Basic Grapes Instinct

Dickie is croping grapes for Christmas! He's very excited about the traditional grape eating race on December 31th, so he picked every possible raceme of grape in his farm and collect them in a big bucket. Nevertheless, he's having a serious problems ordering the racemes because there are so many of them!

In order to organize in a efficient way the humongous amount of grape racemes, he decided to build a stemplot out of them. A stemplot goes as follows:

• The input values for a stemplot are a sequence of numbers and a stem unit.

• Data is classified based on the stem value, i.e., if a number in the sequence is 403 and the stem unit is 100, then this number is classified in the category 4.

• For each category, a list of leaf values are render in a single line, sorted in increasing order.

• Stem values and leafs are separated by a | (pipe character). All | should be vertically aligned.

• Stem values are sorted in increasing order, omitting empty stems.

Given the sequence s = { 44 46 47 49 63 64 66 68 68 72 72 75 76 81 84 88 106 }, the stemplot generated with a stem unit of 10 is rendered as follow:

```
 4 | 4 6 7 9

 6 | 3 4 6 8 8

 7 | 2 2 5 6

 8 | 1 4 8

10 | 6
```

Dickie is having big trouble rendering these stemplots of grape racemes, can you do the job for him and save the traditional grape race of December 31th?

Input

The first line contains an integer T , which specifies the number of test cases. Then, will follow the descriptions of T test cases.

The first line of each case will contain 2 integers N and U , indicating the size of the sequence and the stem unit. Next N lines will contain a single element of the sequence si.

The input must be read from standard input.

Output

For each case you should print the string "Scenario #i:" where i is the test case you are

evaluating (starting from 1), a blank line, and then the corresponding stemplot following the

rules stated above. Print a blank line after each stemplot.

The output must be written to standard output.

Input	Output for the sample input
2	Scenario #1:
17 10	4 | 4 6 7 9
44	6 | 3 4 6 8 8
46	7 | 2 2 5 6
47	8 | 1 4 8
49	10 | 6
63	

64	Scenario #2:
66	0 | 1 9
68	1 | 0
68	3 | 4
72	5 | 12
72	
75	
76	
81	
84	
88	
106	
5 23	
1	
9	
127	
23	
73	

Subtask 1 - 30%

• $1 \leq T \leq 100$

• $1 \leq N \leq 100$

• $1 \leq U \leq 100$

• $1 \leq s_i \leq 1{,}000$

Subtask 2 - 70%

• $1 \leq T \leq 100$

• $1 \leq N \leq 10{,}000$

• $1 \leq U \leq 1{,}000$

• $1 \leq s_i \leq 1{,}000$