

The Problem

Webhosting company provides customer support via email. They record reply waiting time, type of question, category, and service. To improve customer satisfaction, they need an analytical tool to evaluate these data.

Input

The company provides 10 different services, each with 3 variations. Questions are divided into 10 types, each can belong to 20 categories, a category can have 5 subcategories.

First line contains count S (≤ 100.000) of all lines.

Every line starts with character **C** (waiting timeline) or **D** (query).

Waiting timeline:

```
C service_id[.variation_id] question_type_id[.category_id.[sub-category_id]] P/N date time
```

Values in square brackets are optional.

service_id[.variation_id] - service 9.1 represent service 9 of variation 1.

question_type_id[.category_id.[sub-category_id]] - question type 7.14.4 represent question type 7 category 14 and sub-category 4.

P/N - response type 'P' (first answer) or 'N' (next answer).

date - response date format is DD.MM.RRRR, for example, 27.11.2012 (27.november 2012).

time - time in minutes represent waiting time.

Query line:

```
D service_id[.variation_id] question_type_id[.category_id.[sub-category_id]] P/N date_from[-date_to]
```

Represent query, it prints out the **average waiting time** of all records matching specific criteria.

service_id and question_type_id can have special value "*", it means query match all services/question types. In the case of value "*", neither service variation nor service category/subcategory can be specified.

Output

Query line of type 'D' print out **average waiting time** rounded to minutes.

Only matching lines defined **before** the query line is counted.

It prints out "-" if the output is not defined.

Example

Input:

7

C 1.1 8.15.1 P 15.10.2012 83

C 1 10.1 P 01.12.2012 65

C 1.1 5.5.1 P 01.11.2012 117

D 1.1 8 P 01.01.2012-01.12.2012

C 3 10.2 N 02.10.2012 100

D 1 * P 8.10.2012-20.11.2012

D 3 10 P 01.12.2012

Output:

83

100

-

Explanation:

1.query (D 1.1 8 P 01.01.2012-01.12.2012) at line 5:

Valid only for 1.data line, because others have different question type.

1.data line has question type 8.15.1 which matches the query of question type 8.

Result: 83.

2.query (D 1 * P 8.10.2012-20.11.2012) at line 7:

Valid only for 1.data line and 3.data line.

2.data line doesn't match the date filter.

4.data line has a different response type (**D**).

5.data line doesn't match service, response type, and data filter.

Result: $(83+117)/2=100$.

3.query (D 3 10 P 01.12.2012) at line 8:

Doesn't match any data line.

Result: "-".

Solution

Your solution must be optimized for readability, flexibility, and good maintenance of source code.

You can choose Java, Scala, or PHP language, without external libraries.