Problem 2. On the Way to Annapurna

You've hired a Sherpa and he has a list of supplies you both need to go on the way. He has passed you some notes and you have to order them correctly in a diary before you start circling around the town's stores.

Create a program, that lists **stores** and the **items** that can be found in them. You are going to be receiving **commands** with the information you need until you get the "**End**" command. There are **three possible commands**:

- "**Add**->{Store}->{Item}"
  - **Add** the **store** and the **item** in your diary. If the store already **exists**, add just the item.
- "**Add**->{Store}->{Item},{Item1}…,{ItemN}"
  - **Add the store and the items to** your notes. **If the store already exists** in the diary – **add just the items** to it.
- "**Remove**->{Store}"
  - **Remove the store** and its items from your diary, **if it exists**.

In the end, print the collection **sorted by the count of the items** in **descending order** and **then by the names of the stores**, again, **in descending order** in the following format:

Stores list:

{Store}

<<{Item}>>

<<{Item}>>

<<{Item}>>

Input / Constraints

- You will be receiving information until the "**END**" command is given.
- There will always be **at least one** store in the diary.
- Input will always be **valid**, there is no need to check it explicitly.

Output

- Print the list of stores in the format given above.

Examples

Input	Output

	Stores list:
Add->PeakSports->Map,Navigation,Compass	PeakSports
Add->Paragon->Sunscreen	<<Map>>
Add->Groceries->Dried-fruit,Nuts	<<Navigation>>
Add->Groceries->Nuts	<<Compass>>
Add->Paragon->Tent	Groceries
Remove->Paragon	<<Dried-fruit>>
Add->Pharmacy->Pain-killers	<<Nuts>>
END	<<Nuts>>
	Pharmacy
	<<Pain-killers>>

Comments
First, we receive the "**Add**" command with a couple of items and we have to add the store and the items to. We keep doing that for each line of input and when we receive the "**Remove**" command, we delete the store and its items from our records. In the end we print the stores sorted by the **count** of their **items** and **then by** their **names**.

	Stores list:
Add->Peak->Waterproof,Umbrella	Peak
Add->Groceries->Water,Juice,Food	<<Waterproof>>
Add->Peak->Tent	<<Umbrella>>
Add->Peak->Sleeping-Bag	<<Tent>>
Add->Peak->Jacket	<<Sleeping-Bag>>
Add->Groceries->Lighter	<<Jacket>>
Remove->Groceries	
Remove->Store	
END	