

Содержание

Обязательные задачи	2
Задача А. 2-раскраска [1 sec, 256 mb]	2
Задача В. Коммивояжёр возвращается! [4 sec, 256 mb]	3
Задача С. Condense 2. Конденсация графа [1 sec, 256 mb]	4
Обычные задачи	5
Задача D. Мосты и компоненты [1 sec, 256 mb]	5
Задача Е. Хроматическое число [1 sec, 256 mb]	6
Задача F. Avia. Авиаперелеты [1 sec, 256 mb]	7
Дополнительные задачи	8
Задача G. Макс клика [1 sec, 512 mb]	8
Задача H. Длинная дорога [4 sec, 256 mb]	9

Вы не умеете читать/выводить данные, открывать файлы? Воспользуйтесь **примерами**.

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

Обратите внимание, что ввод-вывод во всех задачах стандартный.

Обязательные задачи

Задача А. 2-раскраска [1 сек, 256 mb]

Выясните, можно ли правильно раскрасить данный неориентированный граф в два цвета. Если можно, то приведите правильную раскраску.

Формат входных данных

В первой строке входного файла два целых числа N и M ($1 \leq N \leq 2 \cdot 10^4$, $0 \leq M \leq 2 \cdot 10^4$) — число вершин и рёбер в графе.

В следующих M строках заданы рёбра графа, каждое двумя целыми числами от 1 до N — номерами его концов. Гарантируется, что в графе нет петель и кратных рёбер.

Формат выходных данных

Если правильно раскрасить граф в два цвета невозможно, выведите “NO”.

Иначе выведите “YES” и N целых чисел — цвета вершин в порядке от 1-й до N -й. Цвет каждой вершины должен быть 1 или 2.

Примеры

stdin	stdout
4 3 1 2 3 2 1 4	YES 1 2 1 2
3 3 1 2 1 3 3 2	NO

Задача В. Коммивояжёр возвращается! [4 sec, 256 mb]

Коммивояжёр возвращается в систему Альфы Центавра! Население системы с нетерпением ждёт его прибытия — каждый хочет приобрести что-нибудь с далёких планет!

Как обычно, коммивояжёр хочет минимизировать транспортные расходы. Он выбирает начальную планету, прилетает туда на межгалактическом корабле, после чего посещает все остальные планеты системы в порядке, минимизирующем суммарную стоимость посещения, и на другом межгалактическом корабле улетает обратно. Естественно, коммивояжёр не хочет летать ни на какую планету дважды.

Найдите оптимальный маршрут для коммивояжёра. Массы больше не могут ждать!

Формат входных данных

В системе Альфы Центавра n планет. Это число записано в первой строке входного файла ($1 \leq n \leq 19$). Следующие n строк содержат по n чисел каждая: j -ое число на i -ой из этих строк — стоимость перемещения a_{ij} от i -ой планеты до j -ой. Числа в каждой строке разделены пробелами. Числа a_{ii} не несут полезной информации. Все числа во входном файле положительны и не превосходят 10^8 .

Формат выходных данных

В первой строке выходного файла выведите минимальную суммарную стоимость посещения всех планет. Во второй строке выведите n чисел через пробел — номера планет системы в порядке их посещения. Если оптимальных маршрутов несколько, можно вывести любой из них.

Пример

stdin	stdout
3	5
8 1 6	3 1 2
3 5 7	
4 9 2	

Задача C. Condense 2. Конденсация графа [1 сек, 256 mb]

Требуется найти количество ребер в конденсации ориентированного графа. Примечание: конденсация графа не содержит кратных ребер.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно ($n \leq 10\,000$, $m \leq 100\,000$). Следующие m строк содержат описание ребер, по одному на строке. Ребро номер i описывается двумя натуральными числами b_i , e_i — началом и концом ребра соответственно ($1 \leq b_i, e_i \leq n$). В графе могут присутствовать кратные ребра и петли.

Формат выходных данных

Первая строка выходного файла должна содержать одно число — количество ребер в конденсации графа.

Пример

stdin	stdout
4 4 2 1 3 2 2 3 4 3	2

Обычные задачи

Задача D. Мосты и компоненты [1 sec, 256 mb]

Дан неориентированный граф (не обязательно связный). Граф может содержать петли и кратные ребра.

Выведите все компоненты реберной двусвязности графа (максимальные подмножества вершин, такие что подграф на них не теряет связность при удалении любого ребра).

Формат входных данных

Первая строка содержит числа n и m ($1 \leq n \leq 100\,000$, $0 \leq m \leq 100\,000$) — количество вершин и ребер в графе.

Следующие m строк задают ребра графа.

Формат выходных данных

В первой строке выведите количество компонент, в следующих за ней строках выведите сами компоненты, по одной на строку.

Вершины в каждой компоненте должны идти в возрастающем порядке, компоненты нужно вывести в лексикографическом порядке.

Примеры

stdin	stdout
3 2 1 2 2 3	3 1 2 3
3 3 1 2 2 3 3 1	1 1 2 3
2 2 1 2 1 2	1 1 2
7 8 1 5 5 6 1 6 5 4 4 3 4 2 3 2 7 2	3 1 5 6 2 3 4 7

Задача Е. Хроматическое число [1 sec, 256 mb]

Хроматическое число графа — такое минимальное число k , что граф можно правильно раскрасить в k цветов. Найдите хроматическое число данного графа.

Формат входных данных

В первой строке входного файла два целых числа N и M ($1 \leq N \leq 13$, $0 \leq M \leq 78$) — число вершин и рёбер в графе.

В следующих M строках заданы рёбра графа, каждое двумя целыми числами от 1 до N — номерами его концов. Гарантируется, что в графе нет петель и кратных рёбер.

Формат выходных данных

Выведите искомое хроматическое число.

Примеры

stdin	stdout
3 3 1 2 2 3 3 1	3
2 0	1

Задача F. Avia. АвиAPERелеты [1 sec, 256 mb]

Главного конструктора Петю попросили разработать новую модель самолета для компании «Air Бубундия». Оказалось, что самая сложная часть заключается в подборе оптимального размера топливного бака.

Главный картограф «Air Бубундия» Вася составил подробную карту Бубундии. На этой карте он отметил расход топлива для перелета между каждой парой городов.

Петя хочет сделать размер бака минимально возможным, для которого самолет сможет долететь от любого города в любой другой (возможно, с дозаправками в пути).

Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 1000$) — число городов в Бубундии. Далее идут n строк по n чисел каждая. j -ое число в i -ой строке равно расходу топлива при перелете из i -ого города в j -ый. Все числа не меньше нуля и меньше 10^9 . Гарантируется, что для любого i в i -ой строчке i -ое число равно нулю.

Формат выходных данных

Первая строка выходного файла должна содержать одно число — оптимальный размер бака.

Пример

stdin	stdout
4 0 10 12 16 11 0 8 9 10 13 0 22 13 10 17 0	10

Дополнительные задачи

Задача G. Макс клика [1 sec, 512 mb]

Макс, опытный игрок в доту, постоянно кликал.

Дан случайный неориентированный граф G из n вершин и m ребер. Подкликой называется такое подмножество вершин A : $\forall a, b \in A, a \neq b \quad \exists$ ребро (a, b) . Ваша задача — найти подклику A : $|A|$ максимально.

Формат входных данных

На первой строке число вершин $1 \leq n \leq 40$ и число ребер $m \geq 1$.
Следующие m строк содержат пары чисел от 1 до n — ребра графа.
В графе нет ни петель, ни кратных ребер.

Формат выходных данных

На первой строке выведите k — количество вершин в максимальной подклике. На следующей строке k целых чисел от 1 до n — номера вершин в подклике. Вершины можно выводить в любом порядке. Если максимальных подклик несколько, выведите любую.

Примеры

stdin	stdout
5 8 5 4 3 5 1 5 1 3 2 3 1 4 5 2 3 4	4 1 3 4 5

Задача Н. Длинная дорога [4 sec, 256 mb]

Дорога, дорога, осталось немного...

Дан случайный неориентированный граф G из n вершин и m ребер. Ваша задача — найти гамильтонов путь. Гарантируется, что гамильтонов путь в графе есть.

Формат входных данных

На первой строке число вершин $n \leq 50$ и число ребер $m \geq 1$.

Следующие m строк содержат пары чисел от 1 до n — ребра графа.

В графе нет ни петель, ни кратных ребер.

Поскольку почти полный граф — совсем не интересный тест, $m \leq 500$.

Формат выходных данных

На первой строке выведите n различных чисел от 1 до n — вершины гамильтоного пути в порядке прохода по ним. Начинать и заканчивать можно в любой вершине. Если гамильтоновых путей несколько, выведите любой.

Примеры

stdin	stdout
5 8 3 1 2 5 5 4 3 4 1 4 3 5 3 2 1 2	1 4 3 5 2