

Задача А. В поисках невест

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

Однажды король Флатландии решил отправить k своих сыновей на поиски невест. Всем известно, что во Флатландии n городов, некоторые из которых соединены дорогами. Король живет в столице, которая имеет номер 1, а город с номером n знаменит своими невестами.

Итак, король повелел, чтобы каждый из его сыновей добрался по дорогам из города 1 в город n . Поскольку, несмотря на обилие невест в городе n , красивых среди них не так много, сыновья опасаются друг друга. Поэтому они хотят добраться до цели таким образом, чтобы никакие два сына не проходили по одной и той же дороге (даже в разное время). Так как король любит своих сыновей, он хочет, чтобы среднее время сына в пути до города назначения было минимально.

Формат входного файла

В первой строке входного файла находятся числа n , m и k — количество городов и дорог во Флатландии и сыновей короля, соответственно ($2 \leq n \leq 200$, $1 \leq m \leq 2000$, $1 \leq k \leq 100$). Следующие m строк содержат по три целых положительных числа каждая — города, которые соединяет соответствующая дорога и время, которое требуется для ее прохождения (время не превышает 10^6). По дороге можно перемещаться в любом из двух направлений, два города могут быть соединены несколькими дорогами.

Формат выходного файла

Если выполнить повеление короля невозможно, выведите на первой строке число -1 . В противном случае выведите на первой строке минимальное возможное среднее время (с точностью 5 знаков после десятичной точки), которое требуется сыновьям, чтобы добраться до города назначения, не менее чем с пятью знаками после десятичной точки. В следующих k строках выведите пути сыновей, сначала число дорог в пути и затем номера дорог в пути в том порядке, в котором их следует проходить. Дороги нумеруются, начиная с единицы, в том порядке, в котором они заданы во входном файле.

Пример

<code>brides.in</code>	<code>brides.out</code>
5 8 2	3.00000
1 2 1	3 1 5 6
1 3 1	3 2 7 8
1 4 3	
2 5 5	
2 3 1	
3 5 1	
3 4 1	
5 4 1	

Задача В. Матрица

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

Вам дана матрица целых чисел размера $n \times n$. Ваша задача — найти такой набор координат (k_i, l_i) , в котором каждая координата k_i и каждая координата l_i встречается ровно один раз, такой, чтобы минимизировать сумму выбранных элементов.

Формат входного файла

Первая строка входного файла содержит одно целое число n ($1 \leq n \leq 239$). Следующие n строк содержат по n целых чисел в каждой. Все эти числа не превосходят по абсолютной величине 10^6 .

Формат выходного файла

Первая строка должна содержать значение оптимизируемой функции. В следующие n строк необходимо записать пары чисел, описывающих выбранные ячейки. Первой координатой выводится номер строки.

Пример

<code>matrix.in</code>	<code>matrix.out</code>
2	2
1 1	1 1
1 1	2 2

Задача С. Две кучки

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

Два игрока играют в игру. На столе лежат две кучки: в первой a камней, во второй — b . Игроки ходят по очереди. Каждым ходом игрок выбирает одну кучку и берет какое-то количество камней из нее. Первый игрок может брать a_1, a_2, \dots, a_k камней, второй — b_1, b_2, \dots, b_l . Проигрывает тот, кто не может сделать ход. Определите победителя!

Формат входного файла

В первой строке записаны a и b . Во второй строке записаны k и последовательность a_i , на третьей — l и b_i . $1 \leq a, b \leq 1000$, $1 \leq k, l \leq 10$, $1 \leq a_i, b_j \leq 1000$.

Формат выходного файла

Если выигрывает первый игрок, выведите `First`. Иначе выведите `Second`.

Пример

<code>heaps.in</code>	<code>heaps.out</code>
2 2 2 1 2 1 1	First
2 2 1 1 2 1 2	Second