There is a bag-like data structure, supporting two operations:

1 x	Throw an element x into the bag.
2	Take out an element from the bag.

Given a sequence of operations with return values, you're going to guess the data structure. It is a stack (Last-In, First-Out), a queue (First-In, First-Out), a priority-queue (Always take out larger elements first) or something else that you can hardly imagine!

Input

There are several test cases. Each test case begins with a line containing a single integer n ($1 \leq n \leq 1000$). Each of the next n lines is either a type-1 command, or an integer 2 followed by an integer x. That means after executing a type-2 command, we get an element x **without error**. The value of x is always a positive integer not larger than 100. The input is terminated by end-of-file (EOF).

Output

For each test case, output one of the following:

`stack`	It's definitely a stack.
`queue`	It's definitely a queue.
`priority queue`	It's definitely a priority queue.
`impossible`	It can't be a stack, a queue or a priority queue.
`not sure`	It can be more than one of the three data structures mentioned above.

Sample Input

```
6
1 1
1 2
1 3
2 1
2 2
2 3
6
1 1
1 2
1 3
2 3
2 2
2 1
2
1 1
2 2
4
1 2
1 1
2 1
2 2
7
1 2
1 5
1 1
1 3
2 5
1 4
2 4
```

Sample Output

```
queue
not sure
impossible
stack
priority queue
```