**6 kyu**

**Simple Fun #91: Unique Digit Products**

2397% of 7119 of186[myjinxin2015](https://www.codewars.com/users/myjinxin2015)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/simple-fun-number-91-unique-digit-products/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/simple-fun-number-91-unique-digit-products/solutions/csharp)

[Forks (1)](https://www.codewars.com/kata/simple-fun-number-91-unique-digit-products/forks/csharp)

[Discourse (4)](https://www.codewars.com/kata/simple-fun-number-91-unique-digit-products/discuss/csharp)

* Add to Collection
* |
* Share this kata:

**Task**

Let's call product(x) the product of x's digits. Given an array of integers a, calculate product(x) for each x in a, and return the number of distinct results you get.

**Example**

For a = [2, 8, 121, 42, 222, 23], the output should be 3.

Here are the products of the array's elements:

2: product(2) = 2;

8: product(8) = 8;

121: product(121) = 1 \* 2 \* 1 = 2;

42: product(42) = 4 \* 2 = 8;

222: product(222) = 2 \* 2 \* 2 = 8;

23: product(23) = 2 \* 3 = 6.

As you can see, there are only 3 different products: 2, 6 and 8.

**Input/Output**

* [input] integer array a

Constraints:

1 ≤ a.length ≤ 10000,

1 ≤ a[i] ≤ 1000000000.

* [output] an integer

The number of different digit products in a.

<https://www.codewars.com/kata/simple-fun-number-91-unique-digit-products/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

//public static bool PrimeString(string s)

//{

// //coding and coding..

//}

public static int UniqueDigitProducts(int[] a)

{

//coding and coding..

HashSet<int> hs = new HashSet<int>();

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

{

int prod = 1;

int copia = a[i];

while(copia > 0)

{

int d = copia % 10;

prod \*= d;

copia /= 10;

}

hs.Add(prod);

}

return hs.Count;

}

static void Main(string[] args)

{

Console.ReadLine();

}

}

}