PROBLEM G: LAND ADMINISTRATION IN PRL

Input file: standard input

Output file: standard output

Problem

PRL (Poly-Rectangular Land) is an unusual country. The country is flat, and its frontier consists of straight line sections going South-North or West-East. Over long history of the country agreements were attained with all neighbouring countries to keep frontiers stable and peaceful. Moreover, to avoid ambiguous responsibilities PRL controls only its western and southern frontier segments; other frontier sections are controlled by the respective neighbour countries. A western (southern) frontier segment is such a piece of frontier that crossing it to the West (South) we get to the neighbouring country.

PRL authorities established the Landowners' Law according to which any real estate should be poly-rectangular and singly connected (no "holes" inside). As for state frontiers, western and southern borders of a land property belong to it; northern and eastern borders belong to the neighbouring real estate. Each landowner's property consists of a set of rectangular plots of ground which can be freely traded among neighbouring landowners under the condition that after selling/buying a plot the resulting shape of changed real estates will conform to the Landowners' Law. The subdivision of a land property into rectangular plots of ground is individually decided by its landowner.

The Land Administration Agency (LAA) receives registration requests from landowners whenever they buy/sell some ground plots or decide on new subdivision of their land property or both. A request consists of the landowner's identification and a sequence of ground plot definitions constituting the property. The Agency processes the registration request and delivers so called "border decision" or rejects the request if plot definitions do not conform to the Landowners' Law. You are to write a program for processing registration requests.

Input

The input to the program is a text stream. First line of the input is a positive integer n representing number of requests in the stream. One request consists of a sequence of text lines. First line of a request contains landowner's identification, i.e. a name up to 10 letters long; subsequent lines contain definitions of ground plots, one definition per line, i.e. four integers separated by a blank: x, y-coordinates for South-West corner and x, y-coordinates for North-East corner of the ground plot; last line of the request contains 4 zeros (X-axis is oriented East, Y-axis is oriented North). You can assume the input is well-formed.

Output

For each request the program generates

- first line: REQUEST#*i* FROM <*landowner id*> where *i = 1, ... , n*
- second line iff error detected: ERROR - GROUND PLOTS OVERLAP or
  ERROR - ESTATE NOT SINGLY CONNECTED
- iff request ok: sequence of lines with border definition.
  A border definition consists of:
  1. One line with the x,y-coordinates of the most South-West point of the land property

2. One or more lines with instructions for a walker to walk around the estate starting from the point given above; initially the walker is heading North and walks always having the estate terrain to his right. After coming back to the starting point the walker should be again heading North.

Possible instructions are:

- F*m* go forward *m>0* units along the border belonging to the estate (western or southern border)
- f*m* go forward *m>0* units along the border not belonging to the estate (northern or eastern border)
- T+ or T- turn 90 degrees to the right or left.

A line can contain no more then 10 instructions.

Example

Input file

```
2
JONES
2 0 7 2
1 5 8 9
0 0 2 3
2 2 8 5
0 0 0 0
MORRIS
-5 0 -3 5
1 0 2 3
-3 3 3 5
-3 0 1 2
0 0 0 0
```

Output file

```
REQUEST#1 FROM JONES
0 0
F3T+f2T-F2T-F1T+F4T+
f7T+f7T+F1T-f2T+F7T+
REQUEST#2 FROM MORRIS
ERROR - ESTATE NOT SINGLY CONNECTED
```