

Задача А. Лифт

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Недавно в городе N-ске достроили 1000-этажный небоскреб. Для перемещения по этажам небоскреба используются современные сверхскоростные лифты. У каждого лифта имеются две двери: передняя и задняя. Если человек входит в лифт через переднюю дверь, то выходит он через заднюю, и наоборот. В лифте имеются два поручня, пронумерованных числами 1 и 2. Поручень номер 1 находится слева от входа в переднюю дверь (или, соответственно, справа от входа в заднюю дверь), а поручень номер 2 — справа (слева от входа в заднюю дверь). Известно, что каждый человек в N-ске держится за поручень той рукой, которая у него наиболее развита.

Некоторое время назад город посетил очень важный человек, который не преминул посетить небоскреб и проехать на лифте. Срочно необходимо определить, является ли он левой или правой, зная, в какую дверь он зашел и за какой поручень держался.

Формат входного файла

В первой строке записано, через какую дверь важный человек зашел в лифт. В ней записано «front», если человек зашел в лифт через переднюю дверь, и «back», если через заднюю. Во второй строке содержится целое число a ($1 \leq a \leq 2$), обозначающее номер поручня, за который держался пассажир.

Формат выходного файла

Выведите символ «R», если очень важный человек является правой, или «L», если он является левой.

Примеры

input.txt	output.txt
front	L
1	

Задача В. Что? Где? Когда?

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

В Берляндии очень популярна игра «Что? Где? Когда?». Суть игры заключается в противостоянии команды Знатоков (игровой команды из шести человек) команде Телезрителей. Знатоки должны за одну минуту при помощи мозгового штурма найти ответ на вопрос, присланный телезрителем. Как правило, игрокам задаются вопросы, ответить на которые можно с помощью общих знаний и логики. Вопросы находятся в разложенных на столе по кругу конвертах, подписанных названиями городов участников, приславших

вопрос. Каждый вопрос лежит в отдельном секторе. Посередине стола находится волчок (юла) со стрелкой. Волчок раскручивает распорядитель, и вопрос из сектора, на котором останавливается стрелка, выбирается для игры. Если выпавший вопрос уже сыгран, то выбирается следующий за ним по часовой стрелке еще не сыгравший вопрос. Ваша задача определить, какой по номеру вопрос будет следующим в игре, если волчок покажет на сектор с номером k .

Формат входного файла

В первой строке записаны два натуральных числа n и k ($1 \leq n \leq 1000$ и $1 \leq k \leq n$) — количество секторов и номер сектора, на который указывает стрелка волчка. Во второй строке дано n чисел: $a_i = 0$, если вопрос из сектора i уже сыгран, и $a_i = 1$, если вопрос из сектора i еще в игре ($1 \leq i \leq n$). Сектора заданы по часовой стрелке, после n -го сектора идет первый.

Формат выходного файла

Выведите единственное число — номер сектора, вопрос на котором будет выбран для игры. Гарантируется, что ответ существует, то есть еще не все вопросы были сыграны.

Примеры

input.txt	output.txt
5 5 0 1 0 1 0	2
2 1 1 1	1

Задача С. Винни-Пух и мед

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Винни-Пух, как известно, очень любит мед. Как-то раз они с Пятачком прознали, что у Кролика вновь появилось изрядное количество этого сладкого лакомства. Как вы можете догадаться, вскоре Винни и Пятачок напросились в гости. Итак, перед Винни-Пухом стоят в ряд n баночек с медом, в банке с номером i находится a_i килограмм меда. Винни-Пух ест мед следующим образом: каждый раз он выбирает банку, в которой больше всего меда (в случае, если таких банок несколько, он берет любую из них). Если в этой банке меньше, чем k килограмм меда, или же Винни уже трижды ел мед из этой банки, то он отдает ее Пятачку. В противном случае он ест из нее ровно k килограмм меда и ставит ее на место. Винни делает так до тех пор, пока не отдаст все банки Пятачку. Посчитайте, сколько меда в сумме получит Пятачок после того, как Винни удовлетворит свой аппетит.

Формат входного файла

В первой строке записано два целых числа n и k ($1 \leq n \leq 100$, $1 \leq k \leq 100$). Во второй строке записаны n целых чисел a_1, a_2, \dots, a_n , разделенных пробелами ($1 \leq a_i \leq 100$).

Формат выходного файла

Выведите одно число — сколько килограмм меда достанется Пятачку.

Примеры

input.txt	output.txt
3 3 15 8 10	9

Задача D. Три сына

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Старый фермер оставил трем своим сыновьям наследство — прямоугольное кукурузное поле, разбитое на $n \times m$ клеточек. Про каждую клеточку известно, сколько тонн кукурузы на ней растет. Своих сыновей фермер любил неодинаково, поэтому завещал разделить свое наследство на три части с суммарным количеством кукурузы A , B и C тонн.

Поле должно быть разделено двумя прямыми линиями, которые параллельны одной из сторон поля, а также друг другу. Линии должны проходить исключительно между клетками поля. Каждая полученная в результате часть поля должна состоять хотя бы из одной клеточки.

Ваша задача — найти количество способов разделить поле описанным выше способом, то есть провести две линии, рассекающие поле на три части таким образом, чтобы на одной из получившихся частей росло A тонн кукурузы, на другой B , а на оставшейся C .

Формат входного файла

В первой строке через пробел записаны целые числа n и m — размеры участка ($1 \leq n, m \leq 50, \max(n, m) \geq 3$). Далее следует описание поля: n строк, каждая из которых содержит по m целых чисел c_{ij} , записанных через пробел ($0 \leq c_{ij} \leq 100$) — количество кукурузы в каждой клетке (в тоннах). В последней строке через пробел записаны целые числа A, B, C ($0 \leq A, B, C \leq 10^6$).

Формат выходного файла

Выведите ответ — количество способов разделить наследство так, чтобы на одной из получившихся частей росло A тонн кукурузы, на другой B , а на оставшейся C . Если не существует ни одного способа, выведите 0.

Примеры

input.txt	output.txt
3 3 1 1 1 1 1 1 1 1 1 3 3 3	2
2 5 1 1 1 1 1 2 2 2 2 2 3 6 6	3
3 3 1 2 3 3 1 2 2 3 1 5 6 7	0

Note

Линии раздела могут быть как горизонтальными, так и вертикальными, но должны быть параллельны друг другу.

Задача E. Поставь Коня!

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Петя и Гена играют в очень занимательную игру «Поставь Коня!» на шахматной доске размером $n \times n$. В этой игре ребята по очереди ставят на доску шахматных коней так, чтобы никакие два коня не находились под ударом друг друга. Под ударом шахматного коня, находящегося в клетке (r, c) , находятся клетки $(r-1, c+2)$, $(r-1, c-2)$, $(r+1, c+2)$, $(r+1, c-2)$, $(r-2, c+1)$, $(r-2, c-1)$, $(r+2, c+1)$ и $(r+2, c-1)$ (некоторые из них могут находиться вне доски). Проигрывает тот, кто не может сделать ход. Определите, кто выиграет при оптимальной игре обоих, если первым ходит Петя.

Формат входного файла

В первой строке задано целое число T ($1 \leq T \leq 100$) — количество досок, для которых вам надо определить исход игры. В следующих T строках заданы T целых чисел n_i ($1 \leq n_i \leq 10000$) — размеры шахматных досок.

Формат выходного файла

Для каждой доски $n_i \times n_i$ выведите в отдельной строке «0», если при оптимальной игре выигрывает Петя, иначе выведите «1».

Примеры

input.txt	output.txt
2	1
2	0
1	

Задача F. Пауки

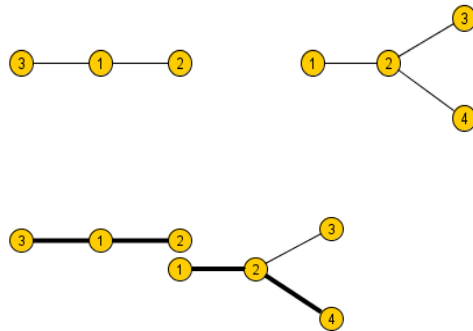
Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Однажды мама попросила Петю разобрать игрушки и избавиться от части из них. Петя нашел целую коробку игрушечных пауков. Они очень ему дороги, и мальчик не хочет их выкидывать. Петя придумал хитрый план: он склеит всех паучков и подвесит к потолку. Также он знает, что чем ниже будет свисать эта конструкция, тем больше понравится маме, и она не станет выкидывать его самые любимые игрушки. Помогите Пете осуществить его план.

Паук состоит из k бусинок, связанных $k - 1$ ниточкой. Каждая ниточка соединяет две различные бусинки, при этом любая пара бусинок, составляющих паука, либо непосредственно связана ниточкой, либо связана некоторой цепочкой из ниточек и бусинок.

Склеивать пауков Петя может непосредственно за бусинки. Длина каждой нитки равна 1. Размерами бусинок можно пренебречь. Поэтому можно считать, что склеивание пауков происходит путем отождествления некоторых бусинок (см. рисунок). При этом конструкция, получаемая при склеивании, также должна представлять собой паука, т.е. для нее должны выполняться указанные свойства.

После того, как Петя склеит всех пауков, он считает длину получившейся поделки. Расстояние между парой бусинок вычисляется как длина ниточек, соединяющих эти две бусинки. Длиной полученной конструкции называется наибольшее расстояние между всеми парами бусинок. Петя хочет получить конструкцию как можно большей длины.



На рисунке изображено 2 паучка из второго примера. К бусинке номер 2 первого паучка мы приклеим второго паучка за бусинку с номером 1. На рисунке выделены ниточки в паучках, которые образуют последовательность ниточек максимальной длины.

Формат входного файла

В первой строке входного файла записано одно целое число n ($1 \leq n \leq 100$) — количество пауков. Следующие n строк содержат описания каждого паука: целое число n_i ($2 \leq n_i \leq 100$) — количество бусинок, далее $n_i - 1$ пара чисел, означающих номера бусинок, соединенных нитками. Бусинки, составляющие каждого паука, нумеруются от 1 до n_i .

Формат выходного файла

Выведите одно число — длину искомой конструкции.

Примеры

input.txt	output.txt
1 3 1 2 2 3	2
2 3 1 2 1 3 4 1 2 2 3 2 4	4
2 5 1 2 2 3 3 4 3 5 7 3 4 1 2 2 4 4 6 2 7 6 5	7

Задача G. Бум

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Рассмотрим известную игру Бум (ака Шляпа) с упрощенными правилами.

В игре участвуют n команд по два человека. Цель игры: объяснить слова своему напарнику, не используя однокоренные и созвучные.

Игрок номер j команды с номером i ($1 \leq i \leq n, 1 \leq j \leq 2$) характеризуется двумя числами: a_{ij} и b_{ij} . Числа обозначают соответственно навык объяснения и навык понимания данного игрока.

Так же для игры используются m карточек. На каждой карточке написано слово. Карточка с номером k ($1 \leq k \leq m$) характеризуется числом c_k — сложностью слова, написанного на ней.

Перед началом игры карточки складываются в колоду и перемешиваются. Далее команды делают ходы в следующем порядке: 1-й игрок 1-й команды, 1-й игрок 2-й команды, ..., 1-й игрок n -ой команды, 2-й игрок 1-й команды, ..., 2-й игрок n -й команды, 1-й игрок 1-й команды и т.д.

Каждый ход длится t секунд. Ход производится следующим образом: изначально время хода равно t . Пока время хода больше 0, ходящий игрок берет верхнюю карточку из колоды и начинает объяснять слово на ней своему напарнику. Время, за которое j -й игрок i -й команды объяснит слово с карточки k своему напарнику (q -му игроку i -й команды) равно $\max(1, c_k - (a_{ij} + b_{iq}) - d_{ik})$ (если $j = 1$, то $q = 2$, иначе $q = 1$). Величина d_{ik} — это количество секунд, которое i -я команда уже потратила на объяснение слова k за предыдущие ходы. Изначально, все d_{ik} равны 0. Если команда успевает до конца хода отгадать слово, то из времени хода вычитается указанное выше время, отгаданная карточка выходит из игры, принося команде одно очко, и ход продолжается. Если же команда не успевает отгадать слово, то карточка кладется вниз колоды, а d_{ik} увеличивается на время хода, потраченное на объяснение этого слова на текущем ходу. Таким образом, когда этой команде достанется это же слово, они начнут его объяснять не с начала, а с того места, на котором остановились. Игра заканчивается тогда, когда все m карточек отгаданы.

Вам даны n команд и колода из m карточек. Вам необходимо для каждой команды определить, сколько очков будет у этой команды в конце игры, и какие слова эта команда отгадает.

Формат входного файла

В первой строке содержатся два целых числа n, t ($1 \leq n, t \leq 100$), обозначающие соответственно количество команд и продолжительность одного хода.

В следующих n строках входного файла содержатся по четыре целых числа: $a_{i1}, b_{i1}, a_{i2}, b_{i2}$ ($1 \leq a_{ij}, b_{ij} \leq 100$) — навыки первого и второго игроков i -й команды. Команды заданы в том порядке, в котором они ходят.

В следующей строке входного файла содержится целое число m ($1 \leq m \leq 100$) — число карточек.

В следующих $2m$ строках содержится описания карточек. Каждая карточка описывается двумя строками. В первой строке описания карточки содержится слово, состоящее не более, чем из 20 символов. Слова содержат только строчные буквы латинского алфавита. Во второй строке описания карточки содержится целое число c_k ($1 \leq c_k \leq 100$) — сложность слова, написанного на k -й карточке. Карточки перечислены в том порядке, в котором они лежат в колоде сверху вниз. Слова на всех карточках различны.

Формат выходного файла

Выведите n строк. В i -й строке сначала выведите число s_i — количество очков, заработанных i -й командой. Далее выведите s_i слов, разделяя их пробелами, — слова с карточек, отгаданных командой i в том порядке, в котором они были отгаданы.

Примеры

input.txt	output.txt
<pre>2 2 1 1 1 1 1 1 1 1 3 home 1 car 1 brother 1</pre>	<pre>2 home car 1 brother</pre>
<pre>2 4 1 2 2 1 2 3 2 2 4 armchair 3 quetzalcoat1 10 pilotage 5 defibrillator 7</pre>	<pre>2 armchair quetzalcoat1 2 pilotage defibrillator</pre>

Задача Н. Краткость — сестра таланта

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Общаясь с людьми, человек узнает много различной информации. Однако процесс общения занимает очень много времени. Это становится понятным, если обратить внимание на слова, которые мы используем в нашей речи.

Можно привести множество примеров простых слов, в которых большое количество букв: «информатика», «клавиатура», «университет», «строительство», «консерватория», «сковородка», «холодильник», «секундомер», «подоконник», «электричество», «государство», «автомобиль» и другие. Разумеется, этот список можно продолжать до бесконечности.

К счастью, решение этой проблемы уже было найдено. Для краткости и простоты общения предлагается заменить все слова нашей речи на такие, которые будут похожи на исходные, но в то же время будут заметно короче. Однако реализации этой идеи пока не существует, поэтому было решено поручить вам исправить ситуацию.

Рассмотрим следующую формальную модель преобразования слов: будем считать, что в разговоре можно использовать n слов. Для каждого слова введем понятие его сокращенного аналога. **Сокращенным аналогом** произвольного слова s назовем такое слово t , которое удовлетворяет следующим условиям:

- встречается в s в качестве **подпоследовательности**,
- имеет длину от одного до четырех символов.

Другими словами, слово t состоит хотя бы из одного и не более чем из четырех символов, которые встречаются в том же порядке, но не обязательно подряд, в слове s . Разрешается не сокращать исходное слово, если его длина не превосходит четырех символов.

Вашей задачей является для заданного списка из n различных слов получить набор их сокращенных аналогов. Сокращенные аналоги всех слов из списка должны быть различными.

Формат входного файла

В первой строке входного файла задано единственное целое число n ($1 \leq n \leq 200$). Далее в n строках задан набор различных непустых слов, состоящих из строчных букв латинского алфавита. Длина каждого слова не превосходит 10 символов.

Формат выходного файла

Если решение существует, в выходной файл выведите ровно n строк, где i -ая строка является сокращенным аналогом i -го слова исходного набора. Если решений несколько, выведите любое из них. Если решения не существует, выведите -1.

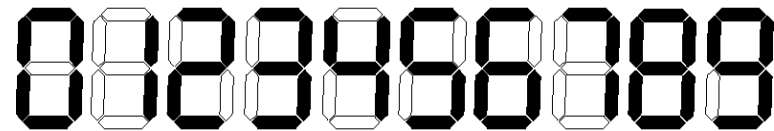
Примеры

input.txt	output.txt
6 privet spasibo codeforces java marmelad normalno	pret sps cdfs java mama norm
5 aaa aa a aaaa aaaaa	-1

Задача I. Счастье в числах

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 second
Ограничение по памяти: 256 megabytes

Вася давно собирает счастливые билеты. В его коллекции уже насчитывается несколько тысяч счастливых трамвайных, троллейбусных и автобусных билетов. Васе уже надоело стандартное понимание того, является ли билет счастливым. Поэтому он ищет новые понимания. Кроме того, Васе непонятно, почему все билеты делятся только на счастливые и несчастливые. Он считает, что все билеты счастливые, только в разной степени. Немного поразмыслив, Вася пришел к определению *счастливости* билета. Пусть билет состоит из $2n$ цифр. Представим, что каждая из этих цифр записана так, как это показано на рисунке:



Цифры такого вида вы наверняка видели на электронных часах: для отображения цифры используются семь сегментов, каждый из которых может быть подсвечен или нет; подсвеченные сегменты образуют цифру. Представив, что цифры записаны именно так, Вася берет правую половину билета и накладывает ее на левую, совмещая первую цифру с $n + 1$ -й, вторую с $n + 2$ -й, ..., n -ю с $2n$ -й. Для каждой пары совмещенных цифр он подсчитывает, сколько сегментов подсвечены как в одной цифре, так и в другой, и суммирует эти количества. Полученную величину он называет *счастливостью* билета. Например, счастливость билета 03 равна четырем, а счастливость билета 2345 — шести.

По заданному номеру билета, состоящему из $2n$ цифр, найдите наименьший по номеру билет среди тех билетов, чей номер больше данного, но состоит также из $2n$ цифр, и чья счастливость больше, чем счастливость данного билета.

Формат входного файла

В первой строке записан номер билета, состоящий из k символов ($k = 2n, 1 \leq n \leq 10^5$).

Формат выходного файла

Выведите номер искомого билета или «-1» (без кавычек), если такого билета не существует.

Примеры

input.txt	output.txt
13	20
2345	2348
88	-1

Задача J. Минимальная сумма

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Вам задан набор из n векторов на плоскости. Для каждого вектора разрешается домножить любые его координаты на -1 . Таким образом, каждый вектор $v_i = (x_i, y_i)$ можно преобразовать в один из следующих четырех векторов:

- $v_i^1 = (x_i, y_i)$,
- $v_i^2 = (-x_i, y_i)$,
- $v_i^3 = (x_i, -y_i)$,
- $v_i^4 = (-x_i, -y_i)$.

Вам нужно найти два вектора из набора и определить, какие из их координат следует умножить на -1 таким образом, чтобы модуль суммы полученных векторов был минимально возможным. Более формально, требуется выбрать два вектора v_i, v_j ($1 \leq i, j \leq n, i \neq j$) и два числа k_1, k_2 ($1 \leq k_1, k_2 \leq 4$), так чтобы значение выражения $|v_i^{k_1} + v_j^{k_2}|$ было минимально.

Формат входного файла

В первой строке записано одно целое число n ($2 \leq n \leq 10^5$). Далее в n строках записаны вектора в виде пар целых чисел « $x_i y_i$ » ($-10000 \leq x_i, y_i \leq 10000$), по одной паре в строке.

Формат выходного файла

Выведите в первой строке четыре числа через пробел « $i k_1 j k_2$ » — ответ на задачу. Если существует несколько вариантов с минимальным модулем суммы, можете вывести любой из них.

Примеры

input.txt	output.txt
5 -8 -3 -9 -6 -4 9 5 3 9 3	5 1 1 1
5 -3 -8 -9 10 -9 0 -9 -2 -6 5	4 4 3 3

Note

Суммой двух векторов $v = (x_v, y_v)$ и $u = (x_u, y_u)$ называется вектор $s = v + u = (x_v + x_u, y_v + y_u)$.

Модулем вектора $v = (x, y)$ называется число $|v| = \sqrt{x^2 + y^2}$.

Во втором примере существует несколько подходящих ответов, вот некоторые из них: (3 1 4 2), (3 1 4 4), (3 4 4 1), (3 4 4 3), (4 1 3 2), (4 1 3 4), (4 2 3 1).