

Задача А. Научная проблема

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Однажды профессор Чегобымнеещепридумать ехал в поезде. Наблюдая унылый пейзаж за окном, он решил придумать тему своей новой научной работы. И тут в его голову внезапно пришла великолепная идея: придумать эффективный алгоритм нахождения такого натурального числа, которое в x раз меньше суммы всех ему предшествующих натуральных чисел, где x — заданное число. Поскольку в поезде у него нет компьютера, чтобы проверить свою идею, эта сложная задача ложится на ваши плечи.

Формат входного файла

В единственной строке входного файла записано целое число x ($1 \leq x \leq 10^9$).

Формат выходного файла

Выведите одно число — ответ на поставленную задачу.

Пример

input.txt	output.txt
1	3
2	5

Пояснение к первому тесту: числа 1 и 2 — это числа, которые предшествуют 3, кроме того верно равенство $(1 + 2) = 3 * 1$.

Задача В. Гадание на ромашке

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Девочка Маша влюблена в Петю. Ей не даёт покоя вопрос, испытывает ли Петя к ней ответные чувства. Лучший способ узнать — это гадание на любовь. Существует множество различных способов гадания, однако Маше больше всего понравилось гадание на ромашке.

Правила гадания следующие. Надо взять в правую руку ромашку, и обрывать лепестки левой рукой, приговаривая заранее задуманные фразы типа «любит», «не любит», «любит искренне», «сомневается», «ждёт свидания», «насмехается». Фразы произносятся в заранее выбранном порядке начиная с первой по кругу, то есть после последней фразы снова произносится первая. Фраза, произнесённая в тот момент, когда срывается последний лепесток, и будет ответом.

Поскольку Маше не хочется идти в лес и искать ромашки, она просит Вас написать программу, выполняющую гадание на ромашке. Программе будет подаваться на вход количество лепестков ромашки N , количество фраз M и сами фразы.

Формат входного файла

В первой строке записаны натуральные числа N и M ($1 \leq N \leq 100$, $1 \leq M \leq 100$).

Далее идут M строк, в каждой из которых записано по одной фразе. Фраза — это слово длиной не менее одного и не более ста символов, состоящее из латинских букв.

Формат выходного файла

Выведите в единственной строке выходного файла фразу — результат гадания.

Пример

input.txt	output.txt
6 4 loves doesnt sincerely doubts	doesnt
9 3 loves doesnt hates	hates

Задача С. Тотализатор

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Недавно Вася, разбирая хлам на чердаке, наткнулся на записи тотализатора, который его дедушка с друзьями организовал на результаты XII чемпионата Берляндии **** года по художественному футболу. Из записей следовало, что каждый участник тотализатора сделал свой прогноз на результат каждого матча (результатом матча, как и в обычном футболе, является счет — число голов, забитых первой и второй командами). В качестве прогноза также выступает счет. Каждый участник получает очки за каждый матч в соответствии со сделанным прогнозом. Правила подсчета в те времена были такие:

- если участник угадал исход (победителя или ничью), то к его сумме прибавляется 2 очка.
- если участник угадал разницу мячей (разность между числом голов у первой и второй команды), то к его сумме прибавляется 3 очка.
- если участник угадал число мячей у первой команды, то к его сумме прибавляется 1 очко.
- если участник угадал число мячей у второй команды, то к его сумме прибавляется 1 очко.

Таким образом, если участник угадал точный исход матча, то он получит 7 очков, а если, к примеру, только исход и разницу, то 5 очков. К сожалению, листок с результатами тотализатора оказался утерян. Но Васе так хочется узнать, сколько же очков набрал каждый участник и так не хочется считать самому...

Формат входного файла

В первой строке записано два целых числа n и m ($1 \leq n \leq 100$, $1 \leq m \leq 100$) — число участников и число матчей. Далее следуют m блоков по $n + 1$ строке, по одному блоку на каждый матч. В первой строке каждого блока содержатся два целых неотрицательных числа, не превосходящие 10 — количество голов, забитых первой и второй командой соответственно. В остальных n строках заданы прогнозы всех участников в таком же формате, i -я строка задает прогноз i -го участника.

Формат выходного файла

В первой строке выведите n чисел через пробел — количества очков каждого участника. Результаты выводите в порядке возрастания номера участника. Участники нумеруются в порядке их описания во входных данных.

Пример

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

Задача D. Угол

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Все знают поисковую систему Угол, однако мало кому известно, что одной из ее возможностей является поиск числовых последовательностей. Любой пользователь, введя

набор натуральных чисел, с ее помощью может найти все числовые последовательности, содержащие этот набор чисел. Разработчики Угла решили усовершенствовать поисковую систему, добавив новую возможность. Если пользователю известны не только те числа, которые входят в искомую последовательность, но и числа, которые в нее точно не входят, то он должен ввести их с противоположным знаком. Например, введя запрос «5 -3 6», пользователь найдет все числовые последовательности, содержащие числа 5 и 6, но не содержащие число 3. Помогите разработчикам усовершенствовать поисковую систему.

Формат входного файла

В первой строке входного файла записаны два целых числа n и m ($1 \leq n \leq 10$, $1 \leq m \leq 10$) — количество числовых последовательностей в базе данных Угла и количество запросов пользователей соответственно. Далее в n строках описываются последовательности чисел. Первое число в строке задает количество чисел k ($1 \leq k \leq 10$) в последовательности. Далее следует k целых чисел a_i ($1 \leq a_i \leq 100$): числа последовательности. Следующие m строк описывают запросы пользователей. Первое число в строке задает количество чисел l ($1 \leq l \leq 10$) в запросе. Далее следует содержимое запроса, l различных по модулю целых чисел b_i ($1 \leq |b_i| \leq 100$, $b_i \neq 0$).

Формат выходного файла

Для каждого запроса выведите на отдельной строке количество последовательностей удовлетворяющих запросу. Далее выведите сами последовательности в том же порядке, как они заданы во входном файле. Выводите последовательности в том же формате, что они заданы во входном файле. Каждая последовательность должна быть выведена на отдельной строке. Порядок чисел в последовательностях должен быть сохранен.

Пример

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

Задача E. Число путей в империи

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Во времена правления династии Цамь люди редко враждовали между собой и с

соседями: у них попросту не было на это времени, ибо они всецело были поглощены решением разных задач. А решали они задачи торговли, ремесла и земледелия, но не только.

Среди обширного списка задач стоят особняком задачи сбора налогов с городов империи. Одну из них придумал император, и по этой причине она имеет государственное значение. Задача императора состоит в том, чтобы посчитать количество различных путей, состоящих из m дорог. Каждый путь должен начинаться и заканчиваться в столице империи. Пути могут неоднократно проходить через одни и те же города и дороги. Более того, они могут неоднократно проходить через столицу империи.

Решите эту задачу, зная, что в империи n провинциальных городов расположены у подножия холма, на вершине которого находится столица. Города расположены вдоль окружности (основания холма). Столица соединена дорогами со всеми городами, а провинциальные города соединены с двумя соседними (кроме столицы).

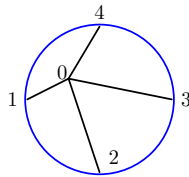


Рис. 1 Империя из столицы (номер 0) и четырех провинциальных городов (номера 1 – 4).

Формат входного файла

В единственной строке входного файла содержатся два целых числа n и m ($3 \leq n \leq 1000$, $0 \leq m \leq 5000$).

Формат выходного файла

Выведите ответ без лидирующих нулей.

Пример

input.txt	output.txt
4 3	8
3 4	21

Пояснение к первому примеру. В империи существует 8 путей: 0-1-2-0, 0-1-2-3-0, 0-3-4-0, 0-4-1-0, 0-2-1-0, 0-3-2-0, 0-4-3-0, 0-1-4-0.

Задача F. Игра в точки

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Недавно Петя открыл для себя новую игру в точки. Правила игры просты. Сначала

имеется только одна точка A_0 с координатами $(0, 0)$. Затем Петя должен поставить еще N точек. Точки нужно ставить последовательно, причем новая точка должна соединяться отрезком ровно с одной из предыдущих. Опишем процесс более формально. На i -м ходу сначала выбирается позиция для точки A_i , не обязательно с целыми координатами. Затем фиксируется та из имеющихся точек, с которой A_i будет соединена отрезком. Пусть это точка B . Должны выполняться следующие условия:

1. Точка A_i не должна совпадать с какой-либо из предыдущих.
2. Точка A_i не должна лежать на ранее проведенных отрезках.
3. Отрезок A_iB не должен пересекать ранее проведенные отрезки нигде, кроме точки B .
4. Отрезок A_iB не должен покрывать ранее поставленные точки, кроме точки B .
5. Длина отрезка A_iB не должна превосходить 1.

После постановки каждой новой точки считаются 2 величины.

1. Наибольшее количество отрезков, имеющих общую точку.
2. Наибольшее декартово расстояние между любой парой построенных точек.

В конце каждого хода Петя получает количество очков, равное произведению этих двух величин.

Найдите, какое максимальное количество очков сможет набрать Петя за всю игру.

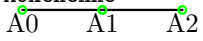
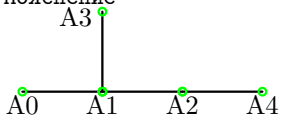
Формат входного файла

Во входном файле содержится единственное целое число N ($0 \leq N \leq 1000$).

Формат выходного файла

Выведите максимальное количество очков, которое сможет набрать Петя с точностью до 3 знаков после десятичной точки.

Пример

input.txt	output.txt
2	5.000 Пояснение  На первом ходу Петя получит одно очко, а на втором 4 очка.
4	20.000 Пояснение  На первом ходу Петя получит одно очко, на втором 4 очка, на третьем 6 очков, а на четвертом 9 очков.

Задача G. Берляндский флаг

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В Берляндии снова проблемы. После последних выборов Всеберляндская Великая Дума разделилась на две коалиции: коалицию Синеглазых и коалицию Красноглазых. В день независимости Берляндии в повестку дня в Думе был внесен важный вопрос: как будет выглядеть государственный флаг страны?

Обе коалиции сошлись во мнении, что флаг должен представлять собой квадрат размером $N^2 \times N^2$ клеток, каждая из которых будет либо красная, либо синяя. Чтобы обе коалиции согласились принять флаг и узаконить его, он должен соответствовать следующим правилам:

1. в любой строчке должно быть ровно K синих клеток
2. в любом столбце должно быть ровно K синих клеток
3. если разрезать флаг на квадраты размером $N \times N$ клеток (которых $N \times N$ штук), то любой полученный квадрат должен содержать ровно K синих клеток.

Вы являетесь главными программистами в Берляндии. Разработка макета флага — это ваша задача, выполните же ее достойно!

Формат входного файла

В единственной строке задано два целых числа N и K ($1 \leq N \leq 20$, $0 \leq K \leq N^2$).

Формат выходного файла

Выходной файл должен содержать описание искомого флага. Выведите N^2 строчек, состоящих из N^2 символов '*' и '.', где '*' обозначает клетку синего цвета, а '.' — красного. Выведите фразу «NO SOLUTION» (без кавычек), если не существует флага, который согласились бы принять обе коалиции. Если решений несколько, то выведите любое.

Пример

input.txt	output.txt
2 2	*.* .** .** *.*
3 1	*..... ...*....*.. .*..... ...*....*. ..*.....*..*

Задача H. Галактика в опасности

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Прошло уже немало лет с тех пор, как на галактику Логос напали мелкие злобные космические существа мисткаферы. Ценой невероятных усилий и больших потерь отважным защитникам галактики удалось разбить основные силы противника на несколько изолированных друг от друга групп. Враги были локализованы на N различных планетах.

Теперь перед Правительством галактики стоит непростая задача — как можно быстрее избавиться от мисткаферов в галактике, а именно вывезти их далеко за пределы галактики и сбросить в черную дыру. Чтобы это сделать, Правительству понадобится помощь крылатых пфердов, которые способны летать через черные дыры.

Так совпало, что у Правительства как раз имеются в распоряжении N пфердов. Каждый пферд приписан к одной из N планет, а к каждой планете приписан ровно один пферд. В течение дня пферды летят к своим планетам и забирают оттуда по одному мисткаферу.

Они настолько дружны, что все делают одновременно и летают только вместе. Поэтому, если на какой-то планете не останется мисткаферов, все пферды откажутся вывозить мисткафер и с других планет.

Во избежание вышеописанной проблемы Правительство может отправить экспедицию ученых на одну из планет и с помощью клонирования увеличить численность мисткаферов на ней вдвое. Экспедиция занимает один день и в этот день пферды не вылетают.

Выясните, какое наименьшее количество дней понадобится Правительству Логоса для избавления от мисткаферов.

Формат входного файла

В первой строке входного файла записано количество планет N ($1 \leq N \leq 100000$). Во второй строке следуют N натуральных чисел, каждое из которых задает количество мисткаферов на соответствующей планете. В начальный момент количество мисткаферов на каждой планете не превосходит одного миллиарда.

Формат выходного файла

В первой строке выходного файла выведите одно число K — ответ на задачу. В случае, если количество дней не превосходит тысячи, в следующих K строках выведите описание действий в хронологическом порядке. Если в i -й день был произведен вылет пфердов, то в $(i + 1)$ -ой строке выведите фразу «flying mission» (без кавычек), в противном случае выведите фразу «science mission to the planet j » (без кавычек), где j — номер планеты, на которой производилось клонирование. Планеты следует нумеровать с 1.

Пример

input.txt	output.txt
2 1 2	3 science mission to the planet 1 flying mission flying mission
3 1025 2 1025	1035

Задача I. Герой Петя

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

После возвращения с Берляндско-Бирляндской войны Петя любит собирать друзей и рассказывать им о своих подвигах. Может быть, и вы уже слышали историю о том, как Петя, оказавшись в окружении, в одиночку взял в плен двух бирляндских офицеров, воспользовавшись их одеждой, и, выведав пароль, проник на военную базу противника, вывел из строя систему управления и таким образом помог берляндской армии захватить базу.

Эта история дошла и до полковника Кругляковского, которого особенно заинтересовала одна деталь. А именно, как Петя смог узнать пароль для входа на военную базу, совершенно не владея бирляндским языком.

Вызванный к полковнику юный герой объяснил, что хотя бирляндская речь ему непонятна, ее несложно записать. Сначала Петя допросил пленных офицеров, записав речь каждого из них в виде строки латинских букв. Он знал, что бирляндские пароли читаются одинаково слева направо и справа налево, т.е. являются палиндромами.

Поэтому осталось воспользоваться программой, находящей в двух строках общую подстроку наибольшей длины, подходящую в качестве пароля. Выслушав рассказ, полковник Кругляковский заявил, что такая программа может быть очень полезна при допросе бирляндцев и при расшифровке их засекреченных сообщений...

Так как никакой программы у него и не было, Петя обратился к вам.

Формат входного файла

Во входном файле содержатся две непустые строки, содержащие только строчные латинские буквы ('a'-'z'). Длина каждой строки не превосходит 2000 символов. Строки имеют хотя бы одну общую букву.

Формат выходного файла

Выведите пароль, полученный с помощью описанной Петей программы. Если возможных ответов несколько, выведите любой.

Пример

input.txt	output.txt
abacaba abracab	aca
abbab babbab	abba

Задача J. Экспедиция

Имя входного файла: input.txt
Имя выходного файла: output.txt
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вот и снова лето! Петя долго его ждал. Неудивительно, ведь он едет в первую в своей жизни настоящую геологическую экспедицию. А что в экспедиции самое главное? Ну, конечно же, большая просторная палатка! Петя заметил, что в разложенном состоянии палатка представляет собой выпуклый многоугольник с N вершинами (вид сверху). Несомненно, в настоящей геологической экспедиции одной палаткой не обойдешься, нужно взять с собой еще много полезных вещей — лопаты, котелок, долото, приборы, гитару, в конце концов! Все это решено было разместить внутри палатки на M стеллажах. Каждый стеллаж достаточно узкий, а значит, на плане палатки его можно представить как некоторый отрезок. Стеллажи бывают разной конструкции, поэтому

соответствующие им отрезки могут касаться друг друга или даже пересекаться. В центре палатки, на полу установили светильник. На плане ему соответствует точка $(0,0)$. Петя заметил, что стеллажи загораживают свет, поэтому ночью часть стен палатки остается неосвещенной. Он вспомнил, что бросил свои сапоги на пол к стене палатки, поэтому его заинтересовал насущный вопрос: какова суммарная длина неосвещенных участков стен палатки. Помогите Пете.

Формат входного файла

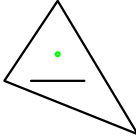
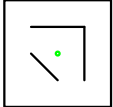
В первой строке входного файла содержатся два целых числа N и M ($3 \leq N \leq 100000$, $0 \leq M \leq 100000$) — количество вершин в многоугольнике, которым характеризуется палатка, и количество стеллажей соответственно.

В каждой из следующих N строк содержится по 2 числа (x_i, y_i) — координаты i -й вершины многоугольника в порядке обхода против часовой стрелки. Никакие две вершины многоугольника не совпадают. Далее идет M четверок чисел $(x_{1j}, y_{1j}) (x_{2j}, y_{2j})$ — координаты концов отрезков, представляющих стеллажи. Все координаты целые и не превосходят 10^6 по абсолютной величине. Гарантируется, что точка $(0,0)$, а также все отрезки лежат строго внутри многоугольника. Ни один из отрезков не проходит через центр палатки.

Формат выходного файла

Выведите ответ на задачу с точностью не менее 6 знаков после запятой.

Пример

input.txt	output.txt
<pre>3 1 0 2 -2 -1 3 -3 -1 -1 1 -1</pre>	<pre>4.615855548972 Пояснительный рисунок:</pre> 
<pre>4 3 -2 -2 2 -2 2 2 -2 2 -1 0 0 -1 1 -1 1 1 -1 1 1 1</pre>	<pre>12.000000 Пояснительный рисунок:</pre> 

Задача К. Развал Берляндского союза

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Всем известно, что Берляндия состоит из четного количества городов, соединенных дорогами. От каждого города в Берляндии можно добраться до любого другого. Между двумя городами не более одной дороги, а также нет дорог, соединяющих город с самим собой. Однако в силу назревших конфликтов Берляндия уже не может существовать как одна страна. Мэры городов разругались друг с другом и не хотят жить под одной крышей. На общем собрании было решено создать Берляндский союз, разбив Берляндию на несколько стран. Было решено, что в каждую страну будет входить как минимум два города, поскольку ни один город не сможет жить самостоятельно без помощи и ресурсов других городов. Кроме этого между любыми двумя городами одной страны должен существовать путь, не выходящий за пределы страны, по дорогам Берляндского союза, при этом только один (это решение принято для уравнивания прав всех стран). Помогите мэрам разделить Берляндию на страны, чтобы все их требования были выполнены. Заметьте, что в этой задаче не требуется минимизировать количество стран.

Формат входного файла

В первой строке записано количество тестов во входном файле tst ($1 \leq tst \leq 50$). Далее следует описание тестов. В первой строке, описывающий тест, заданы целые числа n и m ($2 \leq n \leq 100$, $1 \leq m \leq 1000$, n — четное) — количество городов и количество дорог. Далее в m строках заданы дороги парами номеров городов a и b ($1 \leq a < b \leq n$). Сумма количеств городов во всех тестах не превосходит 100, а сумма количеств дорог — 1000.

Формат выходного файла

Для каждого теста на отдельной строке выведите n чисел. i -ое число — номер страны, которой принадлежит i -ый город. Страны нумеруются с единицы подряд, то есть от 1 до k , где k — количество стран в вашем плане. Если решений несколько, выведите любое.

Пример

input.txt	output.txt
2	1 1 1 1
4 3	1 2 2 1
1 2	
2 3	
1 4	
4 4	
1 2	
2 3	
1 4	
1 3	