



Molecule. Химия!!!

Имя входного файла: `molecule.in`
Имя выходного файла: `molecule.out`

Вася и Сережа играют в следующую игру. В некоторых клетках клетчатого листка Сережа рисует один из символов 'Н', 'О', 'N' или 'С', после чего Вася должен провести между некоторыми находящимися в соседних клетках символами линии так, чтобы получилось корректное изображение химической молекулы. К сожалению, Сережа любит рисовать много символов, и Вася не может сразу определить, возможно ли вообще нарисовать линии нужным способом. Помогите ему написать программу, которая даст ответ на этот вопрос.

В этой задаче проведенные между символами химических элементов линии будем считать корректным изображением молекулы, если они удовлетворяют следующим условиям:

- каждая линия соединяет символы, нарисованные в соседних (по стороне) клетках,
- между каждой парой символов проведено не более одной линии,
- от каждого элемента отходит ровно столько линий, какова валентность этого элемента (1 для Н, 2 для О, 3 для N, 4 для С),
- пустые клетки ни с чем не соединены, и
- хотя бы в одной клетке нарисован какой-то символ.

Формат входного файла

Первая строка входного файла содержит два натуральных числа n и m ($1 \leq n, m \leq 50$) — размеры листочка, на котором рисует Сережа. Далее следуют n строк по m символов в каждой, задающих конфигурацию химических элементов, которую нарисовал Сережа; пустые клетки задаются символом '.'.

Формат выходного файла

В выходной файл выведите одно слово «Valid», если линии провести требуемым образом можно, и «Invalid», если нельзя.

Пример

molecule.in	molecule.out
3 4 НОН. НСОН ОО..	Valid
3 4 НОН. НСОН ООНН	Invalid

Brides. В поисках невест

Имя входного файла: `brides.in`
Имя выходного файла: `brides.out`

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, 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

brides.in	brides.out
5 8 2 1 2 1 1 3 1 1 4 3 2 5 5 2 3 1 3 5 1 3 4 1 5 4 1	3.00000 3 1 5 6 3 2 7 8



LCA. Least Common Ancestor

Имя входного файла: `lca.in`
Имя выходного файла: `lca.out`

Изначально имеется дерево состоящее только из корня (вершина с номером 1). Требуется научиться отвечать на следующие запросы:

- ADD $a\ b$ — подвесить вершину b за вершину a (гарантируется, что вершина a уже существует).
- GET $a\ b$ — вернуть LCA вершин a и b .

Все номера вершин от 1 до N .

В каждый момент времени у нас есть одно дерево.

Формат входного файла

В первой строке входного файла содержится число k — количество запросов. Следующие k строк содержат сами запросы. Гарантируется, что число запросов каждого из типов не превосходит $5 \cdot 10^5$.

Формат выходного файла

Для каждого запроса типа GET выведите в отдельную строку одно целое число — ответ на соответствующий запрос.

Примеры

lca.in	lca.out
9	1
ADD 1 2	1
ADD 1 3	1
ADD 2 4	2
GET 1 3	5
GET 2 3	
GET 3 4	
ADD 2 5	
GET 4 5	
GET 5 5	

RMQ. Range Minimum Query

Имя входного файла: `rmq.in`
Имя выходного файла: `rmq.out`

Есть массив из N целых чисел и M запросов вида: найти минимум на отрезке с концами l_i, r_i .

Формат входного файла

Входной файл содержит T наборов тестовых данных. Каждый набор тестовых данных задаётся числами N, M, A, B ($1 \leq N \leq 25\,000, 1 \leq A, B \leq 10^9$), где N — размер массива, M — число запросов. Массив и запросы нужно получить следующим образом: выпишем последовательность чисел $1 \cdot A + B, 2 \cdot A + B, \dots, (N + 2 \cdot M) \cdot A + B$, взятых по модулю 2^{32} . Первые N чисел последовательности — элементы массива, числа с $N + 1$ по $N + 2 \cdot M$, взятые по модулю N образуют M пар чисел $(l_i - 1, r_i - 1)$ — запросы. Ввод заканчивается строкой 0000. Сумма N по всем наборам тестовых данных не превосходит 10^8 , сумма M по всем наборам тестовых данных не превосходит $2 \cdot 10^7$.

Формат выходного файла

Для каждого набора тестовых данных выведите сумму по всем запросам на отдельной строке.

Пример

rmq.in	rmq.out
10 10 955379886 619166003	7671393960
0 0 0 0	

Замечание

Массив:

1574545889 2529925775 3485305661 145718251 1101098137 2056478023 3011857909
3967237795 627650385 1583030271

Запросы:

8 4
4 10
6 2
8 8
4 10
6 6
2 8
4 10
10 6
2 8