Problem 3 – Exam Score

We are given a table of **students** with **exam score** and **grades** in the following form:

```
-------------------------------------------
| Name             | Exam Score | Grade |
-------------------------------------------
| Peter Ivanov     | 306        | 5.26  |
| George Stefanov  | 120        | 3.12  |
| Maria Petrova    | 400        | 6.00  |
| Petya Georgieva  | 400        | 6.00  |
| Diana Kirova     | 120        | 3.23  |
| Darin Mihaylov   | 400        | 5.00  |
-------------------------------------------
```

Write a program to aggregate the exam score data and print for each **exam score** all **students**, which have that score and the **average grade** for these students. Use the following format **<score> -> [<student₁>, <student₂>, …]; avg=<avg grade>**". Order the **score in ascending order**. Order the **students alphabetically**. Print the average grade rounded with exactly 2 digits after the decimal point. In our example, the output should be the following:

```
120 -> [Diana Kirova, George Stefanov]; avg=3.18
306 -> [Peter Ivanov]; avg=5.26
400 -> [Darin Mihaylov, Maria Petrova, Petya Georgieva]; avg=5.67
```

Input

The input comes from the console. At the first 3 lines stays the header of the form that don't have important information for you, followed by a few lines holding exam information in format **<name> | <score> | <grade>**, separated by whitespaces and pipes. Student names are **unique**. The last line is the footer and consists of '**-**' only. The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

Print for each **exam score** (increasingly) all **students** (alphabetically), which have that score and the **average grade** for these students in the above described format (see also the examples).

Constraints

- The **count** of the input lines is in the range [5…1000] including the table header and borders.
- The **<score>** is an integer in the range [0…400].
- The **<name>** consists of only of **Latin characters and spaces**, with length of [1…50].
- The **<grade>** is a number number in the range [2.00…6.00].
- Time limit: 0.3 sec. Memory limit: 16 MB.

Examples

Input	Output
``---	Name	Exam Score	Grade	---	George Ivanov	306	5.26		George Stefanov	120	3.12		Petya Koleva	400	6.00		Aleksandar Stoyanov	300	5.00		Diana Kirova	120	3.23		Ivan Ivanov	0	2.00		Kalin Petrov	300	5.40		Stoyan Kotsev	400	5.00		Krasimir Mihaylov	400	5.98	---```	```0 -> [Ivan Ivanov]; avg=2.0 120 -> [Diana Kirova, George Stefanov]; avg=3.18 300 -> [Aleksandar Stoyanov, Kalin Petrov]; avg=5.20 306 -> [George Ivanov]; avg=5.26 400 -> [Krasimir Mihaylov, Petya Koleva, Stoyan Kotsev]; avg=5.66```