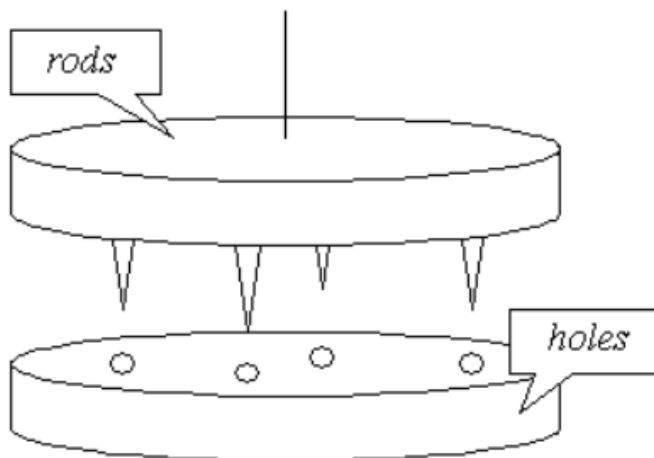Christmas is coming. In the center of the city park, there is a huge toy, shown in the picture.



There are $n$ rods in the upper part and $n$ holes in the lower parts. Both rods and holes are equally spaced. The lengths of the rods are exactly $1 \sim n$ (but not necessarily in this order), and depths of the holes are also exactly $1 \sim n$.

Initially, the two parts are touching each other. That means, each rod is in a hole with the same depth as the rod's length. Then, before the game starts, the upper part is lifted and rotated quickly for several seconds.

Your task is the re-connect the two parts like its initial state (i.e. each rod is in a hole with the same depth as the rod's length). Note that you can see the lengths of the rods, but you can't see the depths of the holes.

## Interaction Protocol

*Your program should read from standard input, and write to standard output. After printing each line to the standard output, you should flush the output, by calling* `fflush(stdout)` *or* `cout << flush` *in C/C++,* `flush(output)` *in Pascal and* `System.out.flush()` *in Java. Please read general instructions for interactive problems for more information.*

First, read the number of test cases $T$ $(1 \le T \le 25)$. For each test case, first read an integers $n$ $(1 \le n \le 100,000)$ in the first line. The next line contains $n$ integers, the lengths of the rods.

Then issue one or more 'Rotate' or 'Drop' command.

| Command | Description |
|---------|-------------|
| Rotate $k$ | Rotate the upper part $k$ unit $(1 \le k < n)$ counter-clockwise. This command does not return anything. |
| Drop | Try to connect the two parts. Returns the distance of the two parts, i.e. maximum $l_i - d_i$, where $l_i$ is the $i$-th rod's length (after rotation), $d_i$ is the depth of the $i$-th hole. When this command returns 0, the current test case ends. |

**If your program violated any of these rules (bad format, invalid arguments etc), the server will exit immediately, and you will receive Protocol Violation (PV).**

## Protocol Limit

For each test case, you can issue at most 5 'Drop' commands (including the last one that successfully connects the two parts), **otherwise you'll get Protocol Limit Exceeded (PLE).**

**Sample Explanation:** The holes' lengths are 1, 3, 4, 2, 5. After 'Rotate 4', the rods' lengths are 4, 2, 5, 1, 3.

## Sample Interaction

```
1
5
2 5 1 3 4
                        Rotate 4
                        Drop
3
                        Rotate 3
                        Drop
0
```