Consider a string containing only letters and whitespaces. It is allowed to replace some (possibly, none) whitespaces with newline symbols to obtain a multiline text. Call a multiline text *beautiful* if and only if each of its lines (i.e. [substrings](keyword://substring) delimited by a newline character) contains an equal number of characters (only letters and whitespaces should be taken into account when counting the total while newline characters shouldn't). Call the length of the line the *text width*.

Given a string and some integers l and r (l ≤ r), check if it's possible to obtain a beautiful text from the string with a text width that's within the range [l, r].

**Example**

* For inputString = "Look at this example of a correct text", l = 5 and r = 15, the output should be  
  beautifulText(inputString, l, r) = true.

We can replace 13th and 26th characters with '\n', and obtain the following multiline text of width 12:

Look at this

example of a

correct text

* For inputString = "abc def ghi", l = 4 and r = 10, the output should be  
  beautifulText(inputString, l, r) = false.

There are two ways to obtain a text with lines of equal length from this input, one has width = 3 and another has width = 11 (this is a one-liner). Both of these values are not within our bounds.

**Input/Output**

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

*Constraints:*  
10 ≤ inputString.length ≤ 40.

* **[input] integer l**

A positive integer.

*Constraints:*  
1 ≤ l ≤ r.

* **[input] integer r**

A positive integer.

*Constraints:*  
l ≤ r ≤ 15.

* **[output] boolean**

<https://codefights.com/arcade/code-arcade/spring-of-integration/PSeEALZTxd93zBHS7>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static bool beautifulText(string inputString, int l, int r)

{

int distancia = inputString.IndexOf(' ');

int i = 0;

while (distancia > -1)

{

bool esDivisible = true;

for (i = distancia; i < inputString.Length; i += distancia)

{

if (inputString[i] == ' ')

{

// Console.WriteLine(i + " " + inputString[i] + " ");

i++;

}

else if (inputString[i] != ' ')

{

esDivisible = false;

break;

}

}

if (i != inputString.Length)

{

esDivisible = false;

}

if (esDivisible)

{

if (distancia >= l && distancia <= r)

return true;

}

distancia = inputString.IndexOf(' ', distancia + 1);

}

return false;

}

static void Main(string[] args)

{

//string s = "123 4567 8 9 12345";

//Console.WriteLine(s.IndexOf(' ',4));

//string s = "012\*456\*890\*234\*567";

//beautifulText(s, 0, 0);

string inputString = "Look at this example of a correct text";

//01234567890123456789012345678901234567

int l = 5, r = 15;

Console.WriteLine(beautifulText(inputString, l, r));

//string inputString = "abc def ghi";

//int l = 4;

//int r = 10;

//Console.WriteLine(beautifulText(inputString, l, r));

Console.ReadLine();

}

}

}