Problem 261: Timecard Helper

Difficulty: Medium

Author: Matt Marzin, King of Prussia, Pennsylvania, United States

Originally Published: Code Quest 2025

Problem Background

At Lockheed Martin, we take great pride in doing what's right. A large part of that involves accurately accounting for the time we spend working each day so that we can accurately bill our customers. As part of an agile development team, you'll most likely have multiple tasks assigned to you on any given day. On top of that, you may also have to help out other members of your team with what they're working on. Keeping track of how much time you spend on each task can get a little tricky when you bounce back and forth between different efforts all day.

Problem Description

For this problem, you've been tasked with making an application to help employees keep track of the hours they've worked on given tasks and summarize that information in a format they can easily enter into their timecards at the end of each day.

You will be provided with a list of task names and their associated charge codes, a 7-digit number that is entered on their time cards. The next input will be a list of task descriptions and how long an employee spent on each task. Sometimes, employees bounce back and forth between tasks, so you may see the same task multiple times. Also, some tasks may use the same charge number.

Sample Input

The first line of your program's input, **received from the standard input channel**, will contain a positive integer representing the number of test cases. Each test case will include:

- A line containing two positive integers, separated by a space:
  - T, representing the number of tasks an employee might be working on, and
  - E, representing the number of times an employee actually worked on a task
- T lines, each containing the name of a task (containing upper- and lower-case letters and spaces), a colon (:), and the task's associated charge number (a seven-digit number). Each task will appear in this list at most one time.
- E lines, each containing the name of a task (from the list provided previously), a colon (:), and a positive number representing the number of hours spent working on that task. A task may appear in this list multiple times.

```
3
6 5
Code App:5944035
Write Documentation:9556458
Attend Design Review:7653329
Deploy To Production:5499311
Attend Planning:6214339
Write Test Procedure:3270799
Code App:2.0
Attend Planning:2.0
Deploy To Production:4.5
Attend Planning:2.6
Write Documentation:5.7
3 10
Write Test Procedure:9543441
Attend Planning:9066150
Write Unit Tests:9589130
Attend Planning:3.7
Attend Planning:0.6
Write Unit Tests:5.9
Write Unit Tests:4.9
Write Test Procedure:1.7
Attend Planning:5.6
Write Unit Tests:0.9
Write Unit Tests:2.8
Write Test Procedure:4.0
Attend Planning:5.0
3 4
Deploy To Production:8871036
Attend Planning:2787693
Attend Design Review:4174936
Attend Design Review:3.3
Deploy To Production:4.9
Attend Design Review:1.6
Deploy To Production:1.1
```

Sample Output

For each test case, your program must print output reflecting what the employee should enter into their timecard.

If the total time worked does not exceed 24 hours, print one line for each of the employee's charge numbers, in increasing numeric order. Each line must contain the charge number, a colon (:), and the total time worked on tasks using that charge number. Print a single decimal place for the hours worked on each charge number.

However, if the total number of hours worked by the employee exceeds the number of hours in a day, print a single line containing the word "Error". In a case where an error occurs, do not print any lines of charge hours.

```
5499311:4.5
5944035:2.0
6214339:4.6
9556458:5.7
Error
4174936:4.9
8871036:6.0
```