

Содержание

Графы	2
1 Задача А. Связанность графа	2
2 Задача В. Расстояние от корня	3
3 Задача С. Получи дерево	4
Интересные задачки	5
4 Задача D. Площадь комнаты	5
5 Задача Е. Поиск	6
6 Задача F. Отрезки	7

Графы

1 Задача А. Связанность графа

Имя входного файла: `edges.in`
Имя выходного файла: `edges.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дан граф, содержащий N вершин и M рёбер ($1 \leq N \leq 1000, 1 \leq M \leq 7000$). Требуется найти наименьшее число рёбер и эти рёбра, которые нужно добавить, чтобы граф стал связным.

Формат входных данных

Во входном файле записаны сначала числа N и M , затем идёт описание рёбер графа — M пар чисел, где каждая пара описывает начало и конец ребра.

Формат выходных данных

В первую строку вывести единственное число K — минимальное количество рёбер, которое нужно добавить. В следующих K строках выведите по 2 числа — начало и конец нового ребра.

<code>edges.in</code>	<code>edges.out</code>
3 1	1
2 1	1 3

2 Задача В. Расстояние от корня

Имя входного файла: `rootdist.in`
Имя выходного файла: `rootdist.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В заданном корневом дереве найдите вершины, максимально удалённые от корня. Расстоянием между вершинами считается количество рёбер в пути.

Формат входных данных

В первой строке задано n — количество вершин в дереве ($1 \leq n \leq 100$). В следующих $n - 1$ строках заданы вершины, являющиеся предками вершин $2, 3, \dots, n$. Вершина 1 является корнем дерева.

Формат выходных данных

В первой строке выведите максимальное расстояние от корня до остальных вершин дерева. Во второй строке выведите, сколько вершин дерева находятся от корня на таком расстоянии. В третьей строке выведите номера этих вершин через пробел в порядке возрастания.

Примеры

<code>rootdist.in</code>	<code>rootdist.out</code>
3	1
1	2
1	2 3
3	2
1	1
2	3

3 Задача С. Получи дерево

Имя входного файла: `graph2tree.in`
Имя выходного файла: `graph2tree.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 16 мегабайт

Дан связный неориентированный граф без петель и кратных ребер. Разрешается удалять из него ребра. Требуется получить дерево.

Формат входных данных

Сначала вводятся два числа: N (от 1 до 100) и M — количество вершин и ребер графа соответственно. Далее идет M пар чисел, задающих ребра. Гарантируется, что граф связный.

Формат выходных данных

Выведите $N - 1$ пар чисел — ребра, которые войдут в дерево. Ребра можно выводить в любом порядке.

Пример

<code>graph2tree.in</code>	<code>graph2tree.out</code>
4 4	1 2
1 2	2 3
2 3	3 4
3 4	
4 1	

Интересные задачи

4 Задача D. Площадь комнаты

Имя входного файла: `roomsquare.in`
Имя выходного файла: `roomsquare.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 16 мегабайт

Требуется вычислить площадь комнаты в квадратном лабиринте.

Формат входных данных

В первой строке вводится число N — размер лабиринта ($3 \leq N \leq 10$). В следующих N строках задан лабиринт ('.' — пустая клетка, '*' — стенка). И наконец, последняя строка содержит два числа — номер строки и столбца клетки, находящейся в комнате, площадь которой необходимо вычислить. Гарантируется, что эта клетка пустая и что лабиринт окружен стенками со всех сторон.

Формат выходных данных

Требуется вывести единственное число — количество пустых клеток в данной комнате.

Пример

<code>roomsquare.in</code>	<code>roomsquare.out</code>
5 ***** **.* *.*.* *.*.* ***** 2 4	3

5 Задача Е. Поиск

Имя входного файла: `find2.in`
Имя выходного файла: `find2.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В этой задаче нужно уметь выяснять, содержится ли число в последовательности.

Формат входных данных

В первой строке входного файла заданы через пробел два целых числа n и k ($1 \leq n \leq 300\,000$, $1 \leq k \leq 300\,000$). Во второй строке задана последовательность из n отсортированных целых чисел a_1, a_2, \dots, a_n , записанных через пробел ($1 \leq a_i \leq 10^9$). В третьей строке записаны запросы — k целых чисел b_1, b_2, \dots, b_k записанных через пробел, в порядке возрастания ($1 \leq b_j \leq 10^9$).

Формат выходных данных

В выходной файл выведите k строк. В j -ой строке выведите “YES”, если число b_j содержится в последовательности $\{a_i\}$, и “NO” в противном случае.

Примеры

find2.in	find2.out
3 3	NO
2 3 5	YES
1 2 3	YES
3 4	YES
1 2 2	YES
1 2 4 5	NO
	NO

6 Задача F. Отрезки

Имя входного файла: `segments.in`
Имя выходного файла: `segments.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 Мебибайт

Даны отрезки на прямой. Какое максимальное количество отрезков можно выбрать так, чтобы никакие два из них не пересекались? Отрезки считаются открытыми.

Формат входных данных

В первой строке входного файла задано целое число n ($1 \leq n \leq 100\,000$). В следующих n строках описаны отрезки; i -я из этих строк содержит два целых числа l_i и r_i через пробел — координаты начала и конца отрезка ($1 \leq l_i < r_i \leq 10^9$).

Формат выходных данных

В первой строке выходного файла выведите одно число — максимальное количество выбранных отрезков.

Примеры

<code>segments.in</code>	<code>segments.out</code>
2 1 2 3 5	2
2 1 3 2 6	1
3 2 4 3 5 1 3	2