# Exam "Fundamentals of programming with C#" – 21 January 2018

## Problem 3. Friends from Rainy Universe

## History

Hey, it’s been a while and your friend and some of his friends from the parallel universe are coming home and they have something very interesting for you. They found that thanks to the different types of raindrops there are also different types of water. They are coming with packs full of jars holding different types of water. However, there is a small problem... they have placed all the jars in only one bag and… this bag is going to arrive a week later… They don’t like to unpack everything because they are lazy, so they send you a message in which is described which jar to whom belongs. Your job is to create a machine which unpacks the bag and separates the jars in a pile for every person. How does the machine know how to separate the jars? **You must implement an algorithm which organizes the information from the message.**

## Description

You must read input lines until you receive the **termination command “End”.** On every input line (except the termination one) you will receive information in the following format:

**“{person name} -> {liquid name}: {jars count}”**

You must **split** the input and **extract** the needed information. Everything will be **valid** in the format described in the Input section.

You must **collect** all the **different** **types** of **liquids** for a **person** and **sum the count** of jars of every **liquid** for **every** **person**.

### Input

The information will be in the following format:

* Person name – **unique** string which will **not contain** any of the characters that you have to split by (space, ‘-’, ‘>’, ‘:’)
* Liquid name – **unique** string which will **not contain** any of the characters that you have to split by (space, ‘-’, ‘>’, ‘:’)
* Jars count – **integer** number in range [0 … 100]

### Output

You must print **all liquids** for a **single person** with their **count of jars**.

**Order** the **people by** their **name (ascending)** and the **liquids by** their **count (ascending).**

{Person Name} Liquids:

--- {Liquid Name}: {Count}

--- {Liquid Name}: {Count}

…

Examples:

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Description** |
| Morgana -> Truelock: 2  Morgana -> Truelock: 89  Garrard -> Perigeaux: 96  Morgana -> Tott: 36  Arleta -> Sinnie: 15  Arleta -> Hutchin: 21  Emlyn -> Winslett: 92  Arleta -> Eam: 68  End | Arleta Liquids:  --- Sinnie: 15  --- Hutchin: 21  --- Eam: 68  Emlyn Liquids:  --- Winslett: 92  Garrard Liquids:  --- Perigeaux: 96  Morgana Liquids:  --- Tott: 36  --- Truelock: 91 | **Arleta is first because of the ordering (by names). The liquids are ordered by count of jars (ascending).** |
| **Input** | **Output** |
| Dino -> Wasiel: 100  Hallie -> Bellefonte: 50  Hallie -> Bellefonte: 25  Hallie -> Bellefonte: 25  Dino -> Banana: 94  Dino -> Banana: 6  End | Dino Liquids:  --- Wasiel: 100  --- Banana: 100  Hallie Liquids:  --- Bellefonte: 100 |