Сказка о долге

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Дождь лил с такой силой, будто само небо готовилось упасть на землю. Я шёл по лабиринту тёмных безлюдных улиц, не обращая внимания на глубокие лужи. Я больше не был славным рыцарем в сияющих доспехах, теперь я был лишь изгоем, которого никто не желал бы увидеть на своём пороге в этот дождливый вечер.

Сказочное Королевство уже не то, что прежде. Новый Король под громкими лозунгами затеял перестройку политической и экономической систем, которая привела к краху экономики и разгулу преступности всех мастей. Ремесленники, купцы и даже владельцы мануфактур были вынуждены платить разбойникам или рыцарям за спокойное существование и возможность дальше заниматься бизнесом. Всё больше людей баловались Синим Чаем — нелегальным порошком, названным так из-за того, что многие хранили его в популярных когда-то чайных шкатулках. Сказать, что настали тёмные времена — не сказать ничего.

Люди рассказывали забавную историю об n купцах, которые занимали друг у друга снова и снова, но никогда не отдавали долги. В конце концов выяснилось, что i-ый купец должен j-ому a_{ij} золотых, причём это были огромные суммы, которые никому из них никогда не накопить. Тогда они пошли к Королю, чтобы решить эту проблему.

Им повезло. Добрый Волшебник тогда ещё работал на Короля. Он подметил, что для любого упорядоченного набора купцов $(i_1, i_2, \ldots, i_k)$ можно просто добавить произвольное целое число s ко всем долгам $a_{i_1 i_2}, a_{i_2 i_3}, \ldots, a_{i_{k-1} i_k}, a_{i_k i_1}$, но только если $\min\left(a_{i_1 i_2}, a_{i_2 i_3}, \ldots, a_{i_{k-1} i_k}, a_{i_k i_1}\right) + s \geqslant 0$. Таким образом можно было увеличивать или уменьшать долги в зависимости от того, положительным или отрицательным было число s. Изменяя таким образом матрицу долгов, Добрый Волшебник добился того, чтобы сумма её элементов была минимальна. Непростая была задача, но старик тогда ещё был в хорошей форме.

Формат входных данных

В первой строке содержится единственное целое число n ($2 \leqslant n \leqslant 1000$) — количество купцов.

Далее в n строках записаны по n целых чисел. В i-й строке на j-й позиции содержится число a_{ij} ($0 \leqslant a_{ij} \leqslant 10^9$, $a_{ii} = 0$) — количество золотых, которое i-й купец должен j-му.

Формат выходных данных

Выведите n строк по n целых неотрицательных чисел в каждой. В i-й строке на j-й позиции выведите b_{ij} — количество золотых, которое i-й купец остался должен j-му после взаимозачёта всех долгов. Взаимозачёт должен производиться так, чтобы сумма всех b_{ij} была минимальна. Если существует несколько возможных ответов, выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
3 0 10 1 1 0 15 20 1 0	0 0 0 5 0 0 5 0 0
7 0 1 2 3 4 5 6 6 0 1 2 3 4 5 5 6 0 1 2 3 4 4 5 6 0 1 2 3 3 4 5 6 0 1 2 2 3 4 5 6 0 1 1 2 3 4 5 6 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Замечание

Пояснение к первому примеру:

1) Рассмотрим цепочку a_{13}, a_{32}, a_{21}. Минимум из них составляет 1. Вычитаем 1 из всех трёх элементов.

2) Рассмотрим цепочку a_{12}, a_{23}, a_{31}. Минимум из них составляет 10. Вычитаем 10 из всех трёх элементов. Теперь матрица имеет вид:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 5 \\ 10 & 0 & 0 \end{pmatrix}$$

3) Рассмотрим цепочку a_{12}, a_{21}. По условию мы можем также прибавлять произвольное число ко всем элементам цепочки. Прибавим к этим элементам 5. Теперь матрица имеет вид:

$$\begin{pmatrix} 0 & 5 & 0 \\ 5 & 0 & 5 \\ 10 & 0 & 0 \end{pmatrix}$$

4) Рассмотрим цепочку a_{12}, a_{23}, a_{31}. Минимум из них составляет 5. Вычитаем 5 из всех трёх элементов. Теперь матрица имеет вид:

$$\begin{pmatrix} 0 & 0 & 0 \\ 5 & 0 & 0 \\ 5 & 0 & 0 \end{pmatrix}$$