Problem 1 - Scoreboard (scoreboard)

Managing a real-time scoreboard with lots of participants can be very difficult, so the Reply Code Masters need your help to analyse the platform logs and create the final rankings.

As you know, in the Reply Code Challenge - Teen Edition there are five problems, each one requires to solve 5 different inputs of increasing score (100, 200, 300, 400 and 500 respectively for inputs one, two, three, four and five). The points for a specific problem and input will be assigned once per team.

Using these rules it's possible for teams to tie. So to solve this problem, we'll need to take into account the time each teams takes to reach their score. In particular, each team will receive a *time penalty*, which equals the sum of the time taken to solve each input.

We need you to write a program that analyses the platform logs and ranks teams by decreasing score and, in the case of a tie, decreasing *penalty time*. If two or more teams have the same score and penalty time, then return them in numerical order.

A platform log has a specific format:

`timestamp` $\texttt{team}_{id}$ $\texttt{problem}_{id}$ $\texttt{input}_{id}$ `scored`

Where:

- `timestamp` is the time the submission was evaluated.
- $\texttt{team}_{id}$ is the team identifier (from 1 to N).
- $\texttt{problem}_{id}$ is the problem identifier (from 1 to 5).
- $\texttt{input}_{id}$ is the input identifier (from 1 to 5).
- `scored` is 1 if the submitted solution is valid, otherwise 0.

The points for a submission will be assigned only if scored is equal to 1.

The timestamp considered for the *penalty time* will be only the first-scored submission for a specific team_{id}, problem_{id} and input_{id}.

Example:

Given $N = 5$ teams and the following $L = 5$ logs:

`timestamp=1` $\texttt{team}_{id}$`=1` $\texttt{problem}_{id}$`=1` $\texttt{input}_{id}$`=1 scored=0`
`timestamp=2` $\texttt{team}_{id}$`=1` $\texttt{problem}_{id}$`=1` $\texttt{input}_{id}$`=1 scored=1`
`timestamp=2` $\texttt{team}_{id}$`=4` $\texttt{problem}_{id}$`=1` $\texttt{input}_{id}$`=2 scored=1`
`timestamp=3` $\texttt{team}_{id}$`=2` $\texttt{problem}_{id}$`=1` $\texttt{input}_{id}$`=2 scored=0`
`timestamp=4` $\texttt{team}_{id}$`=2` $\texttt{problem}_{id}$`=1` $\texttt{input}_{id}$`=1 scored=1`

The final ranking will be:

- Team 4 with 200 points and 2 penalty time.
- Team 1 with 100 points and 2 penalty time.
- Team 2 with 100 points and 4 penalty time.
- Team 3 with 0 points and 0 penalty time.
- Team 5 with 0 points and 0 penalty time.

Input data

The first line of the input file contains an integer **T**, the number of test cases to solve, followed by **T** testcases, numbered from **1** to **T**.

In each test case, the first line contains the two integers **N** and **L**, the number of teams and logs.

The following **L** lines contain the log information as described in the problem statement, one per line.

Output data

The output file must contain **T** lines. For each test case in the input file, the output file must contain a line with the words:

Case #t: $r_1\ r_2\ \ldots\ r_n$

where t is the test case number (from **1** to **T**) and $r_1\ r_2\ \ldots\ r_n$ are the team ids according to the final ranking.

Constraints

- $1 \leq \mathrm{T} \leq 20$.
- $2 \leq \mathrm{N} \leq 5.000$.
- $2 \leq \mathrm{L} \leq 100.000$.
- $1 \leq \mathrm{timestamp}_{id} \leq 10.000$.
- $1 \leq \mathrm{team}_{id} \leq \mathrm{N}$.
- $1 \leq \mathrm{problem}_{id}, \mathrm{input}_{id} \leq 5$.
- $0 \leq \mathrm{scored} \leq 1$.
- The logs are increasingly sorted by timestamp.

Scoring

- **input 1** : $\mathbf{T} = 1$, $\mathbf{N} \leq 5$ and $\mathbf{L} \leq 10$.
- **input 2** : $\mathbf{T} = 5$, $\mathbf{N} \leq 10$ and $\mathbf{L} \leq 100$.
- **input 3** : $\mathbf{T} = 10$, $\mathbf{N} \leq 100$ and $\mathbf{L} \leq 1.000$.
- **input 4** : $\mathbf{T} = 15$, $\mathbf{N} \leq 1.000$ and $\mathbf{L} \leq 10.000$.
- **input 5** : $\mathbf{T} = 20$, $\mathbf{N} \leq 5.000$ and $\mathbf{L} \leq 100.000$.

Examples

input	output
1 5 5 1 1 1 1 0 2 1 1 1 1 2 4 1 2 1 3 2 1 2 0 4 2 1 1 1	Case #1: 4 1 2 3 5