Problem H
Heist of the Century
Time Limit: 4 Seconds, Memory Limit: 2G

After planning the most elaborate heist to steal The Crown Jewel of Count Monte Carlo, you have encountered the final obstacle in your path: a locked safe. However, you have trained for this very moment and honed your safecracking abilities.

The safe has N dials, each of which can be set to any integer value from 1 to $2N$. Count Monte Carlo has configured the safe with a correct, secret value for each dial. To attempt to open the safe, you set each dial to a value of your choice, and then turn the safe door handle. If every single dial is set to its correct secret value, you will feel no resistance and the door will swing open immediately.

Of course, opening the safe by randomly guessing all of the right secret values is highly unlikely to succeed. However, as an expert safecracker, even if your guess is wrong you feel some resistance when you attempt to open the door and you can use that knowledge to help decipher the correct secret values. If a dial has secret value h and the dial is set to d when you try to open the door, the dial will exert resistance $|h - d|$ to turning the door handle. You can feel the *maximum* resistance over all of the dials. (Note that if this value is 0, you have successfully opened the safe and completed your heist!)

Unfortunately, the security team has been made aware of your presence and they are closing in on your location. You are able to make one attempt at opening the safe per second, but they are $4N$ seconds away! Can you complete the heist before you get caught?

Interaction

This is an interactive problem. Interaction begins by reading an integer N ($1 \leq N \leq 500$) from standard input, the number of dials on the safe. Your program may make up to $4N$ attempts to open the safe door by specifying a value to try for each safe dial. You will then be told the resistance you feel from the door handle after each attempt.

To make an attempt, print a line containing N space-separated integers $d_1, d_2, \ldots, d_N$ ($1 \leq d_i \leq 2N$), the values you propose for each dial. Then read a single integer from standard input representing the resistance that you felt from the door handle, $\max_i |h_i - d_i|$, where h_i is the (unknown to you) secret value of dial i. If the resistance is 0, you have opened the safe and your program should terminate. Otherwise, if you have attempts remaining, you may try again.

If you run out of attempts, your program should exit cleanly (though it will be judged incorrect for failing to crack the safe in time to escape the guards).

The secret value of each dial was configured by Count Monte Carlo in advance of the heist and will not change in response to your cracking attempts.

Notes

Do not forget to flush the output stream after each line that you print and to cleanly exit after the interaction is done. Please also make sure that you follow the above interaction protocol **exactly** regarding what information to print on which line of output: for example, if the protocol requires you to print a list of space-separated integers on a single line, the judge **will not** accept each integer on its own line.

If the judge receives invalid or unexpected input, it will print -1 and then immediately exit. Your program must detect this error report and cleanly exit in order to receive a Wrong Answer verdict. If your program blocks waiting for further interaction from the judge, or tries to interpret the -1 as a resistance value, you may receive a different rejected verdict (such as Time Limit Exceeded or Runtime Error) instead of Wrong Answer.

You have been provided with a command-line tool for local testing. The tool has comments at the top explaining its use.

Read	Sample Interaction 1	Write
3		
		1 1 1
5		
		3 4 5
1		
		4 5 6
0		