# Auto Service

*Your father has his own business. One day you will inherit that business and you need to learn fast. That's why he decides to test if you can you handle the duties...*

You will be given an **array** of **strings**. Every **string** represents some action depending on the given command. The possible **commands** are as follows: **instructions**, **addPart** and **repair**.

* When you receive a string that starts with "**instructions"**, for example:   
  "**instructions bmw**", that means your auto service receives the instructions which can be **used to repair** the given **car** **model**. If you receive a car model, which you have **no instructions for**, that means you **can't repair** that car.
* If you receive a string that starts with "**addPart**", for example:  
  **"addPart bmw engine GV1399SSS**", that means you receive a **specific** **part** for a **car** **model**. In the example above, you receive an **engine** for **bmw** with **serial** **number**: **GV1399SSS**. You must **store all received parts** that you get.  
  Note that you can get **more than one** part for a specific car model and you can get the **same part multiple times**.
* When you receive a string that starts with "**repair**", that means some client needs your service. Here is an example:  
  "**repair audi {"engine":"broken","transmission":"OP8766TRS"}**"

That string contains a **car** **model** and some **car** **specifications** in the form of a **stringified** **object**. First, you must **check** if you have **instructions** to repair that car model. If you do, check which **parts** (keys) for that **car specifications** (object) refer to "**broken**". If there are such, that means that **part** is **broken** and needs to be **replaced**/**repaired**. You must check if you have that part for that car model. If you do, just fix it (**change the broken state with the** **part's** **serial** **number**) and **remove** the **used** **part** from the **parts**. In the moment where you can replace/fix some part take the first part from the collection.

### Input

You will receive an **array** of **strings**

### Output

After every **repair** **command** you must return the **current** **car** **state** to the client in the following format:

**"{model} client - {"{part}":"{partState}","{part}":"{partState}..."}"**

* **The object that represents the car specification must be stringified.**
* **If you get a car model for which you have no instructions:  
  "{model} is not supported"**

**The result must be shown whether the car's part has been successfully changed or not.**

When all given commands are executed, print the available parts **sorted** **alphabetically** by **car** **model** in the following format:

"**{model} – {{part}:[serial numbers...],{part}:[serial numbers...]...}**"  
...

* The object that hold **all parts** for each car model also must be **stringified**.

### Example

|  |  |  |
| --- | --- | --- |
| ****Input**** | ****Output**** | ****Comment**** |
| **[**  **'repair mazda {"engine":"broken"}',**  **'instructions bmw',**  **'addPart opel engine GV1399SSS' ]** | **mazda is not supported**  **opel - {"engine":["GV1399SSS"]}** | **We do not have the mazda instructions. After that we receive the bmw instructions, but that’s a different model. We will be able to try to "repair" that model if we receive that model instructions before the repair.** |
| **[  'instructions bmw',**  **'addPart opel engine GV1399SSS',  'addPart opel transmission SMF556SRG',**  **'addPart bmw engine GV1399SSS',**  **'addPart bmw transmission SMF444ORG',**  **'addPart opel transmission SMF444ORG',**  **'instructions opel',**  **'repair opel {"engine":"broken","transmission":"OP8766TRS"}',**  **'repair bmw {"engine":"ENG999FPH","transmission":"broken","wheels":"broken"}'**  **]** | **opel client - {"engine":"GV1399SSS","transmission":"OP8766TRS"}**  **bmw client - {"engine":"ENG999FPH","transmission":"SMF444ORG","wheels":"broken"}**  **bmw - {"engine":["GV1399SSS"],"transmission":[]}**  **opel - {"engine":[],"transmission":["SMF556SRG","SMF444ORG"]}** | **We receive bmw instructions, that means if we receive a repair with bmw model we will be able to try to "repair" it.**  **After that we receive 5 parts: opel engine, opel transmission, bmw engine, bmw transmission and opel transmission. We store all serial numbers.**  **After that we receive another instructions, this time for opel.**  **The last 2 strings from the input are clients (repair case).**  **The first one is opel with broken engine. We have opel instructions and backup opel engine, so we fix it/replace it.**  **The second car model is bmw. We also have these instructions, so we check the car specifications. The broken ones are transmission and wheels. We have bmw transmission but not wheels. We change only the parts that we have.** |