Moving Robots

There are several robots moving on a 2-d whole grid plane. The state of every robot is defined by the current position and direction and every robot moves according to its own finite sequence of commands. Position is described by a pair of integers (x, y). There are four possible robot's directions that are described by the number of degrees: 0, 90, 180 or 270. The commands are of two kinds – turn and move. Turn command has one parameter D which can be one of 90, 180 or 270. The command changes the robot's current direction in D degrees: direction C is changed to direction (C + D) mod 360. Step command has no parameters and makes the robot move one step in its current direction. One step in direction 0 changes robot's current position by (1, 0), in direction 90 – by (0, 1), in direction 180 – by (-1, 0) and in direction 270 – by (0, -1).

A robot performs the commands of its sequence one after another. When the whole sequence is performed the robot stops in the final position.

The movement of a robot is not affected by the other robots in any way. In particular, any number of robots can share the same position.

Before the robots start to move the control center can order some robots to remove some commands from their sequences. So the control center can influence the paths of the robots as well as their final positions. The control center wants all robots to gather in the common final position for inspection. It also wants to achieve this with the minimum possible total number of removed commands.

Task

There are R ($2 \le R \le 10$) robots. Each robot has its own initial position, initial direction and its own sequence of commands that contains no more than 50 commands. Write a program that determines (if possible):

- the minimum total number of commands to be removed from some sequences that all robots would stop in the common final position;
- the common final position in that case.

If there are several such positions then the program should output any one of them.

Input data

Input data should be read from the file `robots.in`. The first line of the file contains integer R ($2 \le R \le 10$) – the number of robots. Then R sections of lines follow. Each section describes one robot. The first line of the section contains four integers, separated by single spaces: x, y – initial robot position (x, y), initial robot direction C (C = 0, 90, 180, or 270) and the length of robot's command sequence n ($1 \le n \le 50$). Then the section contains n lines describing the sequence of commands, one command per line. The line of the step command contains single character `S` in the first position and the line of the turn command contains character `T` in the first position that is followed by turn parameter – integer D (D = 90, 180 or 270) and is separated by a single space.

Output data

Output data should be written to the file `robots.out` . If it is not possible to make the robots stop in the common final position by removing any set of commands then the program should write integer –1 (minus one) to the first (and single) line of the file. Otherwise, the minimum total number of commands to be removed should be written to the first line. The second line should contain 2 integers separated by single space: the first and the second coordinates of the common final position.

Example

`robots.in`

```
2
2 0 270 5
S
T 180
S
S
S
1 -1 0 8
S
S
T 90
S
T 270
S
T 90
S
```

`robots.out`

```
2
2 1
```

Comment

There are 2 moving robots. The first robot has initial position (2, 0), direction 270 and a sequence of 5 commands. The second one has initial position (1, -1), direction 0 and a sequence of 8 commands. The minimum total number of commands to be removed that makes robots share final position is 2: for example, remove the 3-rd command of the first robot and the 5-th command of the second robot. The common final position in that case is (2, 1).