Problem H
Closest Common Ancestors

Input File: H.DAT
Program Source File: H.PAS or H.C or H.CPP

Write a program that takes as input a rooted tree and a list of pairs of vertices. For each pair **(u,v)** the program determines the closest common ancestor of **u** and **v** in the tree. The closest common ancestor of two nodes **u** and **v** is the node **w** that is an ancestor of both **u** and **v** and has the greatest depth in the tree. A node can be its own ancestor (for example in Figure 1 the ancestors of node 2 are 2 and 5)

The data set, which is read from a text file, starts with the tree description, in the form:

```
nr_of_vertices
vertex:(nr_of_successors) successor₁ successor₂ ... successorₙ
…
```

where vertices are represented as integers from **1** to **n**. The tree description is followed by a list of pairs of vertices, in the form:
```
nr_of_pairs
(u v)  (x y)  …
```

The input file contents several data sets (at least one).
Note that white-spaces (tabs, spaces and line breaks) can be used freely in the input.

For each common ancestor the program prints the ancestor and the number of pair for which it is an ancestor. The results are printed on the standard output on separate lines, in to the ascending order of the vertices, in the format: **ancestor:times**
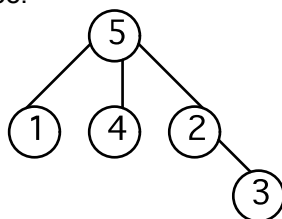
For example, for the following tree:

Figure 1

the program input and output is:

Input	Output
5 5:(3) 1 4 2 1:(0) 4:(0) 2:(1) 3 3:(0) 6 (1 5) (1 4) (4 2) (2 3) (1 3) (4 3)	2:1 5:5