01. Agency

….After the agency have choose their new agents, they have to get their ID numbers.

Every ID number is on a **separate card** o n the desk in a stack.

The agents enter at the registation in queue **one by one.** Each agent **take topmost ID**.

Your task is to write a program that can handle the registration process. On the **first line** you will receive ID numbers available for the registration, **separated by white space.** On the **second** - agents who enter the registration room, also **separated by white space.** After this you start to receive **commands, until** you get the word **"stop",** that you should process to make sure, that every agent get the intended ID number. **Commands,** that you can receive is :

- **"add-ID {ID Number}"-** after the command you will receive an **ID number**, that you should add on the top of the stack with ID's. ***Example:*** "add-ID 2312".
- **"add-agent {Agent Name}" –** You should add an **agent** at the end of the queue. ***Example:*** "add-agent Johnson"
- **"remove-ID" –** you have to remove first **ID number** from the stack, and print the message **"{ID Number} has been removed. "**
- **"remove-agent"-** remove the **last agent** from the queue and print the message**"{Agent Name} has left the queue."**
- **"sort-ID"-** if you receive sort command you have to sort the stack of ID numbers in **descending order.**

Input

The input data should be read from the console.

- **First line** – ID numbers (as string), separated by **whitespace**.
- **Second line** – agent's names, separated by **whitespace.**
- Strings with commands until the **"stop"** command.

The input will always be valid and in the format described. There is no need to check it explicitly.

Output

When you receive **"stop"** command you should print at the console:

- for every agent:

"{Agent Name} takes ID number: {ID Number}"

- If there are **NO** more agents:

Print message **- "No more agents left."**

And then print every ID numbers left in the stack in following format:

"ID number: {ID Number}"

- If there are **NO** ID's left:

Print for every agent left in the queue: **"{Agent Name} does not have an ID."**

Constraints

- The **input text** will be in the range [1 … 100 chars]
- The **input numbers** will be a 32-bit integer in the range [0 … 2 147 483 647].
- Allowed working time for your program: 0.1 seconds.
- Allowed memory: 16 MB.

Examples

Input	Output	Comments
111 222 333 Alpha Bravo Charlie add-ID 444 add-agent Delta stop	Alpha takes ID number: 444 Bravo takes ID number: 333 Charlie takes ID number: 222 Delta takes ID number: 111	Agent Alpha **takes** the topmost id,which is 444 because it has been added last. Agent Delta was also added to the queue, so he receive an ID as well.
223 8275 9223 Delta India Foxtrot remove-ID stop	9223 has been removed. Delta takes ID number: 8275 India takes ID number: 223 Foxtrot does not have an ID.	After the "remove-ID", we remove 9223,from the stack, and print the message.Now we have three agents and only two ID's, so we print the appropriate message
008 003 002 001 Widow Delta sort-ID add-agent Zulu remove-ID remove-agent stop	008 has been removed. Zulu has left the queue. Widow takes ID number: 003 Delta takes ID number: 002 No more agents left. ID number: 001	