

Задача А. Постфиксная запись

Имя входного файла: postfix.in
Имя выходного файла: postfix.out
Ограничение по времени: 1 second
Ограничение по памяти: 64 megabytes

В постфиксной записи (или обратной польской записи) операция записывается после двух операндов. Например, сумма двух чисел A и B записывается как $A B +$. Запись $B C + D *$ обозначает привычное нам $(B+C)*D$, а запись $A B C + D * +$ означает $A+(B+C)*D$. Достоинство постфиксной записи в том, что она не требует скобок и дополнительных соглашений о приоритете операторов для своего чтения.

Дано выражение в обратной польской записи. Определите его значение.

Формат входных данных

В единственной строке записано выражение в постфиксной записи, содержащее однозначные числа и операции $+$, $-$, $*$. Строка содержит не более 100 чисел и операций.

Формат выходных данных

Необходимо вывести значение записанного выражения. Гарантируется, что результат выражения, а также результаты всех промежуточных вычислений по модулю меньше 2^{31} .

Примеры

postfix.in	postfix.out
8 9 + 1 7 - *	-102

Задача В. Игра в пьяницу

Имя входного файла: `card-game.in`
Имя выходного файла: `card-game.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

В игре в пьяницу карточная колода раздаётся поровну двум игрокам. Далее они вскрывают по одной верхней карте, и тот, чья карта старше, забирает себе обе вскрытые карты, которые кладутся под низ его колоды. Тот, кто остаётся без карт — проигрывает.

Для простоты будем считать, что все карты различны по номиналу, а также, что самая младшая карта побеждает самую старшую карту («шестерка берет туза»).

Игрок, который забирает себе карты, сначала кладёт под низ своей колоды карту первого игрока, затем карту второго игрока (то есть карта второго игрока оказывается внизу колоды).

Напишите программу, которая моделирует игру в пьяницу и определяет, кто выигрывает. В игре участвует n карт, имеющих значения от 0 до $n - 1$, большая карта побеждает меньшую, карта со значением 0 побеждает карту $n - 1$.

Формат входных данных

Программа получает на вход три строки. В первой строке содержится целое чётное число n ($2 \leq n \leq 100\,000$). Вторая строка содержит $\frac{n}{2}$ чисел — карты первого игрока, а третья — $\frac{n}{2}$ карт второго игрока. Карты перечислены сверху вниз, то есть каждая строка начинается с той карты, которая будет открыта первой. Гарантируется, что каждая из карт встречается в колодах игроков ровно один раз.

Формат выходных данных

Программа должна определить, кто выигрывает при данной раздаче, и вывести слово «**first**» или «**second**», после чего вывести количество ходов, сделанных до выигрыша. Если на протяжении $2 \cdot 10^5$ ходов игра не заканчивается, программа должна вывести слово «**draw**».

Примеры

<code>card-game.in</code>	<code>card-game.out</code>
10 1 3 5 7 9 2 4 6 8 0	second 5

Задача С. Парикмахерская

Имя входного файла: `saloon.in`
Имя выходного файла: `saloon.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 megabytes

В парикмахерской работает один мастер. Он тратит на одного клиента ровно 20 минут, а затем сразу переходит к следующему, если в очереди кто-то есть, либо ожидает, когда придет следующий клиент.

Даны времена прихода клиентов в парикмахерскую (в том порядке, в котором они приходили).

Также у каждого клиента есть характеристика, называемая *степенью нетерпения*. Она показывает, сколько человек может максимально находиться в очереди перед клиентом, чтобы он дождался своей очереди и не ушел раньше. Если в момент прихода клиента в очереди находится больше людей, чем степень его нетерпения, то он решает не ждать своей очереди и уходит. Клиент, который обслуживается в данный момент, также считается находящимся в очереди.

Требуется для каждого клиента указать время его выхода из парикмахерской.

Формат входных данных

В первой строке вводится натуральное число N , не превышающее 100 — количество клиентов.

В следующих N строках вводятся времена прихода клиентов — по два числа, обозначающие часы и минуты (часы — от 0 до 23, минуты — от 0 до 59) и степень его нетерпения (неотрицательное целое число не большее 100) — максимальное количество человек, которое он готов ждать впереди себя в очереди. Времена указаны в порядке возрастания (все времена различны).

Гарантируется, что всех клиентов успеют обслужить до полуночи.

Если для каких-то клиентов время окончания обслуживания одного клиента и время прихода другого совпадают, то можно считать, что в начале заканчивается обслуживание первого клиента, а потом приходит второй клиент.

Формат выходных данных

В выходной файл выведите N пар чисел: времена выхода из парикмахерской 1-го, 2-го, ..., N -го клиента (часы и минуты). Если на момент прихода клиента человек в очереди больше, чем степень его нетерпения, то можно считать, что время его ухода равно времени прихода.

Примеры

<code>saloon.in</code>	<code>saloon.out</code>
3	10 20
10 0 0	10 40
10 1 1	10 2
10 2 1	
5	1 20
1 0 100	2 20
2 0 0	2 1
2 1 0	2 40
2 2 3	2 3
2 3 0	