

Задача А. 17 стульев

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

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 -й 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

стандартный ввод	стандартный вывод
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 В. Деловые встречи

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
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

стандартный ввод	стандартный вывод
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 3 2

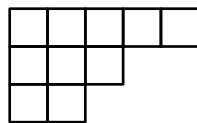
Задача С. Увидеть Юнга и умереть

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

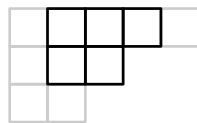
Диаграммы Юнга используются для того, чтобы изобразить разбиение числа на слагаемые. Разбиение числа n на слагаемые представляет собой сумму вида $n = m_1 + m_2 + \dots + m_k$, где $m_1 \geq m_2 \geq \dots \geq m_k$.

Диаграмма состоит из n квадратиков, организованных в виде k рядов, где k количество слагаемых в разбиении. Ряд, соответствующий числу m_i , содержит m_i квадратиков. Все ряды выровнены по левому краю и упорядочены от более длинного к более короткому.

Например, диаграмма Юнга, приведенная на рисунке, соответствует разбиению $10 = 5 + 3 + 2$.



Иногда можно вписать одну диаграмму Юнга в другую. Диаграмму X можно вписать в диаграмму Y , если можно удалить некоторые квадратики из диаграммы Y так, чтобы получилась диаграмма X . Отметим, что разрешается только удалять некоторые квадратики, вращать или отражать диаграмму не разрешается. Например, диаграмма для разбиения $5 = 3 + 2$ может быть вписана в диаграмму для разбиения $10 = 5 + 3 + 2$, как показано на рисунке.



С другой стороны, диаграмму для разбиения $8 = 4 + 4$ нельзя вписать в диаграмму для разбиения $10 = 5 + 3 + 2$.

Для заданного n найдите такое разбиение n на слагаемые, что в соответствующую ему диаграмму Юнга можно вписать максимальное количество различных диаграмм.

Например, в диаграмму для разбиения $10 = 5 + 3 + 2$ можно вписать 36 различных диаграмм. Однако это не максимальное значение. В диаграмму для разбиения $10 = 4 + 2 + 2 + 1 + 1$ можно вписать 41 диаграмму Юнга.

Формат входных данных

Входной файл содержит целое число n ($1 \leq n \leq 100$).

Формат выходных данных

На первой строке выходного файла выведите максимальное число диаграмм Юнга, которые можно вписать в некоторую диаграмму, соответствующую разбиению на слагаемые числа n .

На второй строке выведите одно или более целых чисел — количество квадратиков в каждом из рядов оптимальной диаграммы.

Пример

стандартный ввод	стандартный вывод
10	41 4 3 2 1

Задача D. Леденящая игра

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Чтобы попасть в команду к Шкиперу пингвин должен пройти ряд испытаний: полоса препятствий от Шкипера, спарринг с Рико, расшифровка кода от Прапора и задача от Ковальски.

Вы, пингвин-новобранец, успешно дошли до последнего испытания. Ковальски предлагает вам сыграть в следующую игру. Вам дается m наборов разноцветных льдинок, каждая одного из n цветов. Различные цвета обозначаются различными прописными буквами латинского алфавита. Вы можете взять какое-то подмножество этих наборов при условии, что льдинка каждого цвета будет встречаться не более одного раза в этом подмножестве. Пусть вы выбрали k наборов с индексами i_1, i_2, \dots, i_k , тогда ваш выигрыш составляет $\sum_{j=1}^k l_{i_j} - k$ баллов, где l_{i_j} — количество льдинок в наборе i_j .

Ковальски требует найти подмножество с максимальным количеством баллов.

От вас требуется найти любое подмножество, подходящее под условия Ковальски.

Формат входных данных

В первой строке входного файла находится число n ($1 \leq n \leq 17$) — количество различных цветов. Вторая строка входного файла содержит число m ($1 \leq m \leq 200000$) — количество различных наборов льдинок. В следующих m строках перечислены сами наборы. Набор с номером i задаётся строкой из первых n строчных латинских букв. Длина каждой строки не больше 17 символов.

Формат выходных данных

В первой строке выходного файла выведите k — количество наборов в ответе. Во второй строке выходного файла выведите k чисел — индексы наборов, входящих в ответ, в произвольном порядке.

Примеры

стандартный ввод	стандартный вывод
1 3 aaa aaaa a	0
1 2 aaa aaaa	0
3 3 aba ab c	1 2

Задача Е. 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.

Пример

<code>dowry.in</code>	<code>dowry.out</code>
3 6 8	1
3 10	2
7 3	
8 2	

Задача F. Небоскрёбы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вы когда-нибудь мечтали стать главным героем компьютерной игры? Главный герой этой истории, Бранимир, мечтает сейчас именно об этом.

Мир в мечте Бранимира состоит из N небоскребов, пронумерованных слева направо. Для i -го небоскреба, известна его высота H_i и количество золотых монет G_i на крыше этого небоскреба. Игра начинается с прыжка на любой из небоскребов и состоит из нескольких ходов. На каждом ходу Бранимир может прыгнуть на любой небоскрёб, находящийся **справа** от него, но так, чтобы высота нового небоскрёба была **не меньше** того небоскрёба, на котором сейчас сидит Бранимир. Оказавшись на крыше небоскреба, Бранимир собирает все золотые монеты на ней. Бранимир может закончить игру после любого количества шагов (возможно, нулевого), но он должен собрать не менее K золотых монет, чтобы перейти на следующий уровень.

Бранимир хочет узнать, сколько существует способов сыграть в эту игру так, чтобы перейти на следующий уровень. Две игры называются разными, если существует небоскрёб который был посещен в одной игре, но не был посещён в другой.

Формат входных данных

Первая строка содержит 2 натуральных числа N и K ($1 \leq N \leq 40$, $1 \leq K \leq 4 \cdot 10^{10}$) — число небоскрёбов и количество монет, которые надо набрать соответственно.

Следующие N строк содержат информацию о небоскрёбах. В i -й строке даны 2 числа H_i и G_i ($1 \leq H_i, G_i \leq 10^9$) — высота и количество монет на i -м небоскрёбе.

Формат выходных данных

В единственной строке вывода выведите число возможных игр, в которых Бранимир сможет пройти на следующий уровень.

Система оценки

Решение, корректно работающее при $n \leq 20$ будет оцениваться в 40 баллов.

Примеры

стандартный ввод	стандартный вывод
4 6 2 1 6 3 7 2 5 6	3
2 7 4 6 3 5	0
4 15 5 5 5 12 6 10 2 1	4

Задача G. Мать драконов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.3 секунд
Ограничение по памяти: 256 мегабайт

В Королевстве Ланнистеров n замков и несколько стен, соединяющих два замка, никакие два замка не соединены более, чем одной стеной, ни одна стена не соединяет замок с собой.

Сир Джейме Ланнистер узнал, что Дейенерис Таргариен собирается атаковать его королевство. Он хочет защитить свои владения. У него есть k литров странной жидкости. Он хочет распределить эту жидкость между замками так, чтобы каждый замок содержал некоторое количество жидкости (возможно, нулевое или нецелое количество литров). После этого стабильность стены, соединяющей замки a и b , содержащие x и y литров жидкости, соответственно, равна $x \cdot y$.

Ваша задача — найти максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь

Формат входных данных

Первая строка содержит два целых числа n и k ($1 \leq n \leq 40$, $1 \leq k \leq 1000$).

Далее следует n строк. В i -й из них содержится n целых чисел $a_{i,1}, a_{i,2}, \dots, a_{i,n}$ ($a_{i,j} \in \{0, 1\}$). Если замки i и j соединены стеной, $a_{i,j} = 1$. В противном случае оно равно 0.

Гарантируется, что $a_{i,j} = a_{j,i}$ и $a_{i,i} = 0$ для всех $1 \leq i, j \leq n$.

Формат выходных данных

Выведите одно число — максимальную возможную сумму стабильностей стен, которую Сир Джейме Ланнистер сможет достичь.

Ваш ответ будет считаться правильным, если его абсолютная или относительная точность не превосходит 10^{-6} .

А именно, если ваш ответ равен a , а ответ жюри равен b , то ваш ответ будет зачтен, если $\frac{|a-b|}{\max(1,b)} \leq 10^{-6}$.

Примеры

стандартный ввод	стандартный вывод
3 1 0 1 0 1 0 0 0 0 0	0.250000000000
4 4 0 1 0 1 1 0 1 0 0 1 0 1 1 0 1 0	4.000000000000

Замечание

В первом примере, если замки 1, 2, 3 содержат 0.5, 0.5, 0 литров жидкости, соответственно, ответ равен 0.25.

Во втором примере, если замки 1, 2, 3, 4 содержат 1.0, 1.0, 1.0, 1.0 литров жидкости, ответ равен 4.0.

Задача Н. Последовательности из 0 и 1 без трех единиц подряд

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

По данному натуральному n определите количество последовательностей длины n из «0» и «1», не содержащих трех единиц подряд.

Формат входных данных

В первой и единственной строке задано число n . $1 \leq n \leq 10^9$.

Формат выходных данных

Выведите ответ по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
3	7

Задача I. 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 ($1 \leq a_i \leq k$).

Пример

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 J. Переворот битов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дана таблица $n \times m$ из 0 и 1. За одно действие вы можете поменять все биты на противоположные (то есть все 0 на 1 и все 1 на 0) в любом столбце или в любой строке этой таблицы. Вы можете совершить k действий. Какое максимальное количество 1 вы можете получить?

Формат входных данных

Первая строка содержит три целых числа n , m и k ($n \times m \leq 500$, $k \leq 1000$).

Следующие n строк содержат описание таблицы: в i -й из этих строк содержится битовая строка длины m .

Формат выходных данных

Выведите единственное число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
3 2 2 11 10 11	5
5 4 4 0011 1100 0001 0101 0010	16