

Задача A. 17 стульев

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

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

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. Эстафета

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

Каждый год в честь дня города в Южно-Берляндске проводится открытая эстафета. В конкурсе участвуют команды из k человек, в процессе эстафеты участники команды должна посетить n контрольных пунктов.

Контрольные пункты пронумерованы от 1 до n , место старта обозначим как пункт 0. Соревнование проходит следующим образом: первый участник из команды стартует из пункта 0, пробегает по некоторым a_1 ранее не посещенным контрольным пунктам, возвращается в пункт 0 и передает эстафету второму участнику. После этого второй участник пробегает какие-либо a_2 ранее не посещенных контрольных пунктов, возвращается и передает эстафету следующему. Эстафета продолжается, пока последний участник не посетит a_k ранее не посещенных контрольных пунктов и не вернется на старт. Передача эстафеты происходит мгновенно. Цель команды — пробежать эстафету как можно быстрее.

Учащиеся Южно-Берляндского бегового училища решили заранее подготовиться к состязанию. Они раздобыли план соревнования, из которого они узнали числа a_i , а также выяснили про каждую пару пунктов, за какое время можно успеть добежать от одного до другого. Все участники команды перемещаются с одинаковой скоростью, поэтому это время не зависит от того, кто побегит между этими пунктами.

Помогите участникам составить маршрут, в котором команда пробежит эстафету как можно быстрее.

Формат входных данных

В первой строке находятся два целых числа n и k ($1 \leq n \leq 18$, $1 \leq k \leq n$) — число контрольных пунктов и число участников в команде.

Во второй строке находятся k целых чисел a_i ($1 \leq a_i \leq n$, $a_1 + a_2 + \dots + a_k = n$) — число контрольных пунктов, которые должен пробежать i -й участник.

В следующих $n + 1$ строках находится по $n + 1$ целому числу $b_{i,j}$ ($1 \leq b_{i,j} \leq 10^6$, $b_{i,j} = b_{j,i}$, $b_{i,i} = 0$) — время, за которое можно добежать от i -го пункта до j -го для i и j от 0 до n .

Формат выходных данных

В единственной строке выведите одно целое число — минимальное время, за которое команда может пробежать эстафету.

Примеры

стандартный ввод	стандартный вывод
2 2 1 1 0 1 2 1 0 3 2 3 0	6
4 2 2 2 0 1 4 2 5 1 0 2 6 6 4 2 0 6 6 2 6 6 0 2 5 6 6 2 0	16

Задача С. Симпатичные узоры

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

стандартный ввод	стандартный вывод
2 2	14
3 3	322

Задача D. Два капитана

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

Как известно, у Черной Жемчужины два капитана: капитан Джек Воробей и Барбосса. Корабль содержит ровно n пушек, расположенных в ряд. Во время боя оба капитана раз в минуту одновременно дают команды своим матросам. Команды бывают следующих видов:

- `send l r` — послать своих матросов стрелять из пушек с номерами от l до r включительно
- `back l r` — отозвать всех своих матросов с пушек с номерами от l до r включительно. Если на каких-то пушках из этого отрезка нет матросов, подчиняющихся этому капитану, то с такими пушками ничего не происходит
- `gun` — принести еще одну бутылку рома

Каждая команда выполняется мгновенно, после чего сражение идет еще минуту до следующей команды. Если в какой-то момент у одной и той же пушки окажутся матросы, подчиняющиеся разным капитанам, они подерутся и убьют друг друга. Эта ситуация не устраивает никого из капитанов, и поэтому они обратились к вам с просьбой помочь им в решении этой проблемы.

Перед началом очередного сражения капитан Джек Воробей и Барбосса составили планы своих действий. Известно, что план капитана Джека Воробья состоит из m_1 команд, а план Барбоссы — из m_2 команд. В начале i -ой минуты боя каждый капитан дает своим матросам i -ую команду из своего плана, если в нем есть хотя бы i команд. Вам поручили исправить планы так, чтобы все матросы остались живы. Единственная доступная вам модификация плана сражения — вставка нескольких команд `gun` в любые места. Понятно, что капитаны не очень любят менять свои планы, поэтому суммарное количество команд, добавленных Вами в оба плана, должно быть минимально.

Формат входных данных

В первой строке дано число n — количество пушек на корабле ($1 \leq n \leq 10^9$).

Во второй строке задано число m_1 — количество команд в плане Джека Воробья ($1 \leq m_1 \leq 3000$). В следующих m_1 строках перечислены команды из плана Джека Воробья. Команды заданы так, как они описаны выше. Для всех команд, использующих l и r , верно, что $1 \leq l \leq r \leq n$. Гарантируется, что последняя команда в плане — `back 1 n`.

Во следующей строке задано число m_2 — количество команд в плане Барбоссы ($1 \leq m_2 \leq 3000$). В следующих m_2 строках перечислены команды из плана Барбоссы. Команды заданы так, как они описаны выше. Для всех команд, использующих l и r , верно, что $1 \leq l \leq r \leq n$. Гарантируется, что последняя команда в плане — `back 1 n`.

Формат выходных данных

В единственной строке выведете минимальное количество дополнительных команд.

Система оценки

Первая группа тестов стоит 20 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие, $(n, m_1, m_2 \leq 20)$.

Вторая группа тестов стоит 35 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие, $(n, m_1, m_2 \leq 300)$.

Третья группа тестов стоит 25 баллов. Баллы за эту группу начисляются только при прохождении всех тестов группы. Для всех тестов этой группы выполнено условие, $(n, m_1, m_2 \leq 1000)$.

Четвертая группа тестов стоит 20 баллов. Каждый тест этой группы стоит определенное количество баллов.

Пример

стандартный ввод	стандартный вывод
3	3
4	
send 1 1	
send 2 2	
back 1 1	
back 1 3	
5	
send 2 3	
send 1 1	
back 2 2	
rum	
back 1 3	

Задача E. 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

Задача F. 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 G. Join

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

В этом году ЛКШ проводится на новой базе. Её территория — прямоугольник $n \times m$, на котором расположены квадратные домики (один домик занимает квадратик 1×1).

Так как база совсем новая и между домиками ещё не проложены тропинки, оргкомитет ЛКШ решил проложить их своими силами. Так как ресурсы ограничены, было решено соединять только соседние домики (соседними считаются домики, граничащие по вертикали или горизонтали), причём так, чтобы от любого домика до любого можно было дойти единственным способом.

Эту задачу поручили Тоше — младшему преподавателю параллели A' и он уже готов с энтузиазмом приступить к её выполнению. Но сначала Антон хочет узнать, сколькими способами может это сделать. Конечно, такую простую задачу он может решить и сам, но уже поздно, а Тоша очень устал.

Помогите Тоше, чтобы он успел выспаться перед завтрашними парами.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n, m \leq 9$) — размеры базы. В следующих n строках содержатся по m символов, описывающих карту. Символ "." обозначает домик, а "*" — полянку для игры во фрисби.

Гарантируется, что на базе есть хотя бы один домик.

Формат выходных данных

Выведите одно целое число — число способов соединить домики по модулю 10^9 .

Примеры

стандартный ввод	стандартный вывод
2 2	4
2 2 *. .*	0

Задача Н. Украденный массив

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.8 секунд
Ограничение по памяти: 256 мегабайт

Мальчик Петя любит массивы. Недавно ему подарили огромный массив чисел F размера 2^n . Петя человек странный, и при виде массива из чисел сразу начинает считать какие-то суммы на нём. Специально для этого он купил в магазине новый пустой массив P размера 2^n и начал его заполнять по следующему правилу: $P[i] = \sum_{j \& i = j} F[j]$. Другими словами для каждого j , такого что j — подмаска i (т.е. побитовое «И» чисел i и j равно j), Петя прибавил $F[j]$ к изначально нулевому значению $P[i]$.

Но потом случилось ужасное — массив F украли! Теперь Петя хочет разыскать массив F , но он не помнит, какие значения там были изначально. Единственное что у него есть — массив P . Помогите Пете восстановить массив F .

Формат входных данных

В первой строке входных данных дано одно число n ($1 \leq n \leq 20$).

В следующей строке даны 2^n чисел, i -е из них — значение $P[i]$ ($-10^9 \leq P[i] \leq 10^9$) (нумерация ведётся с нуля).

Формат выходных данных

В одной строке выведите 2^n чисел, i -е из которых — значение $F[i]$ (нумерация ведётся с нуля).

Примеры

стандартный ввод	стандартный вывод
2 1 2 3 4	1 1 2 0
3 1 3 4 10 6 14 16 36	1 2 3 4 5 6 7 8

Задача J. Cake Tasting

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Карина очень любит пирожные. В этом месяце Карине удалось посетить n различных кондитерских, предлагающих дегустацию своих изделий. Для каждой кондитерской Карина записывала, какие виды пирожных она попробовала, в специальное приложение на своем смартфоне. Но после всех дегустаций Карина обнаружила, что приложение показывает только количество различных видов пирожных, которые она попробовала. Причем эту информацию можно посмотреть для любого из $2^n - 1$ подмножества кондитерских. Карина очень расстроена и считает, что даже эти данные приложение вычислило неверно.

Помогите Карине понять, верны ли данные, и, если верны, предъявите любой подходящий вариант записей о дегустациях.

Формат входных данных

В первой строке задано число n ($1 \leq n \leq 19$) — количество кондитерских. В следующей строке содержатся $2^n - 1$ чисел a_i ($1 \leq i \leq 2^n - 1$; $1 \leq a_i \leq 1000$). Число a_i равно количеству различных видов пирожных, которые Карина пробовала в кондитерских с номерами j , где j -й разряд двоичного представления числа i равен единице. Например, если S_k — множество видов пирожных в k -й кондитерской, то $a_1 = |S_1|$, $a_2 = |S_2|$, $a_3 = |S_1 \cup S_2|$, $a_4 = |S_3|$, и так далее.

Формат выходных данных

Если данные приложения неверны, выведите в единственной строке слово «No». Иначе, выведите в первой строке слово «Yes», а в следующих n строках выведите возможный вариант записей Карины. Для каждой кондитерской запись должна начинаться с числа k_i ($1 \leq k_i \leq 1000$), равного количеству видов пирожных в i -й кондитерской. Затем, в той же строке, выведите k_i различных чисел $s_{i,j}$, не превышающих 10^9 по модулю, — виды пирожных.

Если существует несколько возможных вариантов записей, выведите любой. Гарантируется, что если существует решение, то существует также и решение, в котором все $k_i \leq 1000$.

Примеры

стандартный ввод	стандартный вывод
2 2 3 4	Yes 2 1 4 3 1 2 3
2 2 2 5	No
3 3 2 4 3 4 4 5	Yes 3 1 2 5 2 1 4 3 1 2 3

Замечание

В первом примере есть две кондитерские. В первой кондитерской Карина попробовала 2 вида пирожных, во второй — 3, в первой и второй вместе — 4. Один из возможных вариантов, соответствующих этим числам, такой — в первой кондитерской Карина попробовала пирожные 1 и 4 вида, во второй — 1, 2 и 3 вида, тогда в первой и второй вместе она попробовала пирожные четырех видов: 1, 2, 3 и 4.

Во втором примере в первой и второй кондитерской Карина попробовала по 2 вида пирожных, а в сумме — 5 видов. Такого очевидно не могло быть.

В третьем примере есть три кондитерские. Один из возможных ответов такой:

$$a_1 = |S_1| = |\{1, 2, 5\}|,$$

$$a_2 = |S_2| = |\{1, 4\}|,$$

$$a_3 = |S_1 \cup S_2| = |\{1, 2, 4, 5\}|,$$

$$a_4 = |S_3| = |\{1, 2, 3\}|,$$

$$a_5 = |S_1 \cup S_3| = |\{1, 2, 3, 5\}|,$$

$$a_6 = |S_2 \cup S_3| = |\{1, 2, 3, 4\}|,$$

$$a_7 = |S_1 \cup S_2 \cup S_3| = |\{1, 2, 3, 4, 5\}|.$$