Задача 1.

Олимпиадная

**Минимальное остовное дерево - 2**

**Ограничение по времени работы программы: 1 секунда**

Дан связный неориентированный взвешенный граф без кратных рёбер с N вершинами и M рёбрами. Требуется найти вес его минимального каркаса (основного дерева).

ВХОДНЫЕ ДАННЫЕ

В первой строке входных данных записаны числа N (1⩽N⩽20000) и M (0⩽M⩽100000). В следующих M строках идёт описание рёбер графа. Каждое ребро задаётся тремя числами. Первые два числа — номера вершин, соединённых этим ребром, третье число — вес ребра. Все веса рёбер положительны и не превосходят 100000.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести одно целое число — вес минимального остовного дерева.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 4 4 1 2 1 2 3 2 3 4 5 4 1 4 | 7 |

Задача 2.

Сложная

**День объединения**

В Байтландии есть целых n городов, но нет ни одной дороги. Король решил исправить эту ситуацию и соединить некоторые  
города дорогами так, чтобы по этим дорогам можно было бы добраться от любого города до любого другого.  
Когда строительство будет завершено, Король планирует отпраздновать День Объединения. К сожалению, казна Байтландии  
почти пуста, поэтому Король требует сэкономить деньги, минимизировав суммарную длину всех построенных дорог.

ВХОДНЫЕ ДАННЫЕ

Первая строка входного файла содержит натуральное число n (1≤n≤5000)~-—  
количество городов в Байтландии. Каждая из следующих n строк содержит два целых числа  
xi, yi ~-— координаты i-го города (−10000≤xi,yi≤10000).  
Никакие два города не расположены в одной точке.

ВЫХОДНЫЕ ДАННЫЕ

Первая строка выходного файла должна содержать минимальную суммарную длину дорог. Выведите число с точностью не менее  
10−3.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 6 1 1  7 1 2 2 6 2 1 3 7 3 | 9.65685 |

Задача 3.

Сложная

**Острова**

**Ограничение по времени работы: 2 секунды**

Одно разбросанное на островах Океании государство решило создать сеть автомобильных дорог (вернее, мостов). По каждому мосту можно перемещаться в обе стороны. Был разработан план очередности строительства мостов и известно, что после постройки всех мостов можно будет проехать по ним с каждого острова на каждый (возможно, через некоторые промежуточные острова).

Однако, этот момент может наступить до того, как будут построены все мосты. Ваша задача состоит такое минимальное количество мостов, после постройки которых (в порядке строительства по плану) можно будет попасть с любого острова на любой другой.

ВХОДНЫЕ ДАННЫЕ

Первая строка входных данных содержит два числа: $N$ — число островов ($1 \leq N \leq 100\,000и)иM$ — количество мостов в плане ($1 \leq M \leq 200\,000Вкаждойследующейстрокесодержитсяописаниемостадвачисла).Вкаждойследующейстрокесодержитсяописаниемоста–двачислаx$ и $y$ ($0 \leq x, y \lt N$, то есть нумерация начинается с нуля) — номера соединяемых островов.

ВЫХОДНЫЕ ДАННЫЕ

Выведите в выходной файл одно число — минимальное количество построенных мостов, после которых можно будет попасть с любого острова на любой.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 4 5  0 1  0 2  1 2  2 3  3 0 | 4 |

Задача 4.

Лёгкая

**Разрезание графа**

Дан неориентированный граф. Над ним в заданном порядке производят операции следующих двух типов:

* cut — разрезать граф, то есть удалить из него ребро;
* ask — проверить, лежат ли две вершины графа в одной компоненте связности.

Известно, что после выполнения всех операций типа cut рёбер в графе не осталось. Найдите результат выполнения каждой из операций типа ask.

## **Входные данные**

Первая строка входного файла содержит три целых числа, разделённые пробелами — количество вершин графа n, количество рёбер m и количество операций k (1 ≤ n ≤ 50 000, 0 ≤ m ≤ 100 000, m ≤ k ≤ 150 000).

Следующие m строк задают рёбра графа; i-я из этих строк содержит два числа ui и vi (1 ≤ ui, vi ≤ n), разделённые пробелами — номера концов i-го ребра. Вершины нумеруются с единицы; граф не содержит петель и кратных рёбер.

Далее следуют k строк, описывающих операции. Операция типа cut задаётся строкой «cut u v» (1 ≤ u, v ≤ n), которая означает, что из графа удаляют ребро между вершинами u и v. Операция типа ask задаётся строкой «ask u v» (1 ≤ u, v ≤ n), которая означает, что необходимо узнать, лежат ли в данный момент вершины u и v в одной компоненте связности. Гарантируется, что каждое ребро графа встретится в операциях типа cut ровно один раз.

## **Выходные данные**

Для каждой операции ask во входном файле выведите на отдельной строке слово «YES», если две указанные вершины лежат в одной компоненте связности, и «NO» в противном случае. Порядок ответов должен соответствовать порядку операций ask во входном файле.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3 3 7  1 2  2 3  3 1  ask 3 3  cut 1 2  ask 1 2  cut 1 3  ask 2 1  cut 2 3  ask 3 1 | YES  YES  NO  NO |

Задача 5.

Средняя

**Школы**

С целью подготовки к проведению олимпиады по информатике мэр решил обеспечить надежным электроснабжением все школы города. Для этого необходимо провести линию электропередач от альтернативного источника электроэнергии “Майбуття” к одной из школ города (к какой неважно), а также соединить линиямии электропередач некоторые школы между собой.

Считается, что школа имеет надежное электроснабжение, если она напрямую связана с источником “Майбуття”, либо с одной из тех школ, которые имеют надежное электроснабжение.

Известна стоимость соединения между некоторыми парами школ. Мэр города решил выбрать одну из двух наиболее экономичных схем электроснабжения (стоимость схемы равняется сумме стоимостей соединений пар школ).

Напишите программу, которая вычисляет стоимость двух наиболее экономных схем альтернативного электроснабжения школ.

## **Входные данные**

В первой строке входного файла находятся два натуральных числа, разделенных пробелом:N (3 ≤ N ≤ 100), количество школ в городе, и M – количество возможных соединений между ними. В каждой из последующих Mстрок находятся по три числа: Ai, Bi, Ci, разделенных пробелами, где Ci – стоимость прокладки линии электроснабжения (1 ≤ Ci ≤ 300) от школы Ai до школы Bi (i=1,2,…,N).

## **Выходные данные**

В единственной строке выходного файла должны содержаться два натуральных числа S1 и S2, разделенных пробелом – две наименьшие стоимости схем (S1 ≤ S2). S1=S2 тогда и только тогда, когда существует несколько схем надежного электроснабжения наименьшей стоимости.

Гарантируется, что для входных данных существует две различные схемы надёжного электроснабжения.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 5 8  1 3 75  3 4 51  2 4 19  3 2 95  2 5 42  5 4 31  1 2 9  3 5 66 | 110 121 |

Задача 6.

Сложная

**Болото**

Иван-Царевич хочет спасти из плена Василису Прекрасную. По пути к темнице, где Кощей Бессмертный держит пленницу,  
есть болото с параллельными бесконечно длинными берегами ширины H.  
В болоте имеется N кочек, i~--я кочка имеет координаты xi, yi.  
Ось OX направлена параллельно берегу болота, а ось OY направлена перпендикулярно берегу  
болота от начального берега к конечному, точки начального берега имеют координату Y=0. Требуется определить, какой минимальной длиной прыжка должен обладать Иван~--Царевич,  
чтобы перебраться через болото.

### ВХОДНЫЕ ДАННЫЕ

Во входном файле в первой строке находятся числа H (1≤H≤30000) и N (1≤N≤100).  
В следующих N строках записаны координаты кочек xi, yi (1≤xi,yi≤30000).  
Число H и все координаты~-— целые числа.

### ВЫХОДНЫЕ ДАННЫЕ

В выходной файл нужно вывести единственное число~-— минимальную длину прыжка с точностью до 6 знаков после точки.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 10 3 1 3 3 7 6 6 | 4.472135955 |

Задача 7.

Олимпиадная

**Ребра добавляются, граф растет**

В неориентированный граф последовательно добавляются новые ребра. Изначально граф пустой.  
После каждого добавления нужно говорить, является ли текущий граф двудольным.

### ВХОДНЫЕ ДАННЫЕ

На первой строке n -— количество вершин, m -— количество операций <<добавить ребро>>.  
Следующие m строк содержат пары чисел от 1 до n -— описание добавляемых ребер. (1≤n, m≤300000)

### ВЫХОДНЫЕ ДАННЫЕ

Выведите в строчку m нулей и единиц. i-й символ  
должен быть равен единице, если граф, состоящий из первых i ребер, является двудольным.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3 3 1 2  2 3 3 1 | 110 |