2007 South Central USA Regional Programming Contest

The Final Countdown

Introduction:

When Doctor Doom modeled LAI-BACH (the Latverian Aerospace Institute, Bolograd Active Control Headquarters) on Mission Command with NASA, he perhaps cribbed from the source a bit *too* closely.

NASA (and therefore LAI-BACH) have a peculiar way of handling countdowns to missions. While the clock may state that it is, say, thirty minutes until liftoff for a rocket or shuttle, there are built-in *holds* where the clock is stopped for a set amount of time. For example, there may be a 15-minute hold when the countdown reaches the 8-minute mark. The end result is that more time passes from the beginning of the countdown to the end than the numbers on the clock would imply.

LAI-BACH uses a simple "command file" format to manage holds for their countdowns on their computer systems. All commands start with a positive integer no greater than 1440 (countdowns longer than a day make Doctor Doom angry), followed by a directive. The format of the commands is as follows:

t START	*t* is the start time on the countdown clock, in minutes.
t HOLD *n*	A mandatory hold for *n* minutes at countdown clock time *t*.
t HOLD *n* IF *condition*	A hold at countdown clock time *t* for *n* minutes if *condition* is true.
t HOLD *n* IF NOT *condition*	A hold at countdown clock time *t* for *n* minutes if *condition* is **not** true.

All conditions are represented by short strings of lowercase letters, no more than 20 characters long; their actual values are not provided in the command structure, as they are determined by actual conditions during the launch (whether the fuel tanks have been topped off, whether the Fantastic Four is currently fighting Doctor Doom, and so on), but they are always either true or false and do not change during the course of a single countdown. A hold time *n* is a positive integer no more than 60 (minutes), as long delays also irritate Victor.

Given a particular command file, you are to determine both the shortest possible time and the longest possible time that the countdown can run. The commands may be present in any order, but there will always be one and only one `START` directive; there will be precisely as much whitespace on each line as dictated by the format given above. No two commands in a command file will reference the same minute *t*.

Input:

Input to this problem will begin with a line containing a single integer N ($1 \leq N \leq 100$) indicating the number of data sets. Each data set consists of the following components:

- A line containing a single integer L ($1 \leq L \leq 100$) indicating the number of lines in the particular "command file"; and
- A series of L lines representing the command file, in the format described above.

Output:

For each data set, print "*min* `TO` *max*", where *min* is the shortest possible time the countdown can last in minutes, and *max* is the longest possible time the countdown can last in minutes.

Sample Input:

```
2
3
30 START
15 HOLD 5
10 HOLD 5 IF fantasticfour
3
75 HOLD 20 IF yellowlightblinks
150 HOLD 30 IF NOT fueled
300 START
```

Sample Output:

```
35 TO 40
300 TO 350
```

The statements and opinions included in these pages are those of the Hosts of the South Central USA Regional Programming Contest only. Any statements and opinions included in these pages are not those of Louisiana State University or the LSU Board of Supervisors.