

Problem ID: counterfeitcoin

You are a market salesman. Every week you sell your wares in a beautiful authentic market stall. People travel from faraway lands to buy your signature product: fermented shark. Of course, such a delicious product should only be exchanged for a vast amount of shining, valuable coins.

Sometimes customers try to fool you by paying with counterfeit coins. You always spot them, of course; you take out your trusty scale and weigh the coins to determine if any of them are of different weight.

In this particular instance, a rather notorious customer by the name of Peter Rog Rammer, famous for his “off-by-one errors”, has just paid for a fermented shark dish using n seemingly identical coins. However, you are sure that there is one coin that is not quite the same as the others. As the line of customers is getting rather lengthy, you do not want to spend too much time finding the odd one out!



Counterfeit Shilling of Elizabeth I, CC BY-SA 3.0

Interaction

This is an interactive problem. Your submission will be run against an *interactor*, which reads 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 an integer n ($3 \leq n \leq 2 \cdot 10^5$).
- Your submission then repeatedly sends two integers x and y ($1 \leq x, y \leq n$, $x \neq y$) preceded by “?”, indicating that you want to compare the x th and y th coin.
- The interactor replies with a string:
 - “lighter” if the x th coin weighs less than the y th coin.
 - “heavier” if the x th coin weighs more than the y th coin.
 - “equal” if the x th coin weighs the same as the y th coin.
- When you are ready to print the answer, print “!” followed by a single integer x ($1 \leq x \leq n$), indicating that the x th coin is the coin that does not weigh the same as the others.

You may use at most $\lceil n/2 \rceil$ moves. Printing the answer does not count as a move.

Note that for testing purposes, the index of the counterfeit coin is not necessarily fixed at the start of the interaction, but may be set by the interactor at any point (but always in a way that is consistent with its answers to previous queries).

Make sure you flush the buffer after each write.

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

Read

Sample Interaction 1

Write

5

? 1 5

equal

? 4 3

lighter

? 2 4

heavier

! 4