

Задача А. Сокровища

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

Дочь короля Флатландии собирается выйти за прекрасного принца. Принц хочет подарить принцессе сокровища, но он не уверен какие именно бриллианты из своей коллекции выбрать.

В коллекции принца n бриллиантов, каждый характеризуется весом w_i и стоимостью v_i . Принц хочет подарить наиболее дорогие бриллианты, однако король умен и не примет бриллиантов суммарного веса больше R . С другой стороны, принц будет считать себя жадным всю оставшуюся жизнь, если подарит бриллиантов суммарным весом меньше L .

Помогите принцу выбрать набор бриллиантов наибольшей суммарной стоимости, чтобы суммарный вес был в отрезке $[L, R]$.

Формат входных данных

Первая строка содержит число n ($1 \leq n \leq 32$), L и R ($0 \leq L \leq R \leq 10^{18}$). Следующие n строк описывают бриллианты и содержат по два числа — вес и стоимость соответствующего бриллианта ($1 \leq w_i, v_i \leq 10^{15}$).

Формат выходных данных

Первая строка вывода должна содержать k — количество бриллиантов, которые нужно подарить королю. Вторая строка должна содержать номера даримых бриллиантов.

Бриллианты нумеруются от 1 до n в порядке появления во входных данных.

Если составить подарок королю невозможно, то выведите 0 в первой строке вывода.

Пример

dowry.in	dowry.out
3 6 8	1
3 10	2
7 3	
8 2	

Задача В. Симпатичные узоры 2

Имя входного файла: `nice2.in`
Имя выходного файла: `nice2.out`
Ограничение по времени: 2 секунда
Ограничение по памяти: 16 мегабайт

Компания *BrokenTiles* планирует заняться выкладыванием во дворах у состоятельных клиентов узор из черных и белых плиток, каждая из которых имеет размер 1×1 метр. Известно, что дворы всех состоятельных людей имеют наиболее модную на сегодня форму прямоугольника $n \times m$ метров.

Однако при составлении финансового плана у директора этой организации появилось целых две серьезных проблемы: во первых, каждый новый клиент очевидно захочет, чтобы узор, выложенный у него во дворе, отличался от узоров всех остальных клиентов этой фирмы, а во вторых, этот узор должен быть симпатичным.

Как показало исследование, узор является симпатичным, если в нем нигде не встречается квадрата 2×2 метра, полностью покрытого плитками одного цвета.

Для составления финансового плана директору необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

Формат входных данных

На первой строке входного файла находятся два натуральных числа n и m . $1 \leq n \cdot m \leq 300$.

Формат выходных данных

Выведите в выходной файл единственное число — количество различных симпатичных узоров, которые можно выложить во дворе размера $n \times m$ по модулю $2^{30} + 1$. Узоры, получающиеся друг из друга сдвигом, поворотом или отражением считаются различными.

Пример

nice2.in	nice2.out
2 2	14
3 3	322

Задача С. Самый длинный путь

Имя входного файла: path.in
Имя выходного файла: path.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В данном ориентированном графе найдите самый длинный путь такой, что каждая вершина графа встречается в нём не более одного раза.

Формат входных данных

В первой строке входного файла заданы через пробел два целых числа n и m ($1 \leq n \leq 22$, $0 \leq m \leq 1000$). В следующих m строках заданы рёбра графа в формате $u_i v_i$ — номера начальной и конечной вершин ребра i , соответственно. Граф может содержать петли и кратные рёбра.

Формат выходных данных

В первой строке выходного файла выведите длину искомого пути l . Во второй строке выведите $l + 1$ число через пробел — вершины пути в порядке обхода. Если оптимальных ответов несколько, можно вывести любой из них.

Примеры

path.in	path.out
3 3 1 2 2 3 3 1	2 1 2 3
4 6 1 2 2 1 2 3 2 4 3 2 4 2	2 1 2 4
5 3 3 2 2 2 1 5	1 3 2

Задача D. Выбор вершин взвешенного дерева

Имя входного файла: selectw.in
Имя выходного файла: selectw.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 Мегабайт

Дан граф, являющийся деревом. В вершинах графа написаны целые числа. Множество вершин графа называется *допустимым*, если никакие две вершины этого множества не соединены ребром.

Рассмотрим все допустимые множества вершин графа. Для каждого такого множества вычислим сумму чисел, написанных в его вершинах. Какова максимальная из этих сумм?

Формат входных данных

Граф в этой задаче задан в виде *корневого дерева*. В графе выделена вершина — *корень дерева*. Для каждой вершины i , не являющейся корнем, задан номер вершины-предка p_i в корневом дереве. Дерево, заданное таким образом, состоит из рёбер $i - p_i$ для всех вершин i , кроме корня.

В первой строке входного файла записано целое число n — количество вершин в графе ($1 \leq n \leq 100$). В следующих n строках задан граф. В i -й из этих строк записаны через пробел два целых числа p_i и q_i ; здесь p_i — номер вершины-предка i -ой вершины, а q_i — число, записанное в этой вершине. Для корня дерева $p_i = 0$; для всех остальных вершин $1 \leq p_i \leq n$. Числа q_i не превосходят по модулю 10 000.

Гарантируется, что заданный во входном файле граф является деревом.

Формат выходных данных

В первой строке выходного файла выведите одно число — максимальную сумму чисел в допустимом множестве.

Примеры

selectw.in	selectw.out
5 0 1 1 2 1 3 2 4 3 5	10
6 5 8 6 0 5 -1 1 1 0 3 1 2	8

На рисунке показаны графы, заданные в примерах. В каждом графе выделено допустимое множество с максимальной суммой чисел в вершинах.

