

Problem G — Audubon's Stormy Arctic Trip

The program autopun assists in the creation of puns (mostly bad ones). Rather than listing every possible pun on a particular phrase it produces a rooted directed acyclic graph (DAG) as output. That is, a list of choices is given for the first word in a phrase. This choice determines not only a word but the name of another list that determines the choices for the next word, and so on. Each list is a node in the DAG, labelled with a two digit number (00 – 99).

We want to find the *n*th pun specified by autopun. The first is simply the phrase formed by selecting the first alternative for each choice. The next is formed by selecting the next alternative for the latest choice for which an alternative remains, followed by the first alternative for all subsequent choices. That is, the order of the puns is determined by a left-to-right traversal of the leaves in the DAG.

The output of autopun consists of several node entries. The first line of an entry contains a two digit number (from 00 to 99) followed by a colon. Subsequent lines in the entry are indented and consist of 'word:node' entries separated by spaces. A '\$' for a node indicates the end of the pun and 00 is the first node. The input data is finished by a blank line or the end-of-file.

After the node entries there will be zero or more lines containing integers in the range 1 to the highest pun number for the DAG.

Output will consist of one pun per line with the words separated by hypens. A blank line must follow the output for each instance.

Sample Input:

```
00:
  silver:$  zeal:03 sill:03 shill:03 seal:03 ceil:03 Shea:02
  she:02 see:02 sea:02 Sci:02 z:01 s:01 h's:01
01:
  ill:03 I'll:03 ewe:03 eel:03 e:02
02:
  w:03 l:03
03:
  fur:$  fir:$  v:04 f:04
04:
  IR:$
1
2
3
4
5
6
27
```

Sample Output:

```
silver
zeal-fur
zeal-fir
zeal-v-IR
zeal-f-IR
sill-fur
Shea-l-fir
```