

Задача А. Чистые стратегии

Имя входного файла: `pure.in`
Имя выходного файла: `pure.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана матрица игры размера $m \times n$. Требуется найти все чистые стратегии игроков, приводящие к позиции равновесия.

Формат входного файла

В первой строке даны два целых числа m и n ($1 \leq m, n \leq 100$) — размеры матрицы. Далее следуют m строк по n целых чисел a_{ij} в каждой ($|a_{ij}| \leq 10^6$).

Формат выходного файла

В первой строке выведите два целых числа p_1 и p_2 — число оптимальных чистых стратегий первого и второго игроков соответственно.

Во второй строке в любом порядке выведите через пробел p_1 чисел — номера строк, оптимальных для первого игрока.

В следующей строке в любом порядке выведите через пробел p_2 чисел — номера столбцов, оптимальных для второго игрока.

Строки и столбцы нумеруются с единицы. Если игра не содержит позиций равновесия в чистых стратегиях, выведите в выходной файл лишь «0 0».

Примеры

<code>pure.in</code>	<code>pure.out</code>
3 2 0 0 1 1 1 1	2 2 2 3 1 2
2 2 -1 1 1 -1	0 0

Задача В. Смешанные стратегии

Имя входного файла: `mixed.in`
Имя выходного файла: `mixed.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана матрица игры размера $m \times n$. Требуется найти оптимальные смешанные стратегии для каждого из игроков.

Формат входного файла

В первой строке даны два целых числа m и n ($1 \leq m, n \leq 30$) — размеры матрицы. Далее следуют m строк по n целых чисел a_{ij} в каждой ($|a_{ij}| \leq 1000$).

Формат выходного файла

В первой строке выходного файла выведите m чисел — оптимальная стратегия первого игрока. Во второй строке выходного файла выведите n чисел — оптимальная стратегия второго игрока. Если оптимальных стратегий несколько, выведите любую из них.

Примеры

<code>mixed.in</code>	<code>mixed.out</code>
2 2	0.50 0.50
3 1	0.25 0.75
0 2	

Задача С. Графический метод

Имя входного файла: `nx2.in`
Имя выходного файла: `nx2.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана матрица игры размера $2 \times n$. Требуется найти оптимальные смешанные стратегии для каждого из игроков.

Формат входного файла

В первой строке дано целое число n ($1 \leq n \leq 100$) — размер матрицы. Далее следуют две строки по n целых чисел a_{ij} в каждой ($|a_{ij}| \leq 1000$).

Формат выходного файла

В первой строке выходного файла выведите два числа — оптимальная стратегия первого игрока. Во второй строке выходного файла выведите n чисел — оптимальная стратегия второго игрока. Если оптимальных стратегий несколько, выведите любую из них.

Примеры

<code>nx2.in</code>	<code>nx2.out</code>
2 3 1 0 2	0.50 0.50 0.25 0.75
3 2 2 1 1 5 0	1.0 0.0 0.0 0.0 1.0

Задача D. Вполне смешанная игра

Имя входного файла: `completely.in`
Имя выходного файла: `completely.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана матрица вполне смешанной игры размера $n \times n$. Требуется найти оптимальные смешанные стратегии для каждого из игроков.

Формат входного файла

В первой строке дано целое число n ($1 \leq n \leq 50$) — размер матрицы. Далее следуют n строк по n целых чисел a_{ij} в каждой ($|a_{ij}| \leq 1000$).

Формат выходного файла

В первой строке выходного файла выведите n чисел — оптимальная стратегия первого игрока. Во второй строке выходного файла выведите n чисел — оптимальная стратегия второго игрока. Если оптимальных стратегий несколько, выведите любую из них.

Примеры

<code>completely.in</code>	<code>completely.out</code>
2	0.50 0.50
3 1	0.25 0.75
0 2	
3	0.375 0.125 0.500
2 2 1	0.625 0.250 0.125
1 5 0	
2 1 3	

Задача Е. Равновесие по Нэшу

Имя входного файла: `nash.in`
Имя выходного файла: `nash.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана биматричная игра $\Gamma(A, B)$. Матрицы A и B имеют размер $m \times n$. Требуется найти ситуации равновесные по Нэшу.

Формат входного файла

В первой строке даны два целых числа m и n ($1 \leq m, n \leq 100$) — размеры матриц. Далее следует описание матрицы A — m строк по n целых чисел a_{ij} в каждой. Затем идет описание матрицы B в аналогичном формате. Все числа во входном файле по абсолютному значению не превосходят 10^6 .

Формат выходного файла

В первой строке выходного файла выведите одно число k — число равновесных по Нэшу ситуаций. Каждая из следующих k строк должна содержать описание ситуации — номер строки и столбца.

Примеры

<code>nash.in</code>	<code>nash.out</code>
2 2 4 0 0 1 1 0 0 4	2 1 1 2 2
2 2 7 0 10 3 7 10 0 3	1 2 2

Задача F. Сильное равновесие

Имя входного файла: `strong.in`
Имя выходного файла: `strong.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана биматричная игра $\Gamma(A, B)$. Матрицы A и B имеют размер $m \times n$. Требуется найти сильно равновесные ситуации.

Формат входного файла

В первой строке даны два целых числа m и n ($1 \leq m, n \leq 100$) — размеры матриц. Далее следует описание матрицы A — m строк по n целых чисел a_{ij} в каждой. Затем идет описание матрицы B в аналогичном формате. Все числа во входном файле по абсолютному значению не превосходят 10^6 .

Формат выходного файла

В первой строке выходного файла выведите одно число k — число сильно равновесных ситуаций. Каждая из следующих k строк должна содержать описание ситуации — номер строки и столбца.

Примеры

<code>strong.in</code>	<code>strong.out</code>
2 2 3 0 0 1 1 0 0 4	1 2 2 Текст
2 2 7 0 10 3 7 10 0 3	0

Задача Г. Парето

Имя входного файла: `pareto.in`
Имя выходного файла: `pareto.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана биматричная игра $\Gamma(A, B)$. Матрицы A и B имеют размер $m \times n$. Требуется найти оптимальные по Парето ситуации.

Формат входного файла

В первой строке даны два целых числа m и n ($1 \leq m, n \leq 100$) — размеры матриц. Далее следует описание матрицы A — m строк по n целых чисел a_{ij} в каждой. Затем идет описание матрицы B в аналогичном формате. Все числа во входном файле по абсолютному значению не превосходят 10^6 .

Формат выходного файла

В первой строке выходного файла выведите одно число k — число оптимальных по Парето ситуаций. Каждая из следующих k строк должна содержать описание ситуации — номер строки и столбца.

Примеры

pareto.in	pareto.out
2 2 4 0 0 1 1 0 0 4	2 1 1 2 2
2 2 7 0 10 3 7 10 0 3	3 1 1 1 2 2 1