Countdown

Ann Sister owns a genealogical database service, which maintains family tree history for her clients. When clients login to the system, they are presented with a variety of services: searching, printing, querying, etc. One recent question that came up which the system was not quite prepared for was the following: "Which member of my family had the most grandchildren?" The client who posed this question eventually had to answer it by manually searching the family tree database herself. Ann decided to have software written in case this question (or ones similar to it asking for great-grandchildren, or great-great-grandchildren, etc.) is asked in the future.

Input

Input will consist of multiple test cases. The first line of the input will contain a single integer indicating the number of test cases. Each test case starts with a single line containing two positive integers n and d, where n indicates the number of lines to follow containing information about the family tree, and d indicates the specific question being asked about the tree: if d = 1, then we are interested in persons with the most children (1 generation away); if d = 2, then we are interested in persons with the most grandchildren (2 generations away), and so on. The next n lines are of the form

name m dname$_1$ dname$_2$ ... dname$_m$

where *name* is one of the family members' names, *m* is the number of his/her children, and *dname$_1$* through *dname$_m$* are the names of the children. These lines will be given in no particular order. You may assume that all n lines describe one single, connected tree. There will be no more than 1000 people in any one tree, and all names will be at most 10 characters long.

Output

For each test case, output the three names with the largest number of specified descendants in order of number of descendants. If there are ties, output the names within the tie in alphabetical order. Print fewer than three names if there are fewer than three people who match the problem criteria (you should not print anyone's name who has 0 of the specified descendants), and print more than three if there is a tie near the bottom of the list. Print each name one per line, followed by a single space and then the number of specified descendants. The output for each test case should start with the line

Tree *i*:

where *i* is the test case number (starting at 1). Separate the output for each problem with a blank line.

Example

Input:
3
8 2
Barney 2 Fred Ginger
Ingrid 1 Nolan

Cindy 1 Hal
Jeff 2 Oliva Peter
Don 2 Ingrid Jeff
Fred 1 Kathy
Andrea 4 Barney Cindy Don Eloise
Hal 2 Lionel Mary
6 1
Phillip 5 Jim Phil Jane Joe Paul
Jim 1 Jimmy
Phil 1 Philly
Jane 1 Janey
Joe 1 Joey
Paul 1 Pauly
6 2
Phillip 5 Jim Phil Jane Joe Paul
Jim 1 Jimmy
Phil 1 Philly
Jane 1 Janey
Joe 1 Joey
Paul 1 Pauly

Output:
Tree 1:
Andrea 5
Don 3
Cindy 2
Tree 2:
Phillip 5
Jane 1
Jim 1
Joe 1
Paul 1
Phil 1
Tree 3:
Phillip 5