Problem 2: Make Widget - A CLASS-BASED PROBLEM

Overview

Modern manufacturing requires “just in time” delivery of parts. In this problem, you will simulate the manufacturing of widgets, useful devices that everyone needs.

Problem

You will be told how many of each part is needed to make a widget. You will also be told how many widgets you are to manufacture. You start with no parts on hand and must manufacture the appropriate number of widgets. At the end of the process, you must report how many of each part is “on hand”, i.e. have been received from the supplier, but not (yet) used to make a widget. Suppliers should only be contacted when parts are actually needed.

Input

The input to this problem is a series of lines of text. The first line contains two integers; the first (N) is the number of distinct parts used to create a widget. The second (G) is the number of widgets you are to manufacture. Both numbers will be positive integers. Widgets are never made from more than 64 distinct parts and you will never be asked to manufacture more than 1,000,000 widgets at a time.

The remaining N lines of the input consist of a positive integer and a name of a part. The integer will always be less than 207; the name will be a single word consisting only of lower case letters (a-z). Since the parts are distinct, no part name will be repeated.

Output

The output is a series of lines indicating the quantity of each part left over after the manufacturing process is complete along with the name of the part. It should be formatted as in the examples; in particular, the only space on the line is a single space between the integer and the name of the part. The parts should be listed in the same order as they were given in the input.

- over -
About the class

You must use the Supplier class in your solution to this problem. Otherwise, you will not be able to determine anything about the number of parts that you have on hand. All of the necessary methods are provided for you (although you may not need to use everything that is provided). Be sure to pay close attention to documentation of the class, including the pre- and post-conditions for all of the methods.

Example 1

Input

5 50
2 foo
3 bar
1 thingamajig
1 doohickey
1 whazzit

Output

9 foo
1 bar
14 thingamajig
11 doohickey
9 whazzit

Example 2

Input

1 100
2 gizmo

Output

17 gizmo