Duplicated Products

View Try Insights

Points: 50 13 test cases (0 samples)

Coding EASY Implementation Core CS Core Skills General Programming Problem Solving Algorithms Sets

Given *n* complex products, each with name, price and weight, find out how many duplicates of the original product are present within the products. Here, a duplicate is a product with all parameters, i.e. name, price and weight, equal to some other product.

Function Description

Complete the function *numDuplicates* in the editor below. The function has to return a single integer denoting the number of duplicates within the products.

numDuplicates has the following parameter(s):

names: string array of size *n*, where $names_i$ denotes the name of the i^{th} product
prices: int array of size *n*, where $prices_i$ denotes the price of the i^{th} product
weights: int array of size *n*, where $weights_i$ denotes the weight of the i^{th} product

Constraints

- $1 \leq n \leq 10^5$
- $names_i$ is non-empty, has at most 10 characters, and all its characters are lowercase english letters
- $1 \leq prices_i, weights_i \leq 1000$

▼ Input Format Format for Custom Testing

Input from stdin will be processed as follows and passed to the function:

In the first line, there is a single integer n.
Then, n lines follow. In the i^{th} of them, there is a single string $names_i$
In the next line, there is a single integer n.
Then, n lines follow. In the i^{th} of them, there is a single integer $prices_i$
In the next line, there is a single integer n.
Then, n lines follow. In the i^{th} of them, there is a single integer $weights_i$

▼ Sample Case 0

Sample Input

```
5
ball
box
ball
ball
box
5
2
2
2
2
2
5
1
2
1
1
3
```

Sample Output

```
2
```

Explanation

There are *5* products. All *3* balls are the same because they have same names, prices, and weights, so they contribute *2* duplicates. Two other products are boxes, and they are different because they have different weights.