

Задача А. Размеры поддеревьев

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам дано дерево в корне с вершиной 1.

Поддеревом вершины v называется такое наибольшее множество вершин, в котором путь от каждой вершины до корня содержит вершину v .

Для каждой вершины выведите размер ее поддерева.

Формат входных данных

В первой строке вам дано число n - размер дерева ($2 \leq n \leq 10^6$).

Затем в следующей строке следует описание массива p размера $n - 1$. В нем p_i (где $1 \leq i \leq n - 1$) означает, что в дереве есть ребро между вершинами $i + 1$ и p_i ($1 \leq p_i \leq i$).

Формат выходных данных

Для каждой вершины выведите размер ее поддерева.

Примеры

стандартный ввод	стандартный вывод
6 1 1 2 2 3	6 3 2 1 1 1

Задача В. Куча камней

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

У вас есть несколько камней известного веса $w_1, \dots, w_n$. Напишите программу, которая распределит камни в две кучи так, что разность весов этих двух куч будет минимальной.

Формат входных данных

Ввод содержит количество камней n ($1 \leq n \leq 40$) и веса камней $w_1, \dots, w_n$ ($1 \leq w_i \leq 10^9$) — целые, разделённые пробельными символами.

Формат выходных данных

Ваша программа должна вывести одно число — минимальную разность весов двух куч.

Примеры

стандартный ввод	стандартный вывод
5 46 25 62 11 45	7

Задача С. Сокровища

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

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 содержит 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 принцессе. Вторая строка должна содержать номера даримых бриллиантов.

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

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

Примеры

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

Задача D. Разносчик пиццы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Вы подрабатываете разноской пиццы. У вас есть рюкзак размера S и огромный заказ на n пицц, i -я из которых имеет размер a_i . Разумеется, доставить пиццу требуется как можно скорее. К сожалению, у вас нет ни машины, ни друзей, которые могли бы помочь, так что единственный способ перевозки — распределить все пиццы в стопки размера не более S каждая и доставлять стопки по очереди. Вам надо распределить все пиццы из заказа в минимально возможное количество стопок.

Формат входных данных

Входной файл состоит из t тестов ($1 \leq t \leq 10$). Первая строка файла содержит число t , далее следуют описания тестов. Каждый тест описывается двумя строчками: на первой располагаются целые числа n ($1 \leq n \leq 20$) и S ($1 \leq S \leq 10^9$), на второй располагаются целые числа $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq S$).

Формат выходных данных

Для каждого теста выведите на отдельной строке минимальное число стопок m , а на следующих m строчках — описание стопок. i -я из последующих строк должна содержать количество пицц в i -й стопке k_i и список из k_i номеров пицц. Каждая пицца должна встречаться ровно в одной стопке. Если есть несколько оптимальных решений, выведите любое из них.

Примеры

стандартный ввод	стандартный вывод
3	1
1 10	1 1
10	2
2 10	1 1
10 10	1 2
4 10	3
5 7 5 7	1 2
	2 1 3
	1 4

Задача Е. Деловые встречи

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

Алексей — успешный предприниматель, и в течение одного дня у него бывает много встреч с разными деловыми партнёрами. К сожалению, встречи бывают разные и не все приносят ему радость, после других же настроение улучшается. Также, на многие встречи не стоит приходить в слишком плохом или хорошем настроении — результат таких встреч может быть не таким, какой хочется Алексею.

К счастью, недавно Алексей научился оценивать своё настроение с помощью целых чисел. После этого для каждой встречи он оценил, при каком максимальном и минимальном настроении стоит на неё приходить, а также как изменится его настроение после этой встречи. Теперь он хочет распланировать порядок встреч так, чтобы в течение дня совершить максимальное число встреч.

Ваша задача — написать программу, которая по информации о всех встречах и настроении Алексея в начале дня находит порядок проведения встреч такой, что их количество при этом максимально.

Формат входных данных

Первая строка входного файла содержит два целых числа n и k ($1 \leq n \leq 20$, $-100 \leq k \leq 100$) — количество встреч и настроение Алексея в начале дня.

Следующие n строк содержат по три целых числа a_i , b_i и c_i ($-100 \leq a_i, b_i, c_i \leq 100$) — минимальное и максимальное настроение, при котором встреча возможна, и изменение настроения по окончании встречи, соответственно.

Формат выходных данных

В первой строке выходного файла выведите число m — максимально возможно число встреч. В следующей строке выведите m целых чисел — номера встреч в порядке их проведения. Встречи пронумерованы в порядке описания во входном файле.

Если ответов с максимальным числом встреч несколько, выведите любой.

Примеры

meetings.in	meetings.out
3 0 1 3 3 0 1 2 1 3 1	3 2 3 1
3 1 -10 -5 3 -5 5 -2 -3 2 1	2 2 3

Задача F. 17 стульев

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

Остап Бендер снова пытается получить причитающиеся драгоценности, но на этот раз они были заперты в шкатулке, для открытия которой необходимо иметь N ключей. По закономерной случайности каждый из ключей был спрятан в одном из N стульев, распроданных на недавнем аукционе. После аукциона эти стулья были развезены в N городов.

И вот теперь Остап решил на новую безумную затею: заехать в каждый из городов и, провернув в каждом из них аферу, выкрасть необходимые ключи. Чтобы избежать конфликтов с недоброжелателями, Остап не хочет больше одного раза появляться в каком-либо городе. Также у Остапа есть список цен за проезд между каждой парой городов. Изначально Остап находится в городе под номером 1 и после посещения всех городов может незаметно скрыться из этой страны.

Помогите Остапу найти порядок посещения городов, при котором ему потребуется потратить как можно меньше средств на странствия, и тогда, возможно, он поделится с Вами добытыми бриллиантами.

Формат входных данных

Первая строка содержит единственное число N — количество городов ($1 \leq N \leq 17$).

Следующие N строк содержат по N целых неотрицательных чисел. j -тое число в i -той строке означает стоимость проезда из города i в город j ($0 \leq a_{ij} \leq 100$). Если $a_{ij} > 0$, то проезд стоит a_{ij} рублей, иначе — это означает, что из города i в j невозможно проехать напрямую.

Формат выходных данных

В первой строке выведите минимальную сумму денег, необходимую для посещения всех городов Остапом. В следующей строке выведите N чисел — порядок посещения городов, при котором эта сумма достигается. Если затею Остапа невозможно вывести, то в единственной строке выходного файла выведите число -1.

Примеры

<code>trader.in</code>	<code>trader.out</code>
3 0 3 2 3 0 6 2 6 0	8 1 3 2
5 0 6 4 0 0 6 0 7 0 7 4 7 0 0 0 0 0 0 0 2 0 7 0 2 0	20 1 3 2 5 4

Задача G. Раскраска графа

Имя входного файла: `coloring.in`
Имя выходного файла: `coloring.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Дан граф из n вершин, раскрасьте его в минимально возможное число цветов так, чтобы никакие две вершины, соединенные ребром, не были одного цвета.

Формат входных данных

В первой строке содержится число t — количество тестовых примеров ($1 \leq t \leq 5$).

Далее содержится t тестовых случаев, заданных в следующем формате:

В первой строке записаны числа n и m — количество вершин и ребер соответственно ($1 \leq n \leq 17$, $0 \leq m \leq \frac{n \cdot (n-1)}{2}$).

Затем идет m строк, в которых содержится по два числа v_i u_i , что означает, что вершины v_i и u_i соединены ребром ($1 \leq v_i, u_i \leq n, v_i \neq u_i$).

Гарантируется, что все ребра в каждом тестовом случае различны.

Формат выходных данных

Для каждого тестового случая в первой строке выведите минимальное число цветов k .

Во второй строке выведите n чисел a_i — цвета вершин ($1 \leq a_i \leq k$).

Примеры

coloring.in	coloring.out
3	3
3 3	3 2 1
1 2	2
2 3	1 2 2 1 1
3 1	3
5 3	1 3 1 1 2 1
2 1	
3 1	
4 2	
6 7	
1 2	
1 5	
2 5	
2 3	
2 4	
5 6	
5 4	

Задача Н. 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

network.in	network.out
5 NYYYY YNNNN YNNNY YNNNY YNYYN	4

Задача I. Пастбища

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

Фермер Джон решил снабдить каждую из своих коров сотовым телефоном. Для этого ему требуется установить сотовые станции на его N пастбищах, последовательно пронумерованных от 1 до N .

Ровно $N - 1$ пара пастбищ являются соседними, и для любых двух пастбищ A и B имеется последовательность соседних пастбищ таких, что A — первое пастбище этой последовательности, а B — последнее. Сотовые станции размещаются только на пастбищах и имеют достаточный радиус действия, чтобы обеспечить связью это пастбище и все соседние.

Помогите фермеру Джону определить, какое минимальное количество станций он должен установить, чтобы обеспечить связью все пастбища.

Формат входных данных

На первой строке входного файла находится одно целое число N ($1 \leq N \leq 100\,000$). Далее следуют $N - 1$ строк, каждая из которых содержит пару целых чисел — очередную пару соседних пастбищ A и B ($1 \leq A \leq N$; $1 \leq B \leq N$; $A \neq B$).

Формат выходных данных

Выведите в выходной файл одно число — минимальное достаточное количество станций.

Примеры

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