Problem Mìmì

Input file stdin
Output file stdout

Ālín and Lìwéiwū are playing a game. The game works as follows: Ālín has a 0-indexed N by N matrix which initially has the element $i \cdot n + j$ on cell (i, j). Then Ālín shuffles the matrix by swapping two lines or two columns any number of times and by rotating the matrix 90° any number of times. Now Lìwéiwū has to find the number 0 for 100 times in a row and do it as fast as possible. Every time Lìwéiwū find the number 0, Ālín reshuffles the matrix and Lìwéiwū has to find it again, until it does it for 100 times.

To be able to do it, Ālín is forced to answer some questions for Lìwéiwū. Those questions can be of 2 types as follow:

1. Lìwéiwū asks what element ended at position (i, j) and Ālín is forced to answer him truthfully.
2. Lìwéiwū asks where is the value x located but he can do it only one time, and Ālín will answer his question only if $x \neq 0$

Ālín may use the information from the previous games when he shuffles the matrix for the current round.

Implementation

You should submit a file that implements the function `solve`, which will return the answer as a pair of values `(row, column)`. `solve` may call the grader's functions `query_cell` and `query_value` up to 205 times in total. These functions and procedures are described below.

Signature	Details
`int query_cell(int i, int j)`	Returns the value stored in the cell at row `i` and column `j`
`pair<int, int> query_value(int value)`	Returns a pair `(row, column)` representing the cell where `value` is stored. **Can only be called ONCE**.
`pair<int, int> solve(int type, int N)`	The function you have to implement. The parameters are: • `type`, representing the subtask you are currently solving. The value `0` signals you are solving the first subtask, while the value `1` signals you are solving the second one. • `N`, the dimensions of the square matrix which is of `N` by `N`. The return value is, as mentioned before, a pair `(row, column)` of the element `0`.

You should include the header ”mimi.h” in your program.

Restrictions

- $N = 100$
- Number of times the game is played = 100

#	Points	Restrictions
1	50	The matrix is shuffled random
2	50	The matrix is not shuffled random

Scoring

Each subtask is scored as follows. Let's say you have done a total of x queries over all 100 games.

x	Points
$x \leq 4000$	50
$4000 \leq x \leq 5500$	$20 + 30 \cdot \left(1 - \left(\frac{x - 4000}{1500}\right)^2\right)$
$5500 \leq x \leq 10500$	$10 + 10 \cdot \left(1 - \left(\frac{x - 5500}{5000}\right)^2\right)$
$10500 \leq x \leq 20500$	$5 \cdot \left(1 - \left(\frac{x - 10500}{10000}\right)^2\right)$

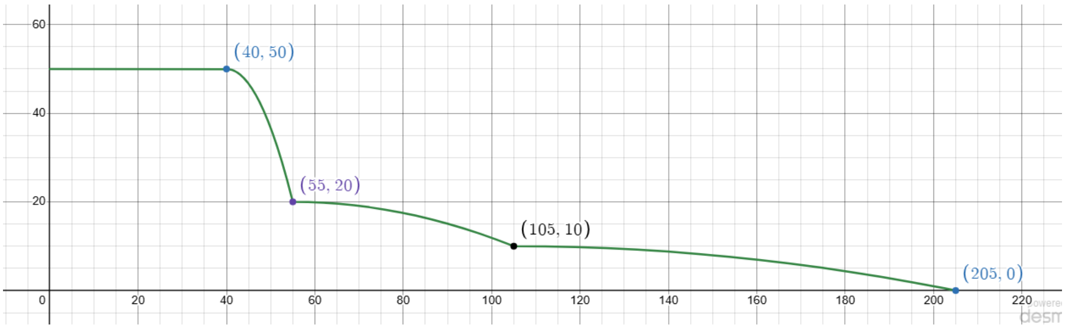

Figure 1: The points you get for the average number of queries

Grader	Solver	Explanations

10
 	
1 3
	query_cell(2, 3)
	
	query_value(7)
	
	$(0,0)$ - returned by solve
		For some reasons Ālín didn't shuffle the matrix at all, but Lìwéiwū doesn't know this so he asks what element is on position 2 3 and he get the answer 10. Now Lìwéiwū want to know the position of 7 and he get the answer $(1,3)$. Since Lìwéiwū is smart enough he knows that 0 is on position $(0,0)$ so he returns the pair $(0,0)$