

## FREEDONIA REGULATIONS

Write a Python program to keep track of the regulations in Freedonia. The program reads the sequence of rules issued by Rufus T. Firefly and calculates the set of rules that are valid in specific days.

In more details, the file `rules.dat` contains the rules in the format:

```
date1: +/-rule1 ... +/-ruleI
date2: +/-rule1 ... +/-ruleJ
...
dateN: +/-rule1 ... +/-ruleK
```

Where the **date** is a string in the format `dd-mm-yyyy`; a **rule** is a string without spaces, for instance `FFP2_mask_in_closed_spaces`, the leading plus `+` or minus `-` signifies that the rule is enforced (`+`) or lifted (`-`). One or more rules might be specified in each date. Dates are in strict chronological order, but they might be non consecutive. The format of the file is correct.

A second file, `dates.dat` contains a list of interesting dates in the same format, these dates might not be in chronological order.

The program must report, for each of such dates, the set of active rules, one per line, in alphabetic order.

## Example

### `rules.dat`

```
01-01-2021: +surgical_mask_outdoor +surgical_mask_indoor
05-01-2021: -surgical_mask_indoor +ffp2_mask_indoor
08-01-2021: +green_pass_workplace
01-02-2021: -ffp2_mask_indoor +surgical_mask_indoor -surgical_mask_outdoor
01-03-2021: +surgical_mask_outdoor
05-03-2021: +super_green_pass_workplace -green_pass_workplace
```

### `dates.dat`

```
25-01-2021
01-01-1999
01-02-2022
```

### output

```
25-01-2021:
ffp2_mask_indoor
green_pass_workplace
surgical_mask_outdoor
```

01-01-1999:

01-02-2022:

super\_green\_pass\_workplace

surgical\_mask\_indoor

surgical\_mask\_outdoor