|  |  |
| --- | --- |
| Time limit | 12 seconds |
| Memory limit | 256Mb |
| Input | standard input or trades.csv |
| Output | standard output |

You are given a content of CSV-file with information about set of trades. It contains the following columns:

* TIME - Timestamp of a trade in format *Hour:Minute:Second.Millisecond*
* PRICE - Price of one share
* SIZE - Count of shares executed in this trade
* EXCHANGE - The exchange that executed this trade

For each exchange find the one minute-window during which the largest number of trades took place on this exchange.

**Note that:**

* You need to send source code of your program.
* You have only 25 attempts to submit a solutions for this task.
* You have access to all standart modules/packages/libraries of your language. But there is no access to additional libraries (numpy in python, boost in c++, etc).

Input format

Input contains several lines. You can read it from standart input or file “trades.csv”

Each line contains information about one trade: TIME, PRICE, SIZE and EXCHANGE. Numbers are separated by comma.

Lines are listed in ascending order of timestamps. Several lines can contain the same timestamp.

Size of input file does not exceed 5 MB.

See the example below to understand the exact input format.

Output format

If input contains information about *k* exchanges, print *k* lines to standart output.

Each line should contain the only number — maximum number of trades during one minute-window.

You should print answers for exchanges in lexicographical order of their names.

Sample

| **Input** | **Output** |
| --- | --- |
| 09:30:01.034,36.99,100,V  09:30:55.000,37.08,205,V  09:30:55.554,36.90,54,V  09:30:55.556,36.91,99,D  09:31:01.033,36.94,100,D  09:31:01.034,36.95,900,V | 2  3 |

Notes

In the example four trades were executed on exchange “V” and two trades were executed on exchange “D”. Not all of the “V”-trades fit in one minute-window, so the answer for “V” is three.