

## 3851 Electronic Document Security

The Tyrell corporation uses a state-of-the-art electronic document system that controls all aspects of document creation, viewing, editing, and distribution. Document security is handled via access control lists (ACLs). An ACL defines a set of entities that have access to the document, and for each entity defines the set of rights that it has. Entities are denoted by uppercase letters; an entity might be a single individual or an entire division. Rights are denoted by lowercase letters; examples of rights are ‘a’ for *append*, ‘d’ for *delete*, ‘e’ for *edit*, and ‘r’ for *read*.

The ACL for a document is stored along with that document, but there is also a separate ACL log stored on a separate log server. All documents start with an empty ACL, which grants no rights to anyone. Every time the ACL for a document is changed, a new entry is written to the log.

An entry is of the form ‘ $ExR$ ’, where  $E$  is a nonempty set of entities,  $R$  is a nonempty set of rights, and  $x$  is either ‘+’, ‘-’, or ‘=’. Entry “ $E+R$ ” says to grant all the rights in  $R$  to all the entities in  $E$ , entry “ $E-R$ ” says to remove all the rights in  $R$  from all the entities in  $E$ , and entry “ $E=R$ ” says that all the entities in  $E$  have exactly the rights in  $R$  and no others. An entry might be redundant in the sense that it grants an entity a right it already has and/or denies an entity a right that it doesn’t have. A log is simply a list of entries separated by commas, ordered chronologically from oldest to most recent. Entries are cumulative, with newer entries taking precedence over older entries if there is a conflict.

Periodically the Tyrell corporation will run a security check by using the logs to compute the current ACL for each document and then comparing it with the ACL actually stored with the document. A mismatch indicates a security breach. Your job is to write a program that, given an ACL log, computes the current ACL.

### Input

The input consists of one or more ACL logs, each 3-79 characters long and on a line by itself, followed by a line containing only ‘#’ that signals the end of the input. Logs will be in the format defined above and will not contain any whitespace.

### Output

For each log, output a single line containing the log number (logs are numbered sequentially starting with one), then a colon, then the current ACL in the format shown below.

#### Note that:

- (1) spaces do not appear in the output;
- (2) entities are listed in alphabetical order;
- (3) the rights for an entity are listed in alphabetical order;
- (4) entities with no current rights are not listed (even if they appeared in a log entry), so it’s possible that an ACL will be empty; and
- (5) if two or more consecutive entities have exactly the same rights, those rights are only output once, after the list of entities.

**Sample Input**

```
MC-p,SC+c
YB=rde,B-dq,AYM+e
GQ+tju,GH-ju,AQ-z,Q=t,QG-t
JBL=fwa,H+wf,LD-fz,BJ-a,P=aw
#
```

**Sample Output**

```
1:CSc
2:AeBerMeYder
3:
4:BHJfwLPaw
```