C — Updating a Dictionary

Input: Standard Input
Output: Standard Output

In this problem, a dictionary is collection of key-value pairs, where keys are lower-case letters, and values are non-negative integers. Given an old dictionary and a new dictionary, find out what were changed.

Each dictionary is formatting as follows:

```
{key:value,key:value,...,key:value}
```

Each key is a string of lower-case letters, and each value is a non-negative integer without leading zeros or prefix '+'. (i.e. -4, 03 and +77 are illegal). Each key will appear at most once, but keys can appear in any order.

Input

The first line contains the number of test cases T ($T<=1000$). Each test case contains two lines. The first line contains the old dictionary, and the second line contains the new dictionary. Each line will contain at most 100 characters and will not contain any whitespace characters. Both dictionaries could be empty.

WARNING: there are no restrictions on the lengths of each key and value in the dictionary. That means keys could be really long and values could be really large.

Output

For each test case, print the changes, formatted as follows:

l   First, if there are any new keys, print '+' and then the new keys in increasing order (lexicographically), separated by commas.
l   Second, if there are any removed keys, print '-' and then the removed keys in increasing order (lexicographically), separated by commas.
l   Last, if there are any keys with changed value, print '*' and then these keys in increasing order (lexicographically), separated by commas.

If the two dictionaries are identical, print 'No changes' (without quotes) instead.

Print a blank line after each test case.

Sample Input

```
3
{a:3,b:4,c:10,f:6}
{a:3,c:5,d:10,ee:4}
{x:1,xyz:123456789123456789123456789}
{xyz:123456789123456789123456789,x:1}
{first:1,second:2,third:3}
{third:3,second:2}
```

Output for Sample Input

```
+d,ee
-b,f
*c

No changes

-first
```