Mike is an avid traveler. He has visited a lot of cities over many years. Whenever he visits a city, he takes a few photos and saves them on his computer. Each photo has a name with an extension ("jpg", "png' or "jpeg") and there is a record of the name of the city where the photo was taken and the time and date the photo; for example: "photo.jpg, Krakow, 2013-09-05 14:08:15". Note that, in some rare cases, photos from different locations may share the time and date, due to timezone differences. Mike notices that his way of filing photos on his computer has become a mess. He wants to reorganize the photos.

First he decides to group the photos by city, then, within each such group, sort the photos by the time they were taken and finally assign consecutive natural numbers to the photos, starting from 1. Afterwards he intends to rename all the photos. The new name of each photo should begin with the name of the city followed by the number already assigned to that photo. The number of every photo in each group should have the same length (equal to the length of the largest number in this group); thus, Mike needs to add some leading zeros to the numbers. The new name of the photo should end with the extension, which should remain the same. Your task is to help Mike by finding the new name of each photo. Each of Mike's photos has the format: '<\<photoname>>.<\<extension>>, <<city_name>>, yyyy-mm-dd, hh:mm:ss", where '<<photo_name>>', '<\<extension>>' and, '<<city_name>>' consist only of letters of the English alphabet.

Write a function that, given a string representing the list of `M` photos, returns the string representing the list of the new names of all photos (the order of photos should stay the same).

```
def solution(s)
end
```

For example, given a string:
```
"photo.jpg, Krakow, 2013-09-05 14:08:15
Mike.png, London, 2015-06-20 15:13:22
myFriends.png, Krakow, 2013-09-05 14:07:13
Eiffel.jpg, Florianopolis, 2015-07-23 08:03:02
pisatower.jpg, Florianopolis, 2015-07-22 23:59:59
BOB.jpg, London, 2015-08-05 00:02:03
notredame.png, Florianopolis, 2015-09-01 12:00:00
me.jpg, Krakow, 2013-09-06 15:40:22
a.png, Krakow, 2016-02-13 13:33:50
b.jpg, Krakow, 2016-01-02 15:12:22
c.jpg, Krakow, 2016-01-02 14:34:30
d.jpg, Krakow, 2016-01-02 15:15:01
e.png, Krakow, 2016-01-02 09:49:09
f.png, Krakow, 2016-01-02 10:55:32
```

g.jpg, Krakow, 2016-02-29 22:13:11"
```

your function should return:

```
Krakow02.jpg
London1.png
Krakow01.png
Florianopolis2.jpg
Florianopolis1.jpg
London2.jpg
Florianopolis3.png
Krakow03.jpg
Krakow09.png
Krakow07.jpg
Krakow06.jpg
Krakow08.jpg
Krakow04.png
Krakow05.png
Krakow10.jpg
```

As there are ten photos taken in Krakow (numbered from 01 to 10). two photos in London (numbered from 1 to 2) and three photos in Florianopolis (numbered from 1 to 3). The new names of the photos are returned in the same order as in the given string.

Assume that:
* `M` is an integer within the range (1..100);
* Each year is an integer within the range (2000..2020);
* Each line of the input string is of the format '<\<photoname>>.<\<extension>>, <<city_name>>, yyyy-mm-dd hh:mm:ss' and lines are separated with newline characters;
* Each photo name (without extension) and city name name consists only of at least 1 and at most 20 letters from the English alphabet;
* Each name of the city starts with a capital letters and is followed by lower case letters;
* No two photos from the same location share the same date and time;
* Each extension is "jpg", "png" or "jpeg". In your solution, focus on correctness.
```