

Задача А. Задача о назначениях

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Дана целочисленная матрица C размера $n \times n$. Требуется выбрать n ячеек так, чтобы в каждой строке и каждом столбце была выбрана ровно одна ячейка, а сумма значений в выбранных ячейках была минимальна.

Формат входных данных

Первая строка входного файла содержит n ($2 \leq n \leq 300$). Каждая из последующих n строк содержит по n чисел: C_{ij} . Все значения во входном файле неотрицательны и не превосходят 10^6 .

Формат выходных данных

В первую строку выходного файла выведите одно число — искомая минимизируемая величина. Далее выведите n строк по два числа в каждой — номер строки и столбца клетки, участвующей в оптимальном назначении.

Пары чисел можно выводить в произвольном порядке.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3 | 3 |
| 3 2 1 | 2 1 |
| 1 3 2 | 3 2 |
| 2 1 3 | 1 3 |

Замечание

Обратите внимание, что эту задачу можно сдать с помощью алгоритма за $O(n^3)$, без использования Венгерского алгоритма, а используя только максимальный поток минимальной стоимости.

Задача С. Расписание работ

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Для защиты местной свалки от расхитителей мусора были наняты несколько охранников. Необходимо составить расписание, по которому охранники будут работать в парах, по одной паре каждую ночь. Начальник свалки нанял вас, чтобы написать программу, которая по характеристикам охранников определит максимальное возможное количество охранников, которые могут быть в расписании (остальные будут уволены). Обратите внимание, что каждый охранник может работать только в одной паре и не может работать один.

Формат входных данных

В первой строке находится одно число $N \leq 222$ — количество охранников. В следующих строках находится неупорядоченные пары (i, j) , каждая из которых означает, что охранники с номерами i и j могут работать вместе, так как можно найти форму, которая устроит их обоих. (На свалке используются разные части формы для разных охранников, например шлемы, штаны, куртки. Нельзя одеть маленький шлем на охранника с большой головой, или большие ботинки на охранника с маленькими ногами). Ввод заканчивается концом файла.

Формат выходных данных

Необходимо вывести оптимальное расписание. В первой строке выведите четное число C , количество охранников в расписании. После этого выведите $C/2$ строк, содержащих по 2 числа i, j каждая, обозначающих, что охранники i и j будут работать вместе.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3 | 2 |
| 1 2 | 1 2 |
| 2 3 | |
| 1 3 | |

Задача D. Расписание работ

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.3 секунд
Ограничение по памяти: 256 мегабайт

Для защиты местной свалки от расхитителей мусора были наняты несколько охранников. Необходимо составить расписание, по которому охранники будут работать в парах, по одной паре каждую ночь. Начальник свалки нанял вас, чтобы написать программу, которая по характеристикам охранников определит максимальное возможное количество охранников, которые могут быть в расписании (остальные будут уволены). Обратите внимание, что каждый охранник может работать только в одной паре и не может работать один.

Формат входных данных

В первой строке находится одно число $N \leq 222$ — количество охранников. В следующих строках находится неупорядоченные пары (i, j) , каждая из которых означает, что охранники с номерами i и j могут работать вместе, так как можно найти форму, которая устроит их обоих. (На свалке используются разные части формы для разных охранников, например шлемы, штаны, куртки. Нельзя одеть маленький шлем на охранника с большой головой, или большие ботинки на охранника с маленькими ногами). Ввод заканчивается концом файла.

Формат выходных данных

Необходимо вывести оптимальное расписание. В первой строке выведите четное число C , количество охранников в расписании. После этого выведите $C/2$ строк, содержащих по 2 числа i, j каждая, обозначающих, что охранники i и j будут работать вместе.

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3 | 2 |
| 1 2 | 1 2 |
| 2 3 | |
| 1 3 | |

Задача Е. Банах

Имя входного файла: `banach.in`
Имя выходного файла: `banach.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Стефан, Дэвид и Феликс готовят соревнование по правилам ICPC. Стефан предлагают следующую задачу:

Дано N точек на плоскости, а также N векторов. Найдите однозначное соответствие между векторами и точками так, что если передвинуть каждую точку на соответствующий вектор, то расстояние между **любой** парой точек не уменьшится.

Дэвид справился с решением этой задачи довольно быстро и утверждает, что эта задача слишком проста для этого соревнования. Для того, чтобы убедить его, Феликс предложил следующее усложнение: среди всех возможных решений, найдите то, которое максимизирует сумму квадратов всех результирующих попарных расстояний.

Дэвиду все еще кажется, что задача слишком простая, а вам?

Формат входных данных

Первая строка входного файла содержит число N , количество точек и векторов ($1 \leq N \leq 500$).

Следующие N строк описывают точки. Каждая из них содержит два числа px_i и py_i ($0 \leq |px_i|, |py_i| \leq 10\,000$).

Затем следуют N строк с описанием векторов. Каждый вектор задан двумя числами vx_i и vy_i ($0 \leq |vx_i|, |vy_i| \leq 10\,000$).

Формат выходных данных

Если способ сопоставить точки и вектора таким образом, чтобы все попарные расстояния не уменьшились существует, выведите "Yes" в первой строке выходного файла. На следующей строке, выведите N различных чисел от 1 до N , i -е из которых равно номеру вектора, сопоставленного i -й точке.

Не забудьте выбрать ответ с максимально возможной суммой квадратов всех результирующих попарных расстояний. Если существует несколько возможных решений, выведите любое из них.

Если требование выполнить невозможно, выведите "No" в единственной строке выходного файла.

Примеры

| <code>banach.in</code> | <code>banach.out</code> |
|-----------------------------------|-------------------------|
| 2 0 0 1 0 2 0 -2 0 | Yes 2 1 |
| 2 2 2 2 2 -1 -1 -1 -1 | Yes 1 2 |

Замечание

В первом примере, существует только два возможных способа сопоставить векторы точкам. В обоих сопоставлениях, "1 2" и "2 1", расстояние между каждой парой точек не уменьшается. В первом случае сумма квадратов расстояний между всеми парами точек равна 9, но во втором случае она равна 25. Поэтому единственным правильным решением является "2 1".

Во втором примере, опять же, только два случая установить соответствие. В обоих случаях, расстояние между всеми парами точек не уменьшается, и сумма квадратов попарных расстояний равна 0. Поэтому, оба ответа корректны.

Задача F. Баланс

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

Назовем матрицу A размера $N \times N$ *сбалансированной*, если $A[i][j] + A[i + 1][j + 1] = A[i + 1][j] + A[i][j + 1]$ для всех $1 \leq i, j \leq N - 1$. Вам дана матрица A размера $N \times N$. Ваша задача вывести другую матрицу B того же размера, что B сбалансированна и $B[i][j] \geq A[i][j]$ для всех $1 \leq i, j \leq N$. Более того, матрица B должна иметь минимально возможную сумму всех элементов.

Формат входных данных

Первая строка содержит число N , число строк и столбцов в матрице ($1 \leq N \leq 50$). Каждая из следующих N строк содержит N чисел. Вместе они описывают матрицу A . Гарантируется, что $0 \leq A[i][j] \leq 35\,000$ для всех $1 \leq i, j \leq N$.

Формат выходных данных

В первой строке выведите сумму значений в сбалансированной матрице B . В следующих N строках, выведите сбалансированную матрицу в том же формате, как и во входных данных.

Значения в выведенной матрице ничем не ограничены (например, они могут превосходить 35 000).

Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 4 | 16 |
| 1 1 1 1 | 1 1 1 1 |
| 1 1 1 1 | 1 1 1 1 |
| 1 1 1 0 | 1 1 1 1 |
| 1 1 1 1 | 1 1 1 1 |

Задача G. Совершенство

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Павел Андреевич очень любит все совершенное. Его новая любовь - совершенные паросочетания! Не огорчайте Павла Андреевича, проверьте, есть ли в данном графе совершенное паросочетание.

Формат входных данных

В первой строке даны два числа N и M — количество вершин и количество ребер в G . $N \leq 100$. Следующие M строк содержат числа a_i и b_i — ребра графа. Гарантируется отсутствие петель и кратных ребер

Формат выходных данных

Выведите слово «YES» или слово «NO» — ответ на поставленную задачу.

Примеры

| стандартный ввод | стандартный вывод |
|--|-------------------|
| 6 7 1 2 2 3 1 3 5 6 6 4 4 5 6 2 | YES |
| 3 3 1 2 2 3 1 3 | NO |