

Задача А. Продавец аквариумов

Имя входного файла: aquarium.in
 Имя выходного файла: aquarium.out
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Продавец аквариумов для кошек хочет объехать n городов, посетив каждый из них ровно один раз. Помогите ему найти кратчайший путь.



Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 13$) — количество городов. Следующие n строк содержат по n чисел — длины путей между городами.

В i -й строке j -е число — $a_{i,j}$ — это расстояние между городами i и j ($0 \leq a_{i,j} \leq 10^6$; $a_{i,j} = a_{j,i}$; $a_{i,i} = 0$).

Формат выходных данных

В первой строке выходного файла выведите длину кратчайшего пути. Во второй строке выведите n чисел — порядок, в котором нужно посетить города.

Пример

aquarium.in	aquarium.out
5	666
0 183 163 173 181	1 3 2 5 4
183 0 165 172 171	
163 165 0 189 302	
173 172 189 0 167	
181 171 302 167 0	

Задача В. Логическое дерево

Имя входного файла:	boolean.in
Имя выходного файла:	boolean.out
Ограничение по времени:	0.5 секунда
Ограничение по памяти:	256 мегабайт

Рассмотрим разновидность двоичного дерева, которую мы назовем логическим деревом. В этом дереве каждый уровень полностью заполнен, за исключением, возможно, последнего (самого глубокого) уровня. При этом все вершины последнего уровня находятся максимально слева. Дополнительно, каждая вершина дерева имеет ноль или двоих детей.

Каждая вершина дерева имеет связанное с ней логическое значение (1 или 0). Кроме этого, каждая внутренняя вершина имеет связанную с ней логическую операцию („И“ или „ИЛИ“). Значение вершины с операцией „И“ — это логическое „И“ значений её детей. Аналогично, значение вершины с операцией „ИЛИ“ — это логическое „ИЛИ“ значений её детей. Значения всех листьев задаются во входном файле, поэтому значения всех вершин дерева могут быть найдены.

Наибольший интерес для нас представляет корень дерева. Мы хотим, чтобы он имел заданное логическое значение v , которое может отличаться от текущего. К счастью, мы можем изменять логические операции некоторых внутренних вершин (заменить „И“ на „ИЛИ“ и наоборот).

Дано описание логического дерева и набор вершин, операции в которых могут быть изменены. Найдите наименьшее количество вершин, которые следует изменить, чтобы корень дерева принял заданное значение v . Если это невозможно, то выведите строку «IMPOSSIBLE» (без кавычек).

Формат входных данных

В первой строке входного файла находятся два числа n и v ($1 \leq n \leq 10\,000$, $0 \leq v \leq 1$) — количество вершин в дереве и требуемое значение в корне соответственно. Поскольку все вершины имеют ноль или двоих детей, то n нечётно. Следующие n строк описывают вершины дерева. Вершины нумеруются от 1 до n .

Первые $(n - 1)/2$ строк описывают внутренние вершины. Каждая из них содержит два числа — g и c , которые принимают значение либо 0, либо 1. Если $g = 1$, то вершина представляет логическую операцию „И“, иначе она представляет логическую операцию „ИЛИ“. Если $c = 1$, то операция в вершине может быть изменена, иначе нет. Внутренняя вершина с номером i имеет детей $2i$ и $2i + 1$.

Следующие $(n + 1)/2$ строк описывают листья. Каждая строка содержит одно число 0 или 1 — значение листа.

Формат выходных данных

В выходной файл выведите ответ на задачу.

Примеры

boolean.in	boolean.out
9 1 1 0 1 1 1 1 0 0 1 0 1 0 1	1
5 0 1 1 0 0 1 1 0	IMPOSSIBLE

Задача С. Передай другу

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

ЛКШ 2011 будет проходить в Рио де Жанейро, где все ходят с белыми бейджиками. Вы не знали этого? А ведь именно такого рода слухи распространяет среди своих друзей лкшонок Остап.

У Остапа есть несколько разных слухов. Для каждого типа слуха он выбирает друзей, которым хочет его рассказать напрямую. Сообщая слух другу, Остап также просит его передать этот слух всем его друзьям. Однако те уже не будут передавать этот слух далее. Например, если Аня и Ваня — друзья, Ваня и Саня — друзья, а Аня и Саня — не друзья, и Остап сообщает какой-то слух Ане, то новость дойдет до Вани, но не до Сани.

Каждому человеку Остап может сообщить напрямую не более одного слуха (при этом если человек получил несколько разных типов слухов через друзей, ничего страшного нет). Какое наибольшее число различных типов слухов Остап может распространить так, чтобы каждый из них дошел до каждого его друга?

Формат входных данных

Во входном файле находится квадратная таблица $n \times n$, где $n \leq 15$ — число друзей Остапа. В i -й строке в j -м столбце содержится символ `Y`, если i -й и j -й друзья Остапа дружат между собой, и `N` — иначе.

Формат выходных данных

Выведите наибольшее количество слухов, которые может распространить Остап среди всех своих друзей.

Примеры

<code>friends.in</code>	<code>friends.out</code>
<pre> NYYN YNYN YNYN NYYN </pre>	3
<pre> NYYN YNYN YYNN NNNN </pre>	1
<pre> NYNNNY YNYNNN NYNYNN NNYNYN NNNYNY YNNNYN </pre>	3

Задача D. Сеть

Имя входного файла: `network.in`
 Имя выходного файла: `network.out`
 Ограничение по времени: 1.3 секунды
 Ограничение по памяти: 256 мегабайт

В компьютерной сети вашей фирмы n компьютеров. В последнее время свитч, к которому они подключены, сильно барахлит, и потому не любые два компьютера могут связаться друг с другом. Кроме того, если компьютер a обменивается информацией с компьютером b , то никакие другие компьютеры не могут в это время обмениваться информацией ни с a , ни с b . Вам необходимо вычислить максимальное количество компьютеров, которые могут одновременно участвовать в процессе обмена информацией.

Формат входных данных

В первой строке файла задано число n ($1 \leq n \leq 18$). Далее идут n строк по n символов, причём j -й символ i -й строки равен 'Y', если i -й и j -й компьютеры могут обмениваться информацией, иначе он равен 'N'. Верно, что i -й символ i -й строки всегда равен 'N' и, кроме того, матрица символов симметрична.

Формат выходных данных

Выведите максимальное количество компьютеров, которые могут одновременно участвовать в процессе обмена информацией.

Пример

<code>network.in</code>	<code>network.out</code>
5 NYYYY YNNNN YNNNY YNNNY YNYYN	4

Задача Е. Самый длинный путь 22

Имя входного файла: `path.in`
 Имя выходного файла: `path.out`
 Ограничение по времени: 0.75 секунд
 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 3
5 3 3 2 2 2 1 5	1 3 2

Задача F. Реки

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

В одном уголке земного шара раскинулась живописная страна, покрытая лесами и реками. Маленькие речушки сливаются в большие, те — в ещё большие, и в конце концов все реки сливаются в одну большую реку, впадающую в море неподалёку от городка.

А ещё в этой стране есть n поселений лесорубов. Каждое поселение расположено около какой-то реки. Лесорубы валят лес, а затем сплавляют его по воде до городка. В городке находится лесопилка.

Чтобы сократить расходы на транспортировку леса по воде, решено было выбрать k поселений лесорубов и построить дополнительные лесопилки прямо у этих поселений. Из остальных поселений лес будет сплавляться вниз по течению до первой же лесопилки, будь то старая лесопилка в городке или одна из k новых. Лес, обработанный в лесопилке, сплавлять дальше по воде не требуется.

Ни одна река не разветвляется, так что из любого поселения существует единственный путь вниз по течению до городка.

Для каждого поселения лесорубов известно, сколько деревьев валят жители этого поселения за год. Требуется поставить новые лесопилки так, чтобы ежегодные расходы на сплав леса оказались как можно меньше. Известно, что транспортировка одного дерева на один километр по любому отрезку речной сети обходится в одну монету. Напишите программу, которая по заданному описанию речной сети, а также числу k вычисляет, каковы будут минимальные ежегодные расходы на сплав леса в монетах.

Формат входных данных

В первой строке ввода содержится два целых числа n и k — количество поселений лесорубов и количество новых лесопилок, которые можно построить ($2 \leq n \leq 100, k \leq 50, k \leq n$). Следующие n строк описывают деревни; i -я из этих строк содержит три целых числа w_i, v_i и d_i — количество деревьев, ежегодно срубаемых жителями i -го поселения, номер ближайшего населённого пункта вниз по течению и расстояние до него ($0 \leq w_i \leq 10000, 0 \leq v_i \leq n, 1 \leq d_i \leq 10000$). Поселения пронумерованы целыми числами, начиная с единицы, в том порядке, в котором они заданы во входных данных, городок обозначается числом 0.

Гарантируется, что до постройки дополнительных лесопилок ежегодные расходы на сплав составляли не более $2 \cdot 10^9$ монет.

Формат выходных данных

В первой строке выведите одно число — минимальные ежегодные расходы на сплав леса в монетах.

Примеры

<code>rivers.in</code>	<code>rivers.out</code>
4 2 1 0 1 1 1 10 10 2 5 1 2 3	4