

А. Две заправки

Есть круговая дорога длиной N километров и километровые столбы на ней (столбы с номерами 0 и N совпадают). По дороге можно ездить в обе стороны.

На дороге есть две заправки. К какой из них вы сейчас ближе всего находитесь?

Даны целые числа N, A, B, X — длина дороги, координаты заправок и ваше местоположение на дороге, $0 \leq N \leq 10^9$, $A, B, X \in [0, N - 1]$.

Программа должна вывести расстояние до ближайшей заправки и соответствующую ей букву. Если две заправки находятся от вас на одинаковом расстоянии, вместо буквы выведите символ '='.

Input	Output
12 8 4 6	2 =
12 11 6 2	3 B
145325 123 67354 67354	0 B

В. Нарушитель и ДПС

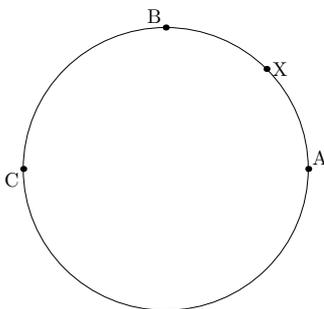
На кольцевой дороге длиной N километров в точке X находится нарушитель. Также на дороге есть два патруля ДПС и съезд с кольца. Нарушитель хочет добраться до съезда, но ему мешают машины ДПС, мимо которых он не может проехать.

Известны 5 чисел: длина кольцевой дороги N ($N \leq 10^9$), координата нарушителя и три остальные координаты A, B и C (машины ДПС и съезд, но неизвестно какое число чекму или кому соответствует). Гарантируется что все числа A, B, C, X попарно различны и меньше N .

Программа должна вывести координату съезда с шоссе.

Input	Output
80 10 60 0 20	60
80 0 10 20 60	20

Иллюстрация к первому тесту: здесь точки A и B это машины ДПС, а точка C — съезд с трассы.



С. Для данного натурального N ($1 < N < 10^{12}$) требуется вынести квадрат из под квадратного корня в выражении \sqrt{N} .

На вход программе подаётся одно натуральное число N .

Требуется вывести наибольшее натуральное число A , такое, что число N может быть представлено в виде $A \cdot \sqrt{B} = \sqrt{N}$, где B — также натуральное число.

Input	Output
48	4
30	1
121	11

Д. «Скрещивание» массивов

Есть два массива A и B одинаковой длины N . Каждый состоит из чисел $1 \dots N$, взятых ровно один раз, в каком-то порядке. Процесс скрещивания состоит из следующих этапов:

- В массиве A выделяется кусок и переносится в массив-результат C на те же индексы.

A: 1 4 5 3 2 6 7 8 9

B: 4 3 8 6 9 1 2 7 5

C: - - 5 3 2 6 - - -

- Оставшиеся места в массиве C слева направо заполняются элементами из массива B , причём они должны следовать в том же порядке, в каком они встречаются в B и в массиве C не должно оказаться повторов.

A: 1 4 5 3 2 6 7 8 9

B: 4 3 8 6 9 1 2 7 5

C: 4 8 5 3 2 6 9 1 7

В первой строке записаны размер массивов N и два индекса: L и R , задающие полуинтервал $[L, R)$ в массиве A ($1 \leq N \leq 10^3$, $0 \leq L < R \leq N$). Во второй и третьей строках записаны числа массивов A и B , соответственно.

Программа должна вывести получившийся после «скрещивания» массив.

Input	Output
9 2 6 1 4 5 3 2 6 7 8 9 4 3 8 6 9 1 2 7 5	4 8 5 3 2 6 9 1 7
9 0 9 1 4 5 3 2 6 7 8 9 4 3 8 6 9 1 2 7 5	1 4 5 3 2 6 7 8 9

Е. Есть ли половина

В данном массиве A из целых чисел определить — можно ли найти два таких индекса i, j ($i \neq j$), что $A[i] = 2A[j]$.

На вход программе даётся натуральное число N ($2 \leq N < 10^5$). Затем вводится N натуральных чисел, абсолютная величина каждого из которых не превосходит 10^9 .

Если числа с указанными свойствами в массиве есть, вывести 'YES', иначе вывести 'NO'.

Input	Output
6 1 2 3 4 5 6	YES
2 3 0 5	NO
5 -100 30 0 2 50	NO

Ф. Пары чисел с общей делимостью

Есть массив из N чисел (N — чётное и не превосходит 10^5) и число P ($1 < P < 10^9$).

Требуется разбить элементы на $\frac{N}{2}$ пар таким образом, чтобы сумма каждой пары делилась нацело на P .

Если это возможно, выведите YES, иначе выведите NO.

В первой строке даны числа N и P . Во второй N положительных целых чисел.

Программа должна вывести YES или NO.

Input	Output
10 5 8 1 2 9 4 5 6 7 10 3	YES
6 10 1 2 3 4 5 6	NO

Г. Преобразование строк

Есть две строки равной длины $s1$ и $s2$, которые состоят только их символов L, R и _.

Символ L разрешается двигать *влево*, если его левый сосед это символ _ — эти соседние символы при этом меняются местами.

Аналогично, символ R разрешается двигать *вправо*, если его правый сосед это символ _ — эти соседние символы при этом меняются местами.

Требуется выяснить, можно ли такими операциями получить из строки $s1$ строку $s2$.

На вход даётся две строки равного размера, не превосходящего 10^5 .

Программа должна вывести YES или NO.

Input	Output
_L_R_R_ L_____RR	YES
R_L_ __LR	NO
R R	NO

Указания к примерам:

В строке $_L_R_R_$ символ L можно сдвинуть влево, а символы R — вправо и получить нужный результат.

Символ R не удастся сдвинуть правее символа L, поэтому из $R_L_$ нельзя получить $_LR$.

Символ R нельзя двигать влево, получить $R_$ нельзя.

Н. Маршрут из левого столбца

Есть прямоугольная таблица размера $M \times N$ с натуральными числами.

В начале можно выбрать любую клетку *крайнего левого столбца* и двигаться по таблице вправо, по диагонали вправо-вверх и по диагонали вправо-вниз. Число в каждой следующей клетке вашего маршрута должно быть строго больше предыдущего.

Требуется определить максимальное количество ходов, которое удастся сделать при таких ограничениях.

В первой строке даны числа M и N — количество строк и столбцов таблицы ($1 \leq M, N \leq 1000$).

Затем в M строках записано по N натуральных чисел в каждой.

Программа должна вывести одно число — ответ на вопрос задачи.

Input	Output
4 4 2 4 3 5 5 4 9 3 3 4 2 11 10 9 13 15	3
3 3 3 2 4 2 1 9 1 1 7	0

Пояснения к тестам:

1 тест: начиная в крайнем левом столбце, можно сделать 3 шага по указанным правилам. Клетки маршрута отмечены на рисунке.

2	4	3	5
5	4	9	3
3	4	2	11
10	9	13	15

2 тест: в этой таблице из клеток левого столбца нельзя сделать ни одного шага, поэтому ответ 0.

3	2	4
2	1	9
1	1	7