H Historical Hierarchy

Time limit: 2s

While taking an online history quiz, you encounter a question that has never been your strong suit: ordering a list of n historic events from different years in chronological order. Upon seeing that you are allowed to make $4n^2$ attempts at submitting an answer, you decide to open the browser console and write a script to automatically submit answers for you.



Do you know the correct ordering of: the first Dutch Programming Championship, the first ICPC Programming Contest ever held, and the first International Olympiad in Informatics?

From the browser console, you find you can only perform one kind of operation. By providing two indices i and j ($1 \leq i, j \leq n$), the following two things will happen:

- The ith element is removed from the list. This will shift all elements to the right of position i one position to the left.
- The deleted element is inserted back at position j. This will shift all elements at position j or to the right of it one position to the right.

After such an operation, the current order is submitted and your answer receives a score, equal to the current length of a Longest Increasing Subsequence.[1]

Interaction

This is an interactive problem. Your submission will be run against an *interactor*, which reads from the standard output of your submission and writes to the standard input of your submission. This interaction needs to follow a specific protocol:

The interactor first sends one line with an integer n ($2 \leq n \leq 100$), the number of historic events.

Then, your program should send at most $4n^2$ commands, to reorder the events in chronological order. Each command is sent by printing one line of the form "i j" ($1 \leq i, j \leq n$), which takes out the element at position i and reinserts it at position j.

The interactor will respond with one line with an integer k ($1 \leq k \leq n$), the current length of a Longest Increasing Subsequence.

When the interactor responds with $k = n$, all events are in order and the interaction will stop. Sending any additional commands after this point or using more than $4n^2$ commands will result in a wrong answer.

The interactor is not adaptive.

Make sure you flush the buffer after each write.

A testing tool is provided to help you develop your solution.

[1] A subsequence is obtained from a sequence $a_1, a_2, \ldots, a_n$ by deleting zero or more elements, without changing the order of the remaining elements.

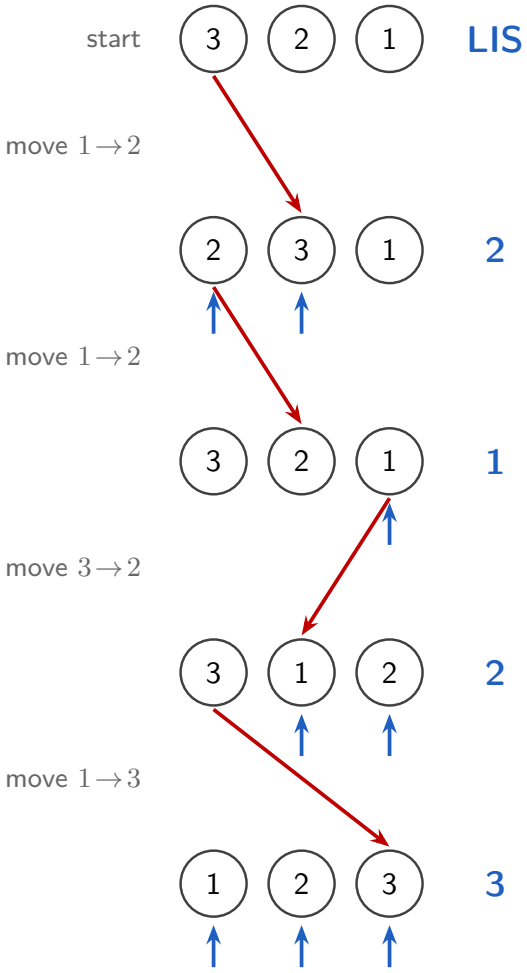


Figure H.1: Visualization of the interaction in the second example. At the start all the historic events are in reverse order. The red arrows show the operations, and the blue arrows show a possible Longest Increasing Subsequence (it may not be unique).

Read	Sample Interaction 1	Write
2		
	1 1	
1		
	1 2	
2		

Read	Sample Interaction 2	Write
3		
	1 2	
2		
	1 2	
1		
	3 2	
2		
	1 3	
3		