

7. Ленивая сортировка (sort)

1 сек

100 очков

Учитель Лаур организовал для N своих учеников соревнование по информатике и теперь хочет наградить их призами, каждого в соответствии с его местом в соревновании. Призы уже находились в правильном порядке на полке, но Тоотс от помутнения толкнул полку, и все они посыпались на пол. Поскольку они находятся в одинаковых коробках, пришлось учителю поместить их назад на полку в случайном порядке.

Однако у призов немного отличается вес, а у учителя Лаура есть рычажные весы, которыми можно узнать, какая из двух коробок должна достаться ученику с результатом получше, а какая — с результатом похуже. Но каждое взвешивание занимает время, и первые M учеников уже выстроились за своими призами. . .

Для того, чтобы сократить время, которое ученики в сумме проведут в очереди, Лаур хочет выдать каждому ученику его приз, используя как можно меньшее число взвешиваний. Напиши программу, которая поможет ему это сделать.

Общение. Это интерактивное задание. После запуска программы, в первой строке ввода будет дано число призов N ($1 \leq N \leq 100$) и количество учеников в очереди M ($1 \leq M \leq N$). Призы пронумерованы $1 \dots N$, в согласии с их местом на полке.

Дальше тестировочная система подаёт программе M запросов. Программе нужно обработать запрос и ответить на него, и только тогда тестировочная система даёт следующий. После ответа на последний запрос программа должна завершить работу.

Каждый запрос — это число K ($1 \leq K \leq N$), которое означает место в соревновании следующего ученика, пришедшего за призом. (У каждого ученика уникальное место.)

Во время обработки запроса программа может попросить учителя сделать некоторые взвешивания. Для каждого взвешивания программа должна вывести строку вида ‘? $A B$ ’, где A и B — номера двух призов на полке ($1 \leq A \leq N$, $1 \leq B \leq N$, $A \neq B$). После этого тестировочная система направит на вход программе в отдельную строку результат взвешивания: знак ‘<’, если приз A предназначен для ученика с результатом получше, или знак ‘>’ в противном случае.

Когда программа определит, какой приз нужно выдать ученику за его K -тое место, программе нужно вывести строку вида ‘! C ’, где C — это номер приза ($1 \leq C \leq N$). Таким образом будет обработан один запрос. Если есть ещё запросы, тестировочная система подаст следующий запрос (в следующей строке). Когда программа ответит на все M запросов, ей нужно завершиться.

Оценивание. В этом задании программа, которая даст хотя-бы одному ученику не его приз, получит за соответствующий тест 0 очков. Если программа распределит все призы правильно, то очки, которые она наберёт, зависят от того, сколько суммарно ученики провели времени в очереди. Точнее, за каждое взвешивание, во время которого в ожидании приза находилось p учеников, программа получает p баллов медлительности. Сколько процентов программа получает от очков за тест можно вычислить по формуле

$$100 \cdot \min(0,1 + 0,9^{100 \cdot P/Q - 99}; 1) \%,$$

где $Q = N \cdot M \cdot \log_2(N \cdot M)/2$, а P — сумма очков медлительности на этом тесте. Можно ожидать, что во всех тестах как расположение подарков на полке, так и то, какие ученики стоят в очереди и в каком порядке — случайно.

Пример.	Входные данные	Выходные данные
	3 2	
	1	? 1 2
	>	? 2 3
	<	! 2
	3	? 1 3
	>	! 1

В этом примере 3 приза и 2 ученика в очереди. Предположим, что порядок призов на полке: 3, 1, 2, т.е. в коробке номер 1 приз для участника, занявшего третье место, в коробке номер 2 — для ученика, занявшего первое место, а в последней коробке — для занявшего второе место.

Первым приходит ученик, занявший первое место. Программа просит учителя взвесить между собой призы 1 и 2 и получает результат, что в коробке 2 приз для ученика с результатом получше. После этого программа просит сделать взвешивание коробок 2 и 3, и приз получше находится в коробке 2. Программа делает вывод, что приз для ученика с первым местом находится в коробке 2 и выводит этот ответ.

Дальше подарок получает ученик, занявший третье место. Программа просит сделать взвешивание коробок 1 и 3, и приз получше находится в коробке 3. Таким образом приз для этого ученика находится в коробке 1, и программа выводит этот ответ.

Программа обработала все запросы и завершает работу. Она получает $2 \cdot 2 + 1 \cdot 1 = 5$ баллов медлительности (по 2 балла за взвешивания, пока ждало 2 ученика и ещё 1 балл за взвешивание, пока ждал 1 ученик), таким образом получая 100 % очков этого теста.

Примечание. Для обеспечения корректной работы программы следует после каждой выведенной строки очищать буфер вывода:

Язык	Команда
C	<code>fprintf(stdout, ...);</code> или <code>printf(...);</code> <code>fflush(stdout);</code>
C++	<code>cout << ... << endl;</code>
Java	<code>System.out.println(...);</code> <code>System.out.flush();</code>
Python	<code>sys.stdout.write(...)</code> или <code>print(...)</code> <code>sys.stdout.flush()</code>