Homework: Combining Data Structures

This document defines the **homework assignments** for the "Data Structures" course @ Software University.

Problem 1. Shopping Center

A **shopping center** keeps a set of **products**. Each product has **name**, **price** and **producer**. Your task is to model the shopping center and design a **data structure holding the products**. Write a program that executes **N** commands, given in the input (a single command at a line):

- **`AddProduct name;price;producer`** – adds a product by given name, price and producer. If a product with the same name / producer/ price already exists, the newly added product does not affect the existing ones (duplicates are allowed). As a result the command prints "**Product added**".
- **`DeleteProducts producer`** – deletes all products matching given producer. As a result the command prints "**X products deleted**" where **X** is the number of deleted products or "**No products found**" if no such products exist.
- **`DeleteProducts name;producer`** – deletes all products matching given product name and producer. As a result the command prints "**X products deleted**" where **X** is the number of deleted products or "**No products found**" if no such products exist.
- **`FindProductsByName name`** – finds all products by given product name. As a result the command prints a list of products in format **{name;producer;price}**, ordered by name, producer and price. Print each product on a separate line. If no products exist with the specified name, the command prints "**No products found**".
- **`FindProductsByProducer producer`** – finds all products by given producer. As a result the command prints a list of products in format **{name;producer;price}**, ordered by name, producer and price. You should print each product on a single line**.** If no products exist by the specified producer, the command prints "**No products found**".
- **`FindProductsByPriceRange fromPrice;toPrice`** – finds all products whose price is greater or equal than **fromPrice** and less or equal than **toPrice**. As a result the command prints a list of products in format **{name;producer;price}**, ordered by name, producer and price. You should print each product on a separate line. If no products exist within the specified price range, the command prints "**No products found**".

All string matching operations are **case-sensetive**.

Input

The input data should be read from the console.

- At the first line you will be given the number **N** of the commands.
- At each of the next **N** lines you will be given a command in the format described above.

The input data will always be valid and in the described format. There is no need to check it explicitly.

Output

The output data should be printed on the console.

The output should contain the output from each command from the input.

Constraints

- **N** will be between 1 and 50 000, inclusive.

- All strings specified in the commands (e.g. product names and producers) consist of alphabetical characters, numbers and spaces. Strings are case-sensitive.
- Prices are given as real numbers with up to 2 digits after the decimal point, (e.g. 133.58, 320.3, or 10)
- The '**.**' symbol is used as decimal separator.
- Prices should be printed with exactly **2 digits** after the decimal point (e.g. 320.30 instead of 320.3).
- Allowed working time for your program: **1.00 seconds** (at the judge environment).
- Allowed memory: **32 MB**.

Examples

Input Example	Output Example
17 AddProduct IdeaPad Z560;1536.50;Lenovo AddProduct ThinkPad T410;3000;Lenovo AddProduct VAIO Z13;4099.99;Sony AddProduct CLS 63 AMG;200000;Mercedes FindProductsByName CLS 63 AMG FindProductsByName CLS 63 FindProductsByName cls 63 amg AddProduct 320i;10000;BMW FindProductsByName 320i AddProduct G560;999;Lenovo FindProductsByProducer Lenovo DeleteProducts Lenovo FindProductsByProducer Lenovo FindProductsByPriceRange 100000;200000 DeleteProducts Beer;Ariana DeleteProducts CLS 63 AMG;Mercedes FindProductsByName CLS 63 AMG	Product added Product added Product added Product added {CLS 63 AMG;Mercedes;200000.00} No products found No products found Product added {320i;BMW;10000.00} Product added {G560;Lenovo;999.00} {IdeaPad Z560;Lenovo;1536.50} {ThinkPad T410;Lenovo;3000.00} 3 products deleted No products found {CLS 63 AMG;Mercedes;200000.00} No products found 1 products deleted No products found