110406   CDVII

Roman 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 CDVII, 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 dollar per trip, plus a two dollar account charge. Your job is to prepare the bill for one month, given a set of license plate photos.

Input

The input begins with an integer 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 24 non-negative integers denoting the toll (cents/km) from 00:00 to 00:59, the toll from 01:00 to 01:59, and so on for each hour in the day. Each photo record consists of the license number of the vehicle (up to 20 alphanumeric characters), the time and date (`mm:dd: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. Unpaired "enter" and "exit" records are ignored. 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 1,000 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

10 10 10 10 10 10 20 20 20 15 15 15 15 15 15 15 20 30 20 15 15 10 10 10
ABCD123 01:01:06:01 enter 17
765DEF 01:01:07:00 exit 95
ABCD123 01:01:08:03 exit 95
765DEF 01:01:05:59 enter 17
```

Sample Output

```
765DEF $10.80
ABCD123 $18.60
```