I Ranking

Hosting a programming contest is fun, but it's also a lot of work. For instance, at the end of the day the jury will have to create a ranking of the teams based on their results during the contest. This can be tedious to do by hand, so we would like you to write a program for this task.

For the BAPC, the rules for the ranking are as follows:

- Teams are ranked according to the most problems solved; teams tied are ordered by increasing total time used.
- The time used for a problem is the number of minutes between the start of the contest and the first accepted run, plus a 20 minute penalty for each rejected run until the first accepted run.
- The total time used is the sum of the times used for each problem solved (as described above). Note that penalty time for problems for which no run was accepted does not count toward the total time used.
- If ties remain at the end of the contest, the point of comparison between tied teams will be the last point in time where their scores differed (e.g. if two teams are tied at the end of the contest, the team which solved their last problem earlier than the other team solved their last problem wins).

During the contest, teams submit solutions for problems, which are processed by the jury as *runs*. Each run has four properties:

- the submission time in minutes since the beginning of the contest (an integer between 1 and 300, inclusive);
- the name of the team that submitted the solution (a non-empty string of at most 20 lower-case letters);
- the identifier of the corresponding problem (an upper-case letter A through J, inclusive);
- the result as determined by the judging software (either *accepted* or $rejected$).

At the end of the contest, we have a list of runs available (ordered by non-decreasing submission times) and we want you to determine the final ranking of the teams.

Teams that are tied will share a position; those teams should be ordered alphabetically in the results.

Input

On the first line an integer t ($1 \leq t \leq 100$): the number of test cases. Then for each test case:

- one line with the number of teams t ($1 \leq t \leq 50$) and the number of runs r ($0 \leq r \leq 5\,000$), seperated by a single space;
- then t lines, each with the name of a team;
- then r lines, each with the description of a run: time, team, problem and result, formatted as described above, and seperated by a single space.

Output

For each test case:

- Print the ordered results: t lines, each with a rank (starting from 1), the team name, the number of correctly solved problems and the total penalty time, separated by single spaces.

Sample in- and output

Input	Output
1 8 28 twente utrecht groningen amsterdam eindhoven leiden delft nijmegen 5 utrecht B rejected 8 eindhoven F accepted 10 utrecht F accepted 17 utrecht B rejected 18 leiden C rejected 23 twente F rejected 25 utrecht B accepted 26 amsterdam D rejected 27 amsterdam D accepted 27 leiden C accepted 27 groningen F accepted 28 twente F rejected 30 nijmegen C rejected 30 nijmegen C accepted 30 delft B accepted 30 delft B rejected 33 twente F accepted 47 groningen D rejected 51 leiden D accepted 51 amsterdam C accepted 51 groningen D accepted 60 utrecht D accepted 65 utrecht J accepted 67 twente F rejected 70 twente F accepted 90 eindhoven D accepted 100 utrecht A rejected 101 utrecht C rejected	1 utrecht 4 200 2 groningen 2 98 3 amsterdam 2 98 3 leiden 2 98 5 eindhoven 2 98 6 delft 1 30 7 nijmegen 1 50 8 twente 1 73