Airport Show

Source code: airport.c/airport.cpp/airport.pas
Time limit: 5 s
Memory limit: 64 MB

The management of the airline company Flying Bugs decided to prepare an aerial show as a part of their advertising campaign. The show will take a place at the company airport which has N runways (numbered $1 \dots N$) and will consist of performances of two airplane acrobatic groups held in parallel. Each performance has a predefined program and to keep its visual effect the planes have to take off and land on specific runways in a prescribed order.

The security system of the airport tracks which runways are in use and it will not let any other plane enter the runway during that time. Each performance is delivered to the security team as a sequence of reservations and releases of specific runways. Unfortunately, it cannot be estimated in advance when the requests for reserving and releasing runways will actually appear. The airport authorities would now like to know whether there is a danger that the performances could be delivered in a such a way that at some point neither of the two performances can continue without violating the order of the uses of the runways.

Task

Your task is to write a program that checks whether the requests of the two performances can appear in such an order that neither of the performances can continue. For each of the two performances, you are given a list of requests on reservations and releases of runways. The order of the reservations and releases for each of the two performances satisfies the following conditions:

- a reserved runway is not reserved until it has been released before,
- only a runway previously reserved and not yet released is released, and
- all reserved runways have been released before the end of the performance.

On the other hand, if a certain runway is needed for one of the performances and it is currently reserved for the other one, the former performance can be suspended until the latter one releases the required runway, i.e., the latter performance can reserve and release several runways during its course and eventually release the requested runway by the former performance.

Still, the performances can "block" the airport, if one of them is requiring reservation of runway A while keeping the runway B reserved and the other one is requiring reservation of runway B while keeping runway A reserved. If this happens, neither of the two performances can continue without violating its order of the take-offs and landings.

Note that your task is not to check whether the performances can be scheduled in such a way that the airport does not get blocked.

Description of input

The first line of the input contains a single integer N, $1 \le N \le 1\,000$, that determines the number of runways at the airport. The rest of the input is comprised of two blocks, each describing one of the performances. The first line of the block contains a single even integer L, $2 \le L \le 5\,000$, that determines the number of reservations and releases of runways done during the course of the performance. Each of the following L lines contains a 3-character string `RES` or `REL` and an integer A, $1 \le A \le N$, which are separated by a single space. The lines starting with the string `RES` represent demands on reservations of the runways and those starting with `REL` on releases of the runways. The integer A then determines the number of the runway to be reserved or released.

Description of output

If the performances always finish without blocking the airport (whatever the time differences between reserving and releasing runways are), the output should contain a single line with the sentence "`The performances will always finish.`".

If there exists a sequence of reservations and releases leading to blocking the airport, the output should contain a description of a possible course of the show leading to such a situation (note that there could be more such courses and you should output an arbitrary one of them). The description of the course of the show should be a sequence $a_1 \ldots a_k$ of the numbers 1 and 2 (with no separating spaces between the numbers). These numbers represent the order in which the demands of the two performances happen (the numbers 1 and 2 represent the first and the second performance): at the beginning, the first of the demands of the a_1-th performance is done, then the first demand of the a_2-th performance if $a_2 \ne a_1$ or the second demand of the a_2-th performance if $a_1 = a_2$, etc. The sequence must be a valid sequence of operations, i.e., at each time, every runway can be reserved by at most one of the two performances. Since the sequence describes the course of the show leading to blocking the airport, the next demand of the first performance which is not covered by the sequence should request reserving a runway reserved for the second performance and the next demand of the second performance should request reserving a runway reserved by the first one.

Example 1

Input:
```
2
4
RES 1
RES 2
REL 1
REL 2
4
RES 2
RES 1
```

```
REL 2
REL 1
```

There are two possible solutions for the above input.

Output 1:
```
12
```

Output 2:
```
21
```


Example 2

Input:
```
4
8
RES 1
RES 2
RES 3
REL 3
REL 2
RES 2
REL 1
REL 2
4
RES 3
REL 3
RES 4
REL 4
```

Output:
```
The performances will always finish.
```