

Problem 2 – Hornet Comm

The Hornet Ex-King – Horny, has established an innovative technology providing communication for his fellow hornets, called Hornet Comm. Inc. Hornet Comm. provides functionality from private messages to wide broadcasts.

You will be given data of several tracked comm. channels, which you must decrypt. The input data will come in the following format:

{firstQuery} <-> {secondQuery}

If the **first query** consists **only of digits** and the **second one** consists of **digits and / or letters**, it is a **private message**.

If the **first query** consists of **anything but digits**, and the **second one** consists of **letters and / or digits**, it is a **broadcast**.

Any input that **does NOT follow** the format, specified above, should be **IGNORED**.

If the **given data** is a **private message**, the first query is the **recipient's code**, and the second query is the **message**. You must **reverse** the **recipient's code** and **store** it along with the message.

If the **given data** is a **broadcast**, the first query is the message, and the second query is the **frequency**. You must take the **frequency** and make **all capital letters – small** and **all small letters – capital**. Then you must **store** it, along with the **message**.

You must **keep** all input broadcasts and messages after you **decrypt** them.

When you receive the command **"Hornet is Green"**, the input sequences **ends**, and you must print the stored broadcasts and messages.

Input

- The input comes in the form of several input lines in the format specified above.
- The input ends when you receive the command **"Hornet is Green"**.

Output

- As output you must print all broadcasts and messages, printing first the broadcasts, in the following format:
 - **Broadcasts:**
 - **{frequency} -> {message}**
 - . . .
 - **Messages:**
 - **{recipient} -> {message}**
 - . . .
- If there are **no messages**, or **no broadcasts**, print **"None"** in their place.

Constraints

- The input lines may consist of **any ASCII** character.
- There will be **NO** more than 1000 lines of input.
- **All data** must be printed in **order of input**.

Examples

Input	Output
213094 <-> BeeQueenDown 213094 <-> Repeat Foxtrot <-> 123321 213094 <-> BeeQueenDown Unicorn <-> 871203 Charlie <-> 56210 Kilo <-> 423211 Hornet is Green	Broadcasts: 123321 -> Foxtrot 871203 -> Unicorn 56210 -> Charlie 423211 -> Kilo Messages: 490312 ->BeeQueenDown 490312 ->Repeat 490312 ->BeeQueenDown
<+>.<+> <-> az13b6 <->.<-> <-> P2Z4x789 12345 <-> Pr1v@teM3ssage Hornet is Green	Broadcasts: AZ13B6 -> <+>.<+> p2z4X789 -> <->.<-> Messages: None