Mr. Nonequation is bad at math. He sees a string of numbers, but he cannot figure out what operator to use to make an equation.

Given a string of numbers, turn it into a valid equation by inserting only one operator (+, -, \*, /, or %) and one equals sign ('=') and return it. If it is not possible, return "Not possible" instead.

The operators should be to the left of the equals sign ("=")

* "+" denotes addition
* "-" denotes subtraction
* "\*" denotes multiplication
* "/" denotes **non-integer** division
* "%" denotes modulus (remainder)

The operator should be place to the left of the equals sign.  
It is guaranteed that there is only one possible answer. It is guaranteed that the numbers in the equation won't contain leading zeros.

**Example**

* For nums = "123123246", the output should be  
  BasicEquationMaker(nums) = "123+123=246".
* For nums = "113", the output should be  
  BasicEquationMaker(nums) = "Not possible".

**Input/Output**

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

A string of numbers.

*Constraints:*  
3 ≤ nums.length ≤ 15.

* **[output] string**

A valid equation, or "Not possible" if it's impossible to construct one.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

static string BasicEquationMaker(string nums)

{

for (int i = 0; i < nums.Length - 1; i++)

{

for (int j = i; j < nums.Length; j++)

{

string a = nums.Substring(0, i);

string b = nums.Substring(i, j - i);

string c = nums.Substring(j, nums.Length - j);

//Console.WriteLine(a + " " + b + " " + c);

if (a.Length > 0 && b.Length > 0 && c.Length > 0)

{

double ia = double.Parse(a );

double ib = double.Parse(b );

double ic = double.Parse(c);

if (ia \* ib == ic)

{

return a + "\*" + b + "=" + c;

}

else if (ib > 0 && ia / ib == ic)

{

return a + "/" + b + "=" + c;

}

else if (ia + ib == ic)

{

return a + "+" + b + "=" + c;

}

else if (ia - ib == ic)

{

return a + "-" + b + "=" + c;

}

else if (ib > 0 && ia % ib == ic)

{

return a + "%" + b + "=" + c;

}

}

}

}

return "Not possible";

}

static void Main(string[] args)

{

Console.WriteLine(BasicEquationMaker("1527"));

Console.ReadLine();

}

}

}