Glitch

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

This is an interactive problem!

There's a hidden array $a_1, a_2, \ldots, a_N$ of N distinct positive integers (where N is odd); an incorrect copy of this array is stored in another hidden array b. Each element of array b is defined as follows:

- $b_1 = a_2$
- $b_N = a_{N-1}$
- $b_i = a_{i-1}$ or $b_i = a_{i+1}$ for each i where $1 < i < N$ (one of the adjacent elements)

You need to determine whether there are two indices i and j ($i \neq j$, $1 \leq i, j \leq N$) such that $b_i = b_j$. If such indices exist, return them; otherwise, indicate that no such indices exist.

You can query the elements of arrays a and b using the following operations:

- `1 i`: returns the value of a_i ($1 \leq a_i \leq 10^9$)
- `2 i`: returns the value of b_i ($1 \leq b_i \leq 10^9$)

You can ask up to a maximum of 20 queries.

Interaction Protocol

First, you must read a line containing a single integer T ($1 \leq T \leq 100$) denoting the number of test cases. Then follows a description of the interaction for T test cases.

For each test case, you must first read a line containing one integer N ($3 \leq N \leq 2\,001,\ N$ is odd).

Then interaction starts. You can print one line, containing:

- `1 x` (Ask about a_x): Then, read one line, consisting of one integer, a_x.
- `2 x` (Ask about b_x): Then, read one line, consisting of one integer, b_x.
- `3 i j` (The answer: two indices i, j, where $b_i = b_j$ or `-1 -1` if there are no such two indices): Then, proceed to the next test if there is any; otherwise, terminate.

After outputting each line, don't forget to flush the output. For example:

- `fflush(stdout)` in C/C++;
- `System.out.flush()` in Java;
- `sys.stdout.flush()` in Python;
- `flush(output)` in Pascal;
- See the documentation for other languages.

Example

standard input	standard output
1 7 2 2 	 2 3 2 1 3 1 3