Pick a positive integer, write it out in English words, then count the total number of letters in the words. Keep repeating this process using the total as the new number to write and eventually you will reach an infinite loop at the number 4, no matter which number you start from.

Your task is to find the number of steps it takes to reach the number 4 from the given integer n.

Here are some examples of how numbers should be written and counted:

* 1 = **one** (3 letters)
* 14 = **fourteen** (8 letters)
* 30 = **thirty** (6 letters)
* 42 = **forty two** (8 letters)
* 216 = **two hundred sixteen** (17 letters)
* 1500 = **one thousand five hundred** (22 letters)
* 1000000 = **one million** (10 letters)
* 987654321 = **nine hundred eighty seven million six hundred fifty four thousand three hundred twenty one** (77 letters)

**Example**

For n = 377, the output should be  
NumberOfLetters(n) = 5.

* 377 would be written as three hundred seventy seven which has 24 letters.
* 24 would be written as twenty four which has 10letters.
* 10 would be written as ten which has 3 letters.
* 3 would be written as three which has 5 letters.
* 5 would be written as five which has 4 letters.

We have now reached 4 and there were 5 steps in total, so the result is 5.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] integer n**

The starting number.

*Constraints:*  
0 < n < 109

* **[output] integer**

The number of steps taken to reach 4 from n.

<https://codefights.com/challenge/wcNrgbtmpWWn4DTwu?utm_source=featuredChallenge&utm_medium=email&utm_campaign=email_notification>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

//http://stackoverflow.com/questions/3213/convert-integers-to-written-numbers

public static string AmountInWords(int n)

{

// var n = (int)amount;

if (n == 0)

return "";

else if (n > 0 && n <= 19)

{

var arr = new string[] { "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine", "Ten", "Eleven", "Twelve", "Thirteen", "Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen" };

return arr[n - 1] + "";

}

else if (n >= 20 && n <= 99)

{

var arr = new string[] { "Twenty", "Thirty", "Forty", "Fifty", "Sixty", "Seventy", "Eighty", "Ninety" };

return arr[n / 10 - 2] + "" + AmountInWords(n % 10);

}

else if (n >= 100 && n <= 199)

{

return "OneHundred" + AmountInWords(n % 100);

}

else if (n >= 200 && n <= 999)

{

return AmountInWords(n / 100) + "Hundred" + AmountInWords(n % 100);

}

else if (n >= 1000 && n <= 1999)

{

return "OneThousand" + AmountInWords(n % 1000);

}

else if (n >= 2000 && n <= 999999)

{

return AmountInWords(n / 1000) + "Thousand" + AmountInWords(n % 1000);

}

else if (n >= 1000000 && n <= 1999999)

{

return "OneMillion" + AmountInWords(n % 1000000);

}

else if (n >= 1000000 && n <= 999999999)

{

return AmountInWords(n / 1000000) + "Million" + AmountInWords(n % 1000000);

}

else if (n >= 1000000000 && n <= 1999999999)

{

return "OneBillion" + AmountInWords(n % 1000000000);

}

else

{

return AmountInWords(n / 1000000000) + "Billion" + AmountInWords(n % 1000000000);

}

}

static int NumberOfLetters(int n)

{

if (n == 4) return 0;

int len = 0;

int ans = 0;

while (true) {

if (len == 4)

{

break;

}

// Console.WriteLine(n+ " " + AmountInWords(n));

len = AmountInWords(n).Length;

n = len;

ans++;

}

return ans;

}

static void Main(string[] args)

{

//Console.WriteLine(AmountInWords(987654321));

Console.WriteLine(NumberOfLetters(4));

Console.ReadLine();

}

}

}