**More Exercise: Dictionaries**

* Additional exercises for the [Python Fundamentals Course @SoftUni](https://softuni.bg/trainings/2442/python-fundamentals-september-2019).   
  **Snowwhite**

Snow White loves her dwarfs, but there are so many and she doesn't know how to order them. Does she order them by name? Or by color of their hat? Or by physics? She can't decide, so its up to you to write a program that does it for her.

You will be receiving **several input lines** which contain **data** about **dwarfs** in the following format:

**{dwarfName} <:> {dwarfHatColor} <:> {dwarfPhysics}**

The **dwarfName** and the **dwarfHatColor** are **strings**. The **dwarfPhysics** is an **integer**.

You must **store** the **dwarfs** in your program. There are several rules though:

* If **2 dwarfs** have the **same name** but **different color**, they should be **considered different dwarfs**, and you should store **both** of them.
* If **2 dwarfs** have the **same name** and the **same color**, **store** the **one** with the **higher physics**.

When you receive the command "**Once upon a time**", the input ends. You must **order** the **dwarfs** by **physics** in **descending order** and thenby **total** **count** of **dwarfs** with the **same hat color** in **descending order**.   
Then you must print them all.

**Input**

* The input will consists of **several input lines**, containing **dwarf data** in the format, specified above.
* The input **ends** when you receive the command "**Once upon a time**".

**Output**

* As output you must print the **dwarfs**, **ordered** in the way , specified above.
* The output format is: **({hatColor}) {name} <-> {physics}**

**Constraints**

* The **dwarfName** will be a **string** which may contain **any ASCII** character except '' (**space**), '**<**', '**:**', '**>**'.
* The **dwarfHatColor** will be a **string** which may contain **any ASCII** character except '' (**space**), '**<**', '**:**', '**>**'.
* The **dwarfPhysics** will be an **integer** in **range [0, 231 – 1]**.
* There will be **no invalid** input lines.
* If **all sorting criteria fail**, the order should be by **order** of **input**.
* Allowed working **time** / **memory**: **100ms** / **16MB**.

**Examples**

|  |  |
| --- | --- |
| **Input** | **Output** |
| Pesho <:> Red <:> 2000  Tosho <:> Blue <:> 1000  Gosho <:> Green <:> 1000  Sasho <:> Yellow <:> 4500  Prakasho <:> Stamat <:> 1000  Once upon a time | (Yellow) Sasho <-> 4500  (Red) Pesho <-> 2000  (Blue) Tosho <-> 1000  (Green) Gosho <-> 1000  (Stamat) Prakasho <-> 1000 |
| Pesho <:> Red <:> 5000  Pesho <:> Blue <:> 10000  Pesho <:> Red <:> 10000  Gosho <:> Blue <:> 10000  Once upon a time | (Blue) Pesho <-> 10000  (Blue) Gosho <-> 10000  (Red) Pesho <-> 10000 |