Задача A. Дорожная реформа

Имя входного файла:	`roads.in`
Имя выходного файла:	`roads.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта
Максимальная оценка за задачу:	120 баллов

Как известно, Флатландское государство состоит из n городов, которые соединены системой двусторонних железнодорожных путей и автомобильных шоссе.

Недавно, после очередной поездки по стране, президент Флатландии остался крайне недовольным состоянием дорог и сделал вывод о необходимости радикальных реформ системы путей и сообщений.

Так как дорог в государстве за долгие годы было построено много, и почти все они нуждались в капитальном ремонте, то комиссией было решено, что необходимо закрыть часть существующих дорог, а на восстановление оставшихся выделить деньги из бюджета. При этом, так как средств в бюджете крайне не хватало, то было решено оставить как можно меньше железных и автомобильных дорог, лишь бы только по ним можно было добраться из любого города государства в любой другой (возможно через промежуточные города, используя несколько железных и/или автомобильных дорог), а именно, решено было оставить и отремонтировать ровно $n-1$ дорогу.

После изучения результатов работы комиссии правительство решило выделить деньги на ремонт a автомобильных и b железных дорог. Но перед ним встала непростая задача: какие же конкретно дороги следует закрыть, а какие — оставить и отремонтировать, чтобы все города государства остались связанными друг с другом?

После тщетных попыток придумать план ремонта дорог, было решено обратиться к вам за помощью.

Формат входного файла

В первой строке входного файла записано четыре целых числа: n $(1 \le n \le 100\,000)$ — количество городов Флатландии, m $(n-1 \le m \le 200\,000)$ — количество существующих автомобильных и железных дорог, a и b $(0 \le a, b;\ a+b = n-1)$ — количество автомобильных и железных дорог, которые необходимо оставить. Следующие m строк описывают имеющиеся дороги. Каждая из них содержит по три целых числа: u_i, v_i $(1 \le u_i, v_i \le n)$ — номера городов, которые соединяет i-я дорога, и t_i — тип дороги, который равен 0 для автомобильной дороги, и 1 для железной.

Гарантируется, что ни один город не связан дорогой сам с собой, и что каждая пара городов соединена максимум одной автомобильной и одной железной дорогой. Также гарантируется, что перед реформой из каждого города можно добраться до каждого.

Формат выходного файла

Если не существует плана ремонта дорог, удовлетворяющего описанным требованиям, то выведите в выходной файл единственное слово «`Impossible`». Иначе выведите через пробел $n-1$ число — номера дорог, которые необходимо отремонтировать (дороги нумеруются числами от 1 до m в порядке, в котором они заданы во входном файле). Если решений несколько, выведите любое.

Примеры

roads.in	roads.out
4 4 1 2 1 2 1 1 3 0 2 3 1 3 4 1	1 2 4
3 2 2 0 1 2 1 2 3 0	Impossible

Задача B. Призыв в армию

Имя входного файла:	`soldiery.in`
Имя выходного файла:	`soldiery.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта
Максимальная оценка за задачу:	120 баллов

В одном военкомате вновь проблема. Скоро очередной призыв, а список призываемых еще не составлен. Подполковник Призывайко хочет решить эту проблему раз и навсегда, но он нуждается в помощи.

Рассмотрим как проходит призыв. Всего на учете в этом военкомате состоят n человек, а призвать необходимо m человек. На каждого стоящего на учете человека заведено личное дело, к которому прикреплена справка о его состоянии здоровья. Состояние здоровья человека характеризуется тремя параметрами: силой, ростом и весом. Правда некоторые люди предоставляют в военкомат «липовые» справки, в которых указаны ложные параметры состояния здоровья.

Военкомат может провести обследование и исправить данные в справке на истинные данные (то есть внести в личное дело истинное значение некоторого параметра здоровья, вместо указанного ранее). За определение каждого из параметров здоровья призывников отвечает отдельный врач. Таким образом, в результате одного обследования значение ровно одного параметра здоровья ровно одного призывника заменяется в его справке на истинное значение.

В силу ограниченности времени в военкомате не могут провести более k обследований призывников.

Задача подполковника: призвать m человек так, чтобы сумма параметров, указанных в справках всех призванных, была максимальна. Ваша задача — написать программу, которая решит проблему призыва в военкомате.

Формат входного файла

В первой строке входного файла записаны числа n ($1 \le n \le 100$), m ($1 \le m \le n$), k ($0 \le k \le 300$). В следующих n строках указаны по 6 чисел. Первые три из них — сила, рост и вес в справке, следующие три — истинная сила, рост и вес. Все параметры здоровья неотрицательны и не превосходят 10^5.

Формат выходного файла

В первой строке выходного файла выведите максимальную возможную сумму параметров в справках всех призванных и количество z медицинских обследований, которые необходимо провести для того, чтобы получить такую сумму.

В каждой из последующих z строк выведите описание одного из обследований, которое требуется провести. Описание должно состоять из двух частей. Первая часть — номер призывника, которого необходимо обследовать. Вторая часть — это слово «`STRENGTH`», если обследование должно быть направлено на выяснение его истинной силы, «`HEIGHT`», если оно должно быть направлено на выявление его истинного роста, и «`WEIGHT`», если оно должно быть направлено на выявление его истинного веса.

В последней строке выходного файла выведите через пробел в возрастающем порядке m номеров призывников, которые подлежат призыву.

Призывники нумеруются натуральными числами от 1 до n в том порядке, в котором они заданы во входном файле.

Примеры

soldiery.in	soldiery.out
3 2 4 1 1 1 1 1 1 1 1 2 2 2 3 7 12 44 0 0 0	70 3 2 STRENGTH 2 HEIGHT 2 WEIGHT 2 3

Задача C. Троллейбусы

Имя входного файла:	`trolleys.in`
Имя выходного файла:	`trolleys.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта
Максимальная оценка за задачу:	120 баллов

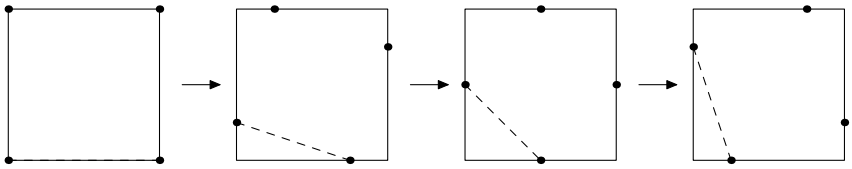
Гоша и Андрюша очень любят ездить в троллейбусах. Обычно Гоша садится в некоторый троллейбус, а Андрюша садится в следующий троллейбус, следующий по тому же маршруту, и они едут, постоянно пытаясь увидеть в окно друг друга. Чем ближе их троллейбусы оказываются друг к другу, тем больше ребята радуются.

Скоро в городе появится новый кольцевой маршрут троллейбуса. Гоша достал откуда-то схему маршрута и теперь просит вас сказать ему, насколько близко к Гошиному троллейбусу будет проезжать следующий троллейбус того же маршрута.

Новый маршрут имеет форму n-звенной замкнутой ломаной, возможно, с самопересечениями и самокасаниями. Троллейбусы ездят по нему с постоянной скоростью. Для простоты будем считать, что они вообще не останавливаются. Также для простоты будем считать троллейбусы точками.

По маршруту будут циркулировать m троллейбусов, в каждой точке маршрута они будут появляться через равные промежутки времени. Гоша будет ехать на одном из троллейбусов, а Андрюша — на следующем.

Например, если четыре троллейбуса будут ездить по квадратному маршруту, то в каждый момент времени они будут находиться в соответствующих точках на разных сторонах квадрата (кроме тех моментов, когда все четыре окажутся в вершинах).



В этом случае минимальное расстояние между соседними троллейбусами будет достигаться в момент, когда троллейбусы окажутся в серединах сторон квадрата.

Формат входного файла

Первая строка входного файла содержит два целых числа n и m — количество прямых отрезков в маршруте ($2 \le n \le 100\,000$) и количество троллейбусов на нем ($2 \le m \le 100\,000$).

Каждая из следующих n строк входного файла содержит два целых числа — координаты соответствующей вершины n-звенной ломаной. Координаты вершин приведены в порядке, в котором их будут проезжать троллейбусы. Координаты не превосходят $10\,000$ по абсолютной величине.

Формат выходного файла

В выходной файл выведите одно вещественное число — наименьшее расстояние между двумя соседними троллейбусами, которое достигается при движении по заданному маршруту. Выведите ответ не менее чем с пятью знаками после десятичной точки.

Примеры

trolleys.in	trolleys.out
4 4 0 0 0 1 1 1 1 0	0.707106781186547524