Given a string s, your mission is to apply the following easy-to-comprehend algorithm to it:

1. Find all the *words* in s, where a *word* is a sequence of consecutive alphanumeric characters with no other letters around it;
2. Reverse the characters in each *word*;
3. For each *word*, swap the cases of its characters so that the case of a character at each position differs from the case at the corresponding position of the original (unreversed) *word*.

Return the obtained string as the answer.

**Example**

For s = "So, what is CodeFights?", the answer should be  
reverseInverse(s) = "oS, TAHW SI sTHGiFEDOC?".

There are 4 *words* in s: "So", "what", "is", and "CodeFights". Let's take the word "CodeFights" as an example:

* The letters 'C' at index 0 and letter 'F' at index 4 are uppercase, while all the other letters are lowercase;
* "codefights" reversed becomes "sthgifedoc";
* With the cases swapped, the letters at indices 0and 4 should be lowercase and all the other letters should be uppercase;
* Thus, the final word is "sTHGiFEDOC".

**Input/Output**

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

A string containing only alphanumeric characters and punctuation marks.

*Guaranteed constraints:*  
0 ≤ s.length < 500.

* **[output] string**

The result of applying the algorithm described above to s.

<https://codefights.com/challenge/bzkdx7vP3QJ8DgQTL>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static string reverseInverse(string s)

{

List<string> pals = new List<string>();

string concat = "";

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

{

if (char.IsLetter(s[i]) || char.IsNumber(s[i]))

{

concat += s[i];

}

else

{

if (concat.Length > 0)

{

pals.Add(concat);

concat = "";

}

}

}

if (concat.Length > 0)

{

pals.Add(concat);

}

//foreach (string elem in pals)

//{

// Console.WriteLine(elem + " ");

//}

List<string> invertidas = new List<string>();

foreach (string elem in pals)

{

char[] revElem = elem.ToCharArray();

Array.Reverse(revElem);

string rev = new string(revElem).ToLower();

string invCase = "";

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

{

if (char.IsLower(elem[i]))

{

invCase += char.ToUpper( rev[i]);

}

else if (char.IsUpper(elem[i]))

{

invCase += char.ToLower(rev[i]);

}

else if (char.IsNumber(elem[i]))

{

invCase += rev[i];

}

}

invertidas.Add(invCase);

}

//foreach (string elem in invertidas)

//{

// Console.Write(elem + " ");

//}

int indTodas = 0;

string todas = string.Join("", invertidas.ToArray());

//Console.WriteLine(todas);

char[] sch = s.ToCharArray();

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

{

if (char.IsLetterOrDigit(sch[i]))

{

if (indTodas < todas.Length)

{

sch[i] = todas[indTodas];

indTodas++;

}

}

}

return new string(sch);

}

static void Main(string[] args)

{

//string s = "So, what is CodeFights?";

string s = "3 days to go. Get your 100 images for $100.";

Console.WriteLine( reverseInverse(s));

Console.ReadLine();

}

}

}