**Frequency of Students**

Attempted by: **820**

/

Accuracy: **81%**

/

Maximum Score: **20**

/

10 Votes

Tag(s):

Easy

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

There is a class consisting of **'N'** students . There can be many students with the same name.

Now, you have to print the names of the students followed by there frequency as shown in the sample explanation given below.

Output the names in the **lexicographical order.**

**Input :**

* First line contains an integer **'N'**, i.e the no. of students in the class.
* Next 'N' lines contains the names of the students.

**Output:**

* Each line consists of the name of student space and separated its frequency.

**Constraints:**

* 1<=N<=1000
* string length<=100
* string consists of lowercase letters

**Note :** For practicing use Map technique only .

**SAMPLE INPUT**

5

sumit

ambuj

himanshu

ambuj

ambuj

**SAMPLE OUTPUT**

ambuj 3

himanshu 1

sumit 1

**Explanation**

There are 5 students in a class, 3 students whose name is ambuj, 1 student having name himanshu and 1 student named sumit.

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**Bash, C, C++, C++14, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), TypeScript, Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift, Swift-4.1, Visual Basic

<https://www.hackerearth.com/practice/data-structures/hash-tables/basics-of-hash-tables/practice-problems/algorithm/frequency-of-students/description/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

//static int maxCuadrado(string[] s)

//{

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

// {

// for (int j = 0; j < s[i].Length; j++)

// {

// }

// }

//}

static void Main(string[] args)

{

int N = int.Parse(Console.ReadLine());

SortedDictionary<string, int> frec =

new SortedDictionary<string, int>();

while (N-- > 0)

{

string name = Console.ReadLine().Trim();

if(frec.ContainsKey(name)) frec[name]++;

else frec[name] = 1;

}

//var items = from pair in frec

// orderby pair.Value descending , pair.Key

// select pair;

foreach (KeyValuePair<string, int> kvp in frec )

{

Console.WriteLine(kvp.Key + " " + kvp.Value);

}

Console.ReadLine();

}

}

}