

With *Jet Smart Cart* the more items you buy, the more you save. The algorithm behind how this works is a bit complicated, and since it's your first day at Jet you decide to ask one of your co-workers for more information. In return, you offer to implement a new *cart update* algorithm for them. The update algorithm works like this:

Whenever a new customer visits jet.com, their shopping cart is initially empty. Once the customer starts shopping, the cart can receive any of the following requests:

- `add : <item_name>`: the `<item_name>` item was added to the cart;
- `remove : <item_name>`: the `<item_name>` item was removed from the cart;
- `quantity_upd : <item_name> : <value>`: the `<item_name>` quantity in the cart was changed by `<value>`, which is an integer in the format `+a` or `-a`;
- `checkout`: the customer has paid and the cart is now empty.

Given a list of requests in the formats described above, return the state of the cart after they have been processed.

#### Example

- For

```
requests = ["add : milk",  
            "add : pickles",  
            "remove : milk",  
            "add : milk",  
            "quantity_upd : pickles : +4"]
```

the output should be

```
shoppingCart(requests) = ["pickles : 5", "milk : 1"];
```

- For

```
requests = ["add : rock",  
            "add : paper",  
            "add : scissors",  
            "checkout",  
            "add : golden medal"]
```

the output should be

```
shoppingCart(requests) = ["golden medal : 1"].
```

#### Input/Output

- **[time limit] 4000ms (py)**

- **[input] array.string requests**

Array of strings, where each string is a request in one of the formats described above. The following holds true:

- the add request item is guaranteed not to be present in the cart;
- the remove request item is guaranteed to be present in the cart;
- the quantity\_upd request item is guaranteed to be present in the cart, and the quantity of the item is guaranteed to remain more than 0 after the update.

- **[output] array.string**

Array of elements in the cart once all requests have been processed in the order they are received. Each string should be formatted as `<item_name> : <item_quantity>`, where `<item_name>` is the name of the item, and `<item_quantity>` is its quantity.