

SWE2004: Principles in Programming (Spring 2013)

Programming Lab #3

Due-date : March 28th, 11:59 PM

Description

Korea Roads are famous for their sound engineering. Unfortunately, sound engineering does not come cheap, and some modern neo-Caesars have decided to recover the costs through automated tolling.

A particular toll highway, the CSLII, has a fare structure that works as follows: travel on the road costs a certain amount per km traveled, depending on the time of day when the travel begins. Cameras at every entrance and every exit capture the license numbers of all cars entering and leaving. Every calendar month, a bill is sent to the registered owner for each km traveled (at a rate determined by the time of day), plus one thousand won per trip. Your job is to prepare the bill for one month, given a set of license plate photos.

Input

The input begins with an integer $n < 100$ on a line by itself indicating the number of test cases, followed by a blank line. There will also be a blank line between each two consecutive test cases.

Each test case has two parts: the fare structure and the license photos. The fare structure consists of a line with 12 non-negative integers denoting the toll (won/km) from 00:00 to 01:59, the toll from 02:00 to 03:59, and so on for each hour in the day. Each photo record consists of the license number of the vehicle (5 alphanumeric characters), the time and date (2013-03-27 hh mm), the word enter or exit, and the location of the entrance or exit (in km from one end of the highway). All dates will be within a single month. Each ``enter" record is paired with the chronologically next record for the same vehicle, provided it is an ``exit" record. You may assume that no two records for the same vehicle have the same time. Times are recorded using a 24-hour clock. There are not more than 500 photo records.

Output

For each test case, print a line for each vehicle indicating the license number and the total bill in alphabetical order by license number. The output of two consecutive cases must be separated by a blank line.

Sample Input

```
1
100 100 200 200 150 150 150 200 300 200 150 100
ABC13 06 01 enter 15
765DE 22 00 exit 225
ABC13 08 03 exit 96
765DE 08 59 enter 32
```

Sample Output

```
765DE 29950 won
ABC13 17200 won
```

input	output
<p>1</p> <p>100 100 200 200 150 150 150 200 300 200 150 100 ABC13 06 01 enter 15 765DE 22 00 exit 225 ABC13 08 03 exit 96 765DE 08 59 enter 32</p>	<pre>v@v-Virtual-Machine ~/algorithm_training \$./a.out < lab4_input1.txt 765DE 29950 won ABC13 17200 won</pre>
<p>2</p> <p>120 120 220 220 170 170 180 200 280 180 150 100 DEF13 02 01 enter 11 567AB 22 10 exit 225 DEF13 08 03 exit 96 567AB 05 59 enter 24</p> <p>140 140 150 210 210 210 150 200 290 300 140 140 ABC13 13 59 exit 126 ABC13 23 01 exit 392 ABC24 07 00 exit 42 ABC13 15 03 enter 133 CD5AB 06 30 enter 15 ABC13 01 02 enter 5 ABC24 05 59 enter 14 CD5AB 17 30 exit 111</p>	<pre>v@v-Virtual-Machine ~/algorithm_training \$./a.out < lab4_input2.txt 567AB 45220 won DEF13 11200 won ABC13 70740 won ABC24 5200 won CD5AB 21160 won</pre>