

Задача А. Битовая сортировка 2

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Отсортируем все целые числа от 1 до n включительно по количеству единиц в их двоичном представлении. Таким образом, число $4 = 100_2$ идёт раньше числа $3 = 11_2$, так как в двоичном представлении имеет на одну единицу меньше. В случае одинакового количества единиц раньше идёт меньшее число.

Пример отсортированных чисел для $n = 10$: 1, 2, 4, 8, 3, 5, 6, 9, 10, 7.

Даны числа n и k . Требуется найти k -е число в указанном выше порядке.

Нумерация начинается с 1.

Формат входных данных

В первой строке записаны два целых числа n и k ($1 \leq k \leq n \leq 10^{18}$).

Формат выходных данных

В единственной строке выведите k -е число в последовательности.

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
10 7	6
123 1	1
51 44	30

Задача В. Шахматы

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Маленькому Ване всегда нравилось наблюдать за тем, как папа играет в шахматы. И вот недавно он попросил своего папу объяснить и ему правила этой увлекательной игры. Теперь Ваня уже знает названия всех фигур и правила, по которым они ходят. Больше всего ему нравятся слон, ладья и ферзь.

Сегодня Ваня придумал для себя следующую задачу. Сначала он ставит на какую-то из клеток доски одну из своих любимых фигур. А затем для всех остальных клеток доски пытается определить, за какое минимальное число ходов поставленная им фигура может до туда добраться, если будет ходить согласно шахматным правилам. Только вот незадача — Ваня не может проверить, правильно ли он решил свою задачу, ведь папа сейчас на работе. Помогите маленькому Ване с проверкой его решения.

Формат входных данных

В первой строке записано название фигуры: «queen», «bishop» или «rook» — для ферзя, слона и ладьи соответственно. Во второй строке даны координаты клетки, в которой стоит данная фигура, в шахматной нотации. Клетка задана вертикалью и горизонталью. Вертикали пронумерованы буквами от «a» до «h» слева направо, горизонтали — от 1 до 8 снизу вверх.

Формат выходных данных

Выведите 8 строк по 8 символов, каждый символ обозначает минимальное количество ходов до соответствующей клетки поля. Клетка «a1» находится в нижнем левом углу. Если до клетки нельзя добраться ни за какое число ходов, то соответствующий символ равен «*». Иначе данный символ — это число, равное минимальному количеству ходов, за которое до неё можно прийти. Гарантируется, что если до клетки можно добраться, то минимальное число ходов будет меньше 10.

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
queen b6	21212222 11122222 10111111 11122222 21212222 21221222 21222122 21222212
bishop a1	*2*2*2*1 2*2*2*1* *2*2*1*2 2*2*1*2* *2*1*2*2 2*1*2*2* *1*2*2*2 0*2*2*2*

Подсказка

Ферзь ходит на любое число полей по вертикали, горизонтали или диагонали

Ладья ходит на любое число полей по вертикали или горизонтали

Слон ходит на любое число полей по диагоналям

Задача С. Какой сегодня день? - 2

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На Новый две тысячи тринадцатый год Егорке подарили большие красивые наручные часы. Кроме точного времени эти часы показывали ещё и дату. Егорка был очень рад новой вещи, однако вскоре он заметил, что дата на часах отображается в непривычном формате — указан только номер дня в году. Например, 1 января у него на часах выглядит как число 1, а 1 февраля — как число 32.

Егорка хочет по числу на его часах определять месяц и номер дня в этом месяце, которые этому числу соответствуют. Помогите Егорке в этом. Егорка будет счастлив, если ваша программа будет работать только для дат в 2013 году.

Формат входных данных

В первой строке дано число n — номер интересующего Егорку дня в 2013 году ($1 \leq n \leq 365$).

Формат выходных данных

Выведите через пробел название месяца и номер дня в этом месяце, которые соответствуют n -ому дню 2013 года. Названия месяцев должны выглядеть следующим образом (в порядке от января до декабря): «January», «February», «March», «April», «May», «June», «July», «August», «September», «October», «November», «December».

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
256	September 13
55	February 24

Задача D. Тесты на заболевания

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Пётр Петрович уже много лет работает в диагностическом отделе ветеринарной клиники. Его работа заключается в том, чтобы на основании проводимых им тестов делать выводы о возможных заболеваниях конкретного животного.

Каждый тест представляет собой следующую процедуру. Пётр Петрович смешивает образец крови исследуемого животного с тестовым материалом. Если данное животное страдает хотя бы одним из заболеваний, которое данный тест проверяет, то только в этом случае происходит химическая реакция, эффект которой легко пронаблюдать. Однако такая положительная реакция на различные заболевания одна и та же, то есть если тест определяет присутствие заболеваний A и B и в результате будет иметь место химическая реакция, то нельзя будет установить, каким из заболеваний A или B страдает объект исследования. Пётр Петрович называет тестовые материалы *эквивалентными*, если они способны определять в точности одинаковый набор заболеваний.

Тестовые материалы можно смешивать между собой. Тестовый материал, полученный в результате смешивания нескольких других материалов может определять наличие в точности тех заболеваний, которые мог идентифицировать хотя бы один из исходных. К примеру, если Пётр Петрович смешивает два тестовых материала, один из которых определяет присутствие болезней A и B , а другой — B и C , то он получит новый материал, диагностирующий заболевания A , B и C .

Недавно директор ветеринарной клиники обратил внимание Петра Петровича на то, что для диагностирования заболеваний в клинике используется слишком много различных тестовых материалов, а владение лицензией на использование каждого из них обходится учреждению довольно дорого. Теперь Петру Петровичу необходимо сообщить начальству, от использования какого наибольшего количества типов тестовых материалов, на которые нужна лицензия, он готов отказаться. Материалы, полученные путём смешивания, не требуют отдельной лицензии, кроме того, их так же можно использовать для смешивания. Множество тестовых материалов, которые можно получить с помощью некоторого, возможно нулевого, количества смешиваний материалов из конкретного набора называется *расширением* этого набора. Два набора тестовых материалов считаются *эквивалентными*, если для каждого материала из расширения первого набора существует эквивалентный ему тестовый материал из расширения второго набора, и наоборот. Естественно, Пётр Петрович может отказываться лишь от такого набора материалов, что оставшийся набор эквивалентен исходному.

Формат входных данных

В первой строке записано целое число n — количество используемых Петром Петровичем тестовых материалов, для которых нужна лицензия ($1 \leq n \leq 200$). В каждой из следующих n строк содержится описание очередного тестового материала. Первое число в описании k_i — количество болезней, которые диагностирует i -ый материал ($1 \leq k_i \leq 300$). За ним через пробел следуют k_i чисел — идентификаторы этих болезней. Идентификаторы болезней — целые числа от 1 до 300.

Формат выходных данных

В первой строке выведите наибольшее количество типов тестовых материалов, от которых Пётр Петрович может позволить себе отказаться.

Пример

Входные данные	Выходные данные
5 2 1 2 2 2 3 3 1 2 3 4 1 2 3 4 1 4	2

Задача Е. Бабушкин сериал

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Маленький Вася очень любит играть с числами. Чаще всего он играет в следующую игру. Сначала Вася берёт какое-нибудь целое положительное число и называет его текущим. На каждом ходу он применяет к текущему числу следующую операцию: поделить строковое представление числа в десятичном виде на две части так, чтобы в каждой части была хотя бы одна цифра, после чего перемножить эти части как числа и получить новое значение текущего числа. Например, число 114 можно разделить двумя способами. Первый способ — 11 и 4, и после умножения у маленького Васи получится число 44. Второй — 1 и 14, в этом случае после умножения текущее число станет равным 14. Когда у Васи остаётся число, состоящее из одной цифры, игра заканчивается.

Главной проблемой для маленького Васи всегда было придумать число, с которого начнётся игра. Вчера вечером за ужином Васина бабушка невзначай обмолвилась, что только что посмотрела n -ую серию её любимого сериала. И когда сегодня Васе вдруг опять захотелось поиграть с числом, он вспомнил об этом и решил начать с числа n .

Конечно же, Вася хочет играть как можно дольше, и ему интересно узнать, какое наибольшее число ходов может продолжаться игра, начавшаяся с числа n .

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите максимальное количество ходов, которые Вася может сделать, начав с числа n . Если Вася сможет играть бесконечно, выведите -1 .

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
114	3
14	1

Задача F. Жизнь

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На прямой расположены n клеток, они пронумерованы целыми числами от 1 до n вдоль этой прямой. Клетки считаются соседними, если их номера отличаются ровно на 1. В начальный момент времени k из них живые, остальные — мёртвые. Если в течение минуты хотя бы одна из соседних клеток мёртвой клетки была живой, то мёртвая клетка оживает. Живые клетки остаются живыми независимо от своих соседей. Сколько будет живых клеток через t минут?

Формат входных данных

В первой строке записаны три целых числа n, k, t ($1 \leq n \leq 10^9; 1 \leq k \leq 1000; 0 \leq t \leq 10^9$). В следующей строке заданы k целых чисел через пробел — номера живых клеток ($1 \leq a_i \leq n$). Гарантируется, что номера всех живых клеток различны.

Формат выходных данных

Выведите количество живых клеток через t минут.

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
10 2 2 3 8	10
100 1 1 50	3

Задача G. Тетрис

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Маленький Гриша часто играет в тетрис. Гришин тетрис немного отличается от обычного. В игре по-прежнему требуется управлять падающими фигурками, но есть некоторые нюансы.

При старте на экране можно увидеть игровое поле и n фигурок, расположенных в том порядке, в котором они будут появляться в игре.

Игровое поле представляет собой прямоугольник шириной 10 и высотой 20 клеток. Верхняя левая клетка поля имеет координаты $(1, 1)$, а нижняя правая — $(20, 10)$. Каждая клетка поля может быть либо занята блоком, либо свободна.

Фигурка представляет собой набор блоков и задается квадратом 4×4 , некоторые клетки которого закрашены. Верхняя строчка квадрата всегда содержит хотя бы одну закрашенную клетку. Все фигурки в игре связные, то есть между любыми двумя закрашенными клетками существует закрашенный путь из смежных по стороне клеток.

В начале каждого хода на поле появляется очередная фигурка. Фигурка появляется так, что левая верхняя клетка соответствующего ей квадрата совмещается с клеткой поля с координатами $(1, 4)$. Если фигурка не может появиться на поле, а именно, клетка, которую она хочет занять, уже занята, то игра заканчивается. Появившись на поле, фигурка с постоянной скоростью движется вниз, сдвигаясь за шаг целиком на одну клетку. Если фигурка не может сдвинуться вниз по причине того, что клетка, которую она хочет занять, уже занята, фигурка останавливается.

После остановки фигурки происходит следующий процесс. Пока на поле есть строчка, целиком занятая блоками, часть поля, расположенная выше этой строчки, сдвигается на одну клетку вниз. При этом верхняя строчка поля становится свободной. Когда занятые строчки заканчиваются, начинается новый ход, то есть сверху появляется следующая фигурка.

Пока фигурка не остановилась, игрок может ее двигать влево или вправо на произвольное расстояние. При движении ни один блок фигурки не должен накладываться на уже занятую клетку или выходить за пределы поля. Гриша — довольно опытный игрок, поэтому он всегда успевает сдвинуть фигурку до того, как она опустится на одну клетку ниже или остановится.

Заметим, что в Гришином тетрисе осуществлять поворот и движение вверх фигурок не разрешается.

Гриша уже давно разработал свою стратегию игры в этот тетрис. Из всех мест, где может остановиться фигурка, он выбирает такое, чтобы фигурка оказалась как можно ниже, а в случае неопределённости — как можно левее. Теперь Гришу интересует вопрос, сможет ли он продержаться первые n ходов, строго придерживаясь своей стратегии.

Формат входных данных

В первой строке дано единственное число n ($1 \leq n \leq 1000$). В следующих 20 строках содержится по 10 символов — описание поля в начале игры. Занятая клетка обозначается символом «#», а свободная — «.». Гарантируется, что изначально на поле нет строк, состоящих полностью из занятых клеток. Следующие n блоков описывают фигурки в порядке их появления в игре. Каждый блок представляет собой 4 строки по 4 символа — описание квадрата фигурки в том же формате, что и поля. Описание поля и первый блок, а также соседние блоки с описанием фигурок разделены между собой переводом строки.

Формат выходных данных

Выведите «Game over on step k » где k — номер первой фигурки, которая не смогла появиться на поле. Если таких фигурок не было, выведите «Win». Вне зависимости от исхода выведите описание поля на момент конца игры.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
<pre> 4#..... #####..... ######. ..#. ..#.##. .##.##. .#.. .#.. </pre>	<pre> Game over on step 4#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...#..... ...###... ##..#.#... ##..#.#... </pre>

Задача N. Ширина многоугольника

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Рассмотрим многоугольник без самопересечений и самокасаний, но не обязательно выпуклый. Назовём *шириной* многоугольника минимальное расстояние между двумя параллельными прямыми, такими что каждая точка многоугольника либо лежит на одной из них, либо между ними.

Ваша задача состоит в том, чтобы найти ширину данного многоугольника.

Формат входных данных

В первой строке записано одно целое число n — количество вершин многоугольника ($3 \leq n \leq 200$). Далее в n строках перечислены пары координат вершин в порядке обхода по или против часовой стрелки x_i, y_i ($-1000 \leq x_i, y_i \leq 1000$).

Формат выходных данных

Выведите ширину многоугольника с абсолютной или относительной погрешностью 10^{-6} .

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
4 0 0 0 1 1 1 1 0	1.000

Задача I. Простая игра

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Маша и Таня решили сыграть в следующую игру. Маша записывает на доску число n и предоставляет Тане право ходить первой. За один ход необходимо выполнить одно из двух действий.

1. Разделить записанное на доске число на один из его простых делителей p .
2. Разделить записанное на доске число на один из его простых делителей p , а затем результат умножить на простое число q , такое что $q < p$.

В обоих случаях полученное в результате число нужно записать на доску вместо исходного.

Ходы девочки делают по очереди. Проигрывает та из них, кто не может сделать очередной ход.

Через несколько игр Таня заподозрила, что Маша всегда подбирает такое число n , которое обеспечивает ей победу. Помогите Тане понять, оправданы ли её подозрения. Для этого по заданному числу n определите, кто из девочек окажется победителем, при условии что и Таня, и Маша играют оптимальным образом.

Формат входных данных

В первой строке записано одно целое число n ($1 \leq n \leq 10^{12}$).

Формат выходных данных

Выведите «Masha», если победу одержит Маша, записав в начале игры число n , иначе выведите «Tanya».

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
16	Masha
2	Tanya

Задача J. Цифровой корень

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Обозначим за $S(x)$ сумму цифр в десятичной записи числа x . Цифровым корнем целого положительного числа x будем называть число $D(x)$ такое, что $D(x) = x$, если x меньше 10, и $D(x) = D(S(x))$ иначе. По данному числу k найдите такое n , чтобы $D(n) = k$.

Формат входных данных

В первой строке записано целое число k ($1 \leq k \leq 9$).

Формат выходных данных

Выведите число n такое, что его цифровой корень равен k ($1 \leq n \leq 10^6$). Гарантируется, что ответ всегда существует. Если есть несколько возможных вариантов ответа, то выведите любой.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
9	189

Задача К. Автозамена

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дана строка из строчных латинских букв и несколько запросов к ней. Каждый запрос характеризуется двумя строчными латинскими буквами a_i и b_i и заключается в том, чтобы заменить в строке все вхождения буквы a_i на b_i . Ваша задача заключается в том, чтобы вывести строку после последовательного выполнения всех запросов. Запросы должны быть выполнены в том порядке, в котором они заданы на входе.

Формат входных данных

В первой строке записано два числа n и m — длина строки и количество запросов ($1 \leq n, m \leq 10^5$). В следующей строке записана исходная строка длиной n , состоящая из строчных латинских букв. В следующих m строках записано по две строчные латинские буквы a_i и b_i через пробел, описывающие запросы.

Формат выходных данных

Выведите строку после выполнения всех запросов.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
5 3 ababa a c b a c b	babab

Задача L. Переливания

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Имеются две пустые канистры объемом a и b литров и бак с водой. За один шаг можно выполнить одно из трёх действий: набрать из бака воды в канистру, пока та не заполнится; вылить всю воду из канистры в бак; перелить воду из одной канистры в другую, пока либо не закончится вода, либо не заполнится другая канистра. Изначально в баке $a + b$ литров воды.

Необходимо проделать некоторую последовательность переливаний, чтобы в результате в одной из канистр оказалось ровно k литров воды.

Формат входных данных

В первой строке записаны три целых числа a , b и k ($1 \leq a, b \leq 100$, $0 \leq k \leq \max(a, b)$)

Формат выходных данных

Если серией переливаний можно добиться того, чтобы в одной из канистр было ровно k литров воды, то в первой строке выведите n — количество необходимых для этого переливаний ($0 \leq n \leq 2 * 10^4$), в следующих n строках выведите по два числа от 0 до 2. Первое число обозначает, откуда переливать воду, второе — куда (0 соответствует баку, 1 — первой канистре, 2 — второй).

Если указанной цели достичь невозможно, выведите -1 . Если цели можно добиться несколькими удовлетворяющими условию способами, то вы можете вывести любой из них.

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
3 4 1	2 0 2 2 1
2 4 3	-1