Given an array of integers, find the rightmost [round number](keyword://round-number) in it and output its position in the array (0-based). If there are no *round numbers* in the given array, output -1.

**Example**

* For inputArray = [0, 5, 10, 15], the output should be  
  rightmostRoundNumber(inputArray) = 2;
* For inputArray = [1, 2, 3, 4, 5], the output should be  
  rightmostRoundNumber(inputArray) = -1.

**Input/Output**

* **[execution time limit] 3 seconds (cs)**
* **[input] array.integer inputArray**

*Guaranteed constraints:*  
0 ≤ inputArray.length ≤ 10,  
0 ≤ inputArray[i] ≤ 104.

* **[output] integer**

**[C#] Syntax Tips**

// Prints help message to the console

// Returns a string

string helloWorld(string name) {

Console.Write("This prints to the console when you Run Tests");

return "Hello, " + name;

}

<https://codefights.com/challenge/npgAxzzMjLxhaXwSM/solutions>

static int rightmostRoundNumber(int[] inputArray)

{

for(int i = inputArray.Length-1;i>=0; i--)

{

if(inputArray[i]%10 ==0)

{

return i;

}

}

return -1;

}