

Задача А. Ремонт дорог

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

Славная страна Матроидландия состоит из N городов, соединенных M шоссе. Каждое шоссе соединяет два различных города, и по каждому шоссе можно перемещаться в обе стороны. Некоторые пары городов могут быть соединены более чем одной дорогой. Правительство Матроидландии решило произвести модернизацию дорожной системы в стране. В стране есть несколько компаний, занимающихся ремонтом дорог. Они уже договорились, кто что будет ремонтировать, таким образом, для каждого шоссе указана компания, которая будет его ремонтировать.

Чтобы поддержать малый бизнес в развивающейся стране, Правительство решило, что каждая компания может ремонтировать не более, чем одно шоссе. Так как жители Матроидландии не очень любят думать, между любыми двумя городами должен существовать максимум один путь по отремонтированным дорогам. Ваша задача состоит в том, чтобы определить максимальное число шоссе, которые можно отремонтировать, чтобы все были довольны.

Формат входных данных

Первая строка входного файла содержит два числа — N и M ($1 \leq N \leq 100$, $0 \leq M \leq 5000$). Следующие M строк содержат описания шоссе. Каждое описание состоит из двух номеров городов u и v , которые соединяет шоссе, и номера c компании, которая это шоссе собирается ремонтировать ($1 \leq v, u \leq N$, $1 \leq c \leq 200$).

Формат выходных данных

На первой строке выведите единственное число K — искомое максимальное количество шоссе. На следующей строке выведите номера этих шоссе в любом порядке. Если существует несколько решений, выведите любое.

Примеры

highways.in	highways.out
4 5 1 2 1 3 1 1 2 3 1 1 4 2 3 4 3	3 1 4 5

Задача В. Два остовных дерева

Имя входного файла:	<code>twost.in</code>
Имя выходного файла:	<code>twost.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Задан неориентированный граф, рёбра которого можно разбить на два непересекающихся остовных дерева. Вам необходимо найти одно из таких разбиений.

Формат входных данных

Первая строка входного файла содержит два натуральных числа N ($N \leq 600$) и M — количество вершин и ребер в графе. Следующие M строк содержат описания ребер графа. Каждое ребро задается номерами концов. Гарантируется, что в графе нет петель, но могут быть кратные ребра. Вершины и рёбра графа нумеруются с единицы.

Формат выходных данных

Выходной файл должен содержать искомое разбиение рёбер графа. В первой строке выведите номера рёбер, которые войдут в первое остовное дерево, во второй — номера рёбер, которые войдут во второе остовное дерево. Каждое ребро графа должно появиться ровно в одной из этих двух строк.

Примеры

twost.in	twost.out
6 10 1 2 2 3 3 4 4 5 5 6 6 1 6 2 2 5 5 3 3 4	3 4 5 6 8 1 2 7 9 10
18 34 4 13 3 5 11 12 16 7 9 3 15 2 13 9 6 2 12 10 11 1 5 4 8 12 10 9 6 10 13 18 1 4 2 6 14 16 2 4 17 11 10 3 5 3 17 5 9 12 1 3 14 8 18 5 4 12 8 13 16 14 18 3 7 15 15 17 7 11	1 2 3 4 5 6 7 8 9 12 13 14 16 20 27 30 32 10 11 15 17 18 19 21 22 23 24 25 26 28 29 31 33 34

Задача С. Дорожные вопросы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Короли Берляндии и Флатландии в очередной раз запланировали дорожные реформы в своих государствах. На этот раз они решили действовать сообща.

Берляндия и Флатландия — два крупных государства, в каждом из которых есть n городов, между которыми правители хотят построить дороги. В результате тендера короли выбрали m подрядчиков, i -й из которых построит двустороннюю дорогу между городами a_i и b_i в Берляндии и двустороннюю дорогу между городами c_i и d_i во Флатландии, в результате чего суммарное благополучие этих стран возрастёт на w_i . Обратите внимание, что w_i может быть отрицательным, так как не все подрядчики добросовестно выполняют свою работу. В случае если подрядчик будет нанят, он построит обе дороги, нельзя попросить его построить только одну из них.

Короли Берляндии и Флатландии очень заботятся об экономичности транспортных систем своих стран, а именно они никогда не допустят, чтобы между какими-то двумя городами в их странах существовало два различных простых пути между этими городами. Путь называется простым, если он посещает каждый город не более одного раза, два пути называются различными, если различаются множества дорог, которые эти пути используют. Обратите внимание, что короли могут построить дорожную систему, в которой между некоторыми двумя городами нет пути по дорогам. Иными словами, короли хотят, чтобы графы, образованные дорогами каждой страны, были лесами — множествами неориентированных деревьев.

Короли ещё не решили, сколько же дорог они построят, поэтому просят вас для каждого k от 1 до m определить, какого максимального суммарного благополучия стран они могут добиться, если наймут k подрядчиков, при условии, что дорожные сети каждой страны должны получиться экономичными.

Формат входных данных

В первой строке задано два целых числа n и m ($2 \leq n \leq 800$, $1 \leq m \leq 800$) — количество городов в каждой из стран и число подрядчиков, соответственно.

В следующих m строках заданы описания подрядчиков.

Описание i -го подрядчика задано пятью целыми числами a_i , b_i , c_i , d_i , и w_i ($1 \leq a_i, b_i, c_i, d_i \leq n$, $a_i \neq b_i$, $c_i \neq d_i$, $-10^9 \leq w_i \leq 10^9$) — номера городов в Берляндии, которые соединит дорога i -го подрядчика, номера городов во Флатландии, которые соединит дорога i -го подрядчика и величина, на которую возрастёт суммарное благополучие стран, если дороги i -го подрядчика будут построены.

Формат выходных данных

Выведите m строк, в i -й строке выведите одно целое число — максимально возможное суммарное благополучие стран, если будут наняты **ровно** i подрядчиков или «Impossible» (без кавычек), если нельзя получить экономичную дорожную сеть, наняв i подрядчиков.

Примеры

стандартный ввод	стандартный вывод
4 3 1 2 1 2 7 1 3 2 1 8 2 3 3 2 6	8 14 Impossible
6 4 1 2 1 3 34 2 3 3 2 11 2 4 3 1 5 2 1 3 5 8	34 45 24 Impossible
3 2 3 1 2 3 -9 2 3 1 3 -21	-9 -30

Замечание

Рассмотрим первый пример.

При $k = 1$ выгодно нанять подрядчика с номером 2. При $k = 2$ выгодно нанять подрядчиков с номерами 2 и 3. Единственный способ нанять трёх подрядчиков — это нанять всех подрядчиков, однако получившаяся дорожная сеть не будет экономичной так как во Флатландии будет существовать два простых пути между городами 1 и 2 (по дороге, построенной первым подрядчиком, и по дороге, построенной вторым подрядчиком).

В третьем примере все возможные множества подрядчиков построят эффективные дорожные сети.

Система оценки

Тесты к этой задаче состоят из семи групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп.

Группа	Баллы	Дополнительные ограничения			Необх. группы	Комментарий
		n	m	w_i		
0	0	—	—	—	—	Тесты из условия.
1	15	$n \leq 40$	$m \leq 20$	—	0	
2	6	—	$m = n - 1$	—	—	$a_i = i, b_i = i + 1$
3	15	—	—	$w_i = 0$	—	
4	17	$n \leq 70$	$m \leq 70$	—	0, 1	
5	8	$n \leq 150$	$m \leq 150$	—	0, 1, 4	
6	14	$n \leq 500$	$m \leq 500$	—	0, 1, 4, 5	
7	25	—	—	—	0, 1, 2, 3, 4, 5, 6	