

Задача А. Гладкие числа

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

Назовем число гладким, если его цифры, начиная со старшего разряда, образуют неубывающую последовательность. Упорядочим все такие числа в возрастающем порядке и присвоим каждому номер. Вам требуется по номеру N вывести N -ое гладкое число.

Формат входных данных

На вход программы поступает номер N ($1 \leq N \leq 2147483647$).

Формат выходных данных

Выведите соответствующее номеру N гладкое число.

Примеры

стандартный ввод	стандартный вывод
3	3
11	12

Задача В. Трипростые числа

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

Будем называть натуральное число трипростым, если в нем любые подряд идущие 3 цифры образуют трехзначное простое число. Требуется по данному N найти количество N -значных трипростых чисел.

Формат входных данных

На вход подаётся одно натуральное число N : ($3 \leq N \leq 10^4$).

Формат выходных данных

Ответ должен содержать количество N -значных трипростых чисел, которое следует вывести по модулю $10^9 + 9$.

Примеры

стандартный ввод	стандартный вывод
3	143
4	204
4793	851557205

Задача С. 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 D. 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 Е. 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 F. Раскраска графа

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

стандартный ввод	стандартный вывод
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 G. Симпатичные узоры

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

Компания Broken Tiles планирует заняться выкладыванием во дворах у состоятельных клиентов узоров 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 Васе необходимо узнать, сколько клиентов он сможет обслужить, прежде чем симпатичные узоры данного размера закончатся. Помогите ему!

Формат входных данных

Вводятся два положительных целых числа N и M ($1 \leq N \cdot M \leq 30$).

Формат выходных данных

Выведите единственное число – количество различных симпатичных узоров, которые можно выложить во дворе размера $N \times M$. Узоры, получающиеся друг из друга сдвигом, поворотом или отражением, считаются различными.

Примеры

стандартный ввод	стандартный вывод
1 1	2
1 2	4
4 1	16

Задача Н. Симпатичные узоры 2

Имя входного файла:	nice2.in
Имя выходного файла:	nice2.out
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 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.

Примеры

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

Задача I. Разносчик пиццы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
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.

Пример

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 J. Макс клика

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Макс, опытейший игрок в доту, постоянно кликал.

Дан случайный неориентированный граф G из n вершин и m ребер.
Подкликой называется такое подмножество вершин A : $\forall a, b \in A, a \neq b \quad \exists$ ребро (a, b) .
Ваша задача — найти подклику A : $|A|$ максимально.

Формат входных данных

На первой строке число вершин $n \geq 1$ и число ребер $m \geq 1$.
Следующие m строк содержат пары чисел от 1 до n — ребра графа.
В графе нет ни петель, ни кратных ребер.

Формат выходных данных

На первой строке выведите k — количество вершин в максимальной подклике. На следующей строке k целых чисел от 1 до n — номера вершин в подклике. Вершины можно выводить в любом порядке. Если максимальных подклик несколько, выведите любую.

Система оценки

Подзадача 1 (10 баллов) $n \leq 20$.
Подзадача 2 (10 баллов) $n \leq 25$.
Подзадача 3 (20 баллов) $n \leq 40$.
Подзадача 4 (20 баллов) $n \leq 60$.
Подзадача 5 (20 баллов) $n \leq 70$.
Подзадача 6 (20 баллов) $n \leq 80$.

Пример

стандартный ввод	стандартный вывод
5 8	4
5 4	3 5 1 4
3 5	
1 5	
1 3	
2 3	
1 4	
5 2	
3 4	