Length)

p2 += s1.Substring(i + 1, p1.Length);

Console.WriteLine(p1 +" "+ p2);

if (p1 == p2)

{

//ans = p1.Length;

return p1.Length;

//break;

}

}

return s.Length;

}

static void Main(string[] args)

{

// string s = "cabca";

//string s = "ccccccccccc";

//Console.WriteLine(cyclicString(s));

//string s = "bcacbcacbcacbc";

//string s = "123456";

// string s = "cacaca";

string s = "baa";

Console.WriteLine(cyclicString(s));

Console.ReadLine();

}

}

}