

Задача 1. Сколково.

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Строители Сколково побывали в Новосибирском Академгородке и заметили, что на здании ВЦ год постройки изображен в двоичной системе счисления. Это натолкнуло их на идею изобразить номера домов Сколково в двоичной системе счисления. Очевидно, что для этого таблички нужно сделать больше обычных. Помогите строителям определить длину строки символов, необходимую для записи в двоичной системе счисления заданного числа.

Входные данные

В первой строке задано число N ($0 \leq N \leq 1\,000\,000\,000$).

Выходные данные

В единственной строке одно целое число – искомая длина строки.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
1	1

Задача 2. Четыре числа

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Определим следующую операцию:

$$f(x_1, x_2, x_3, x_4) = f^1(x_1, x_2, x_3, x_4) = (x_1 + x_2, x_2 + x_3, x_3 + x_4, x_4 + x_1) / 2,$$

$$f^n(x_1, x_2, x_3, x_4) = f(f^{n-1}(x_1, x_2, x_3, x_4)), n \geq 2, n - \text{натуральное число},$$

$$f^0(x_1, x_2, x_3, x_4) = (x_1, x_2, x_3, x_4).$$

По заданным целым числам x_1, x_2, x_3, x_4 определить максимально возможное число n такое, что $f^n(y_1, y_2, y_3, y_4) = (x_1, x_2, x_3, x_4)$, причем все числа, входящие в векторы $f^k(y_1, y_2, y_3, y_4)$, k – произвольное число от 0 до n включительно, – целые.

Входные данные

В первой и единственной строке 4 целых числа – x_1, x_2, x_3, x_4 , по модулю не превосходящие 1000.

Выходные данные

В первой и единственной строке одно целое неотрицательное число n – максимальное число, удовлетворяющее условию задачи. Если максимума не существует, вывести INF.

Пример

Входные данные	Выходные данные
0 0 1 1	1

Задача 3. Совет племени

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Чтобы избежать конфликтов с иноземцами, племя индейцев переселилось в глубь джунглей. Их поселок расположился в непроходимых зарослях на поляне, имеющей форму многоугольника. Вокруг поляны росли высокие деревья, укрывавшие своей зеленью индейское поселение, а внутри поляны деревьев не было вообще. Тем не менее, чтобы повысить безопасность, индейцы расставили по периметру охрану — в каждый угол поляны они поставили по опытному, хорошо вооруженному индейцу. Более того, несколько менее опытных индейцев расположились в произвольных местах границы. Всем индейцам, выставленным для охраны племени, было строго-настрого запрещено покидать свой пост.

Спустя некоторое время вождь решил собрать совет племени. Для этого на территории поселения нужно было выбрать участок, представляющий собой многоугольник, удовлетворяющий следующему условию — любые два индейца, находящиеся внутри или на границе участка, должны видеть друг друга, то есть соединяющий их отрезок прямой не должен пересекать территорию джунглей, на которой есть деревья.

В то же время вождь считал, что в совете племени должно принять участие как можно больше индейцев, поэтому территорию хотел выбрать так, чтобы в нее попало наибольшее число индейцев, выставленных по периметру поселения. Помогите вождю определить, насколько полным будет совет племени, если выбрать участок для совета самым лучшим образом.

Входные данные

В первой строке одно натуральное число N , $3 \leq N \leq 60$ – количество индейцев, охраняющих территорию поселения. Далее N строк по два целых числа – координаты индейцев в порядке границы территории против часовой стрелки. Каждая из координат по модулю не превышает 100 000.

Выходные данные

В первой и единственной строке одно натуральное число – максимально возможное число индейцев, охраняющих территорию, которые смогут принять участие в Совете племени.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
10 1 1 2 2 3 3 6 3 7 7 5 6 5 5 3 5 2 4 1 4	6

Задача 4. Индейская головоломка

Время на тест: 1 секунда

Ограничение по памяти 256Мб

По вечерам, сидя у костра, индейцы решали различные головоломки. Вот одна из них.

На клетчатой доске из трёх рядов по N клеток в первом ряду располагается N камней с номерами от 1 до N по одному в каждой клетке слева направо. Требуется передвинуть все эти камни в третий ряд согласно следующим правилам:

- За один ход один камень можно передвинуть на одну клетку вперед или назад,
- Нельзя перемещать камень на ряд, в котором левее уже есть камень,
- Нельзя перемещать камень с ряда, в котором левее есть камень.

Индейцы могли решить эту головоломку даже для $N = 100$. От Вас же требуется намного меньше — определить в зависимости от размеров поля, какое минимальное количество ходов необходимо для решения головоломки, и вывести решение только в том случае, если количество ходов невелико.

Входные данные

В первой строке и единственной строке одно натуральное число N – длина поля, $1 \leq N \leq 25$.

Выходные данные

В первой строке одно натуральное число M – минимальное количество ходов, необходимое для решения головоломки.

Если число $M < 3000$, то вывести самую оптимальную последовательность действий в следующем формате:

- Описание каждого хода в отдельной строке,
- Перемещение i -го камня вперед описывается строкой “ $i F$ ”, назад - “ $i B$ ”.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
2	8 1 F 1 F 2 F 1 B 1 B 2 F 1 F 1 F

Задача 5. Бранное имя

Время на тест: 2 секунды

Ограничение по памяти 256Мб

Индейцы имели свой язык и письменность. Их алфавит состоял из нескольких латинских букв, которыми они владели в совершенстве. Совершенство было настолько полным, что из этих букв они даже составили нецензурные трехбуквенные слова. Эти слова были очень нужны в повседневной жизни — на охоте, рыбалке, при собирательстве, однако, вместе с ними появились дополнительные проблемы.

Например, возникла проблема составления имен. Каждый приличный индеец хотел себе имя подлиннее и непременно жаждал выцарапать его на Скале Памяти, чтобы оставить след в истории. Но что, если некоторые буквы в имени сотрутся, а оставшийся фрагмент будет представлять собой нецензурное слово? Такого нельзя было допустить, ведь к Скале Памяти ходят и дети. Поэтому вождь племени решил посчитать, сколько различных имен заданной длины можно составить таким образом, чтобы в них не встречалось ни одного нецензурного слова.

Входные данные

В первой строке алфавит языка индейцев — одна строка из различных строчных латинских букв.

Во второй строке два натуральных числа N и L — количество нецензурных слов ($0 \leq N \leq 10$) и длина имени ($1 \leq L \leq 1\,000\,000$).

Далее N строк, в каждой из которых по одному нецензурному слову из трех букв.

Выходные данные

Одно целое неотрицательное число M — количество имен длины L , не содержащих нецензурные слова. Считается, что имя содержит нецензурное слово, если в нем есть три последовательные буквы, составляющие одно из нецензурных слов.

Важно отметить, что считать индейцы умели только до 1000, потому число M нужно вывести по модулю 1000.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
ab 2 4 abb bab	9

Задача 6. Испытание

Время на тест: 8 секунд

Ограничение по памяти 256Мб

Индийское племя не отличалось особой жестокостью, поэтому даже у пленных был шанс выжить. Для этого им нужно было пройти следующее пустынное испытание.

Пленного выводили на большое прямоугольное поле. По периметру этого поля, начиная с левого нижнего угла, через каждый метр стояли индейцы, вооруженные луком и стрелами. Пленному нужно было всего лишь продержаться 10 минут внутри этого поля, и он был свободен.

К величайшему сожалению, индейцам никак не удавалось проявить великодушие и отпустить пленного на свободу. Никто не мог продержаться и минуты. Тогда они, будучи очень добрыми (правда, внутри), решили упростить пленным задачу и расставить на поле плоские загородки-стенки. Теперь пленный мог просто найти точку, в которой его не видит ни один индеец, и сидеть там хоть целый час. “Слишком уж это просто”, — решили индейцы и придумали следующее - выпустить на поле своих собак, которые не дадут пленному отсиживаться где попало. От собак же он сможет укрыться только в заранее расставленных по полю клетках. Клеток индейцы поставили много — на пересечение каждой пары отрезков, параллельных сторонам поля и проходящих через индейцев, стоящих по периметру. Теперь им стало интересно, сколько клеток действительно являются спасительными для пленного.

Входные данные

В первой строке три натуральных числа M, N, K , $2 \leq M, N \leq 500$, $0 \leq K \leq 20$ – размеры поля с севера на юг и с запада на восток и количество стенок на поле, соответственно. Далее K строк по 4 натуральных числа XS, YS, XF, YF – координаты начала и конца стенок ($1 \leq XS, XF \leq N - 1$, $1 \leq YS, YF \leq M - 1$), стенки представляют собой отрезки прямых.

Левый нижний угол поля имеет координаты $(0, 0)$, а правый верхний — (N, M) . Все координаты в метрах.

Выходные данные

В первой и единственной строке одно целое неотрицательное число – количество точек поля с целыми координатами таких, что отрезок, соединяющий каждую из этих точек с произвольной точкой границы поля, также имеющей целые координаты, имеет общую точку хотя бы с одной стенкой.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
6 10 7 2 5 9 4 1 4 1 1 1 1 3 1 2 2 6 2 2 3 3 4 5 1 7 3 8 2 8 4	3

Задача 7. Охотники за сокровищами

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Группа, отправленная на поиски сокровищ древнего индейского племени, долго блуждала по джунглям в поисках места, где было расположено индейское поселение. Когда надежды найти хоть какие-то следы древней цивилизации были практически утрачены, они внезапно вышли на небольшую поляну, посреди которой был расположен колодец. Спустившись вниз по веревочной лестнице, они оказались в темном подземелье. Оглядевшись, на одной из стен в свете факелов они увидели план лабиринта.

План представлял собой клеточное поле, в каждой клетке которого было по одному символу — либо кружок, либо палочка. При этом два кружка отличались от остальных — в одном был нарисован человечек, а в другом — еще один кружок. Посоветавшись, искатели решили, что кружки обозначают свободное пространство, палочки — стены лабиринта, кружок с человечком — их текущее положение в лабиринте, а кружок с кружком — место, где хранятся сокровища.

Казалось бы, теперь им оставалось проложить путь по лабиринту — и цель достигнута. Но все было не так просто — было непонятно, можно ли дойти до сокровищ, не пройдя сквозь стены. Сквозь стены ходить, конечно же, никто не умел. К счастью, у искателей был набор инструментов, потому они могли проделать лаз в стенах. Однако для того, чтобы сделать один участок стены проходимым, требовалось много сил и времени. В связи с этим возникла необходимость определить, какое минимальное количество стен нужно разрушить, чтобы пройти от текущего положения до места, где спрятаны сокровища.

Входные данные

В первой строке два натуральных числа M и N , $2 \leq M, N \leq 1000$ – размеры лабиринта с севера на юг и с запада на восток. Далее M строк по N символов (1 или 0) – план лабиринта (1 – стена, 0 – свободное пространство). В следующей строке 4 натуральных числа XS , YS , XF , YF – координаты клетки с человечком и клетки с двойным кружком соответственно ($1 \leq XS, XF \leq M$, $1 \leq YS, YF \leq N$).

Ходить по лабиринту можно только в четырех направлениях: север — юг, запад — восток.

Выходные данные

В первой и единственной строке одно целое неотрицательное число – минимальное количество стен, которое необходимо разрушить.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
6 6 011000 110000 111111 111111 110000 000000 6 1 1 1	4

Задача 8. Игры с числами.

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Ходит легенда, что Фибоначчи придумал свой ряд чисел, наблюдая за размножением кроликов. Напомним, что каждое следующее число Фибоначчи является суммой двух предыдущих, и классическая последовательность чисел Фибоначчи 1, 1, 2, 3, 5, 8, 13 ..., по легенде, описывает количество кроликов в каждом следующем поколении. Эта последовательность полностью определяется двумя первыми членами. Один из последователей Фибоначчи решил придумать свою последовательность. Он решил посчитать не количество кроликов, а их прирост. То есть, если f_i – последовательность из n первых чисел Фибоначчи, то i -й элемент новой последовательности из $n-1$ чисел s_i определяется как $s_i = f_{i+1} - f_i$. Получив новую последовательность, новатор задумался «А что будет, если посчитать прирост прироста кроликов? А если прирост прироста прироста?» Вычисление последовательностей ему надоело довольно быстро, и он решил получить «суперчисло», которое останется, если применять операцию вычисления разностей соседних членов $n - 1$ раз! Обозначим через $s(k)$ последовательность из $n-k$ чисел, полученных применением операции вычисления разностей соседних членов k раз. То есть

$$s(1)_i = f_{i+1} - f_i, s(2)_i = s(1)_{i+1} - s(1)_i, \dots, s(k)_i = s(k-1)_{i+1} - s(k-1)_i \dots$$

Тогда последовательность $s(n-1)$ состоит из одного числа, которое математик-новатор назвал «суперчисло». Далее он решил изучать свойства этого «суперчисла» и начал с определения его делимости на целые числа. Но тут пришла мама и отправила математика спать... Помогите математику провести исследования его «суперчисла».

Входные данные

В первой строке четыре натуральных числа n, k, p, q , разделенные пробелами. n – длина начальной последовательности чисел Фибоначчи ($2 \leq n \leq 1\,000\,000$), p и q первые два числа в последовательности ($|p| \leq 1\,000\,000\,000$, $|q| \leq 1\,000\,000\,000$), k – делитель ($2 \leq k \leq 1\,000\,000\,000$).

Выходные данные

В единственной строке одно целое число – «суперчисло» по модулю k (число a по модулю b определено как минимальное неотрицательное число $m < b$, для которого $a = b + mt$ для некоторого целого t).

Пример

Входные данные	Выходные данные
7 10 1 1	5

Задача 9. Подготовка к ремонту.

Время на тест: 1 секунда

Ограничение по памяти 256Мб

На предприятии, где Вы работаете Главным программистом, собрались делать ремонт системы отопления. Для этого отдел снабжения заказал трубы нескольких диаметров. Однако поставщики немного ошиблись и поставили не все необходимые элементы. Главный слесарь просит Вас решить следующую проблему (непонятные слесарские термины заменены многоточием).

- У меня есть N ... труб с известными диаметрами (диаметры ... труб – целые положительные ... числа), из которых я должен собрать одну ... длинную ... трубу. По ... техническому ... регламенту я должен соединить их в порядке возрастания ... диаметров, при этом для соединения ... труб разного ... диаметра нужны ... специальные ... переходники, поскольку у меня есть только ... муфты для соединения двух ... труб одинакового ... диаметра. Эти ... переходники бывают только на соседние (отличающиеся ровно на 1) диаметры. Если разница ... диаметров двух соседних ... труб больше 1, то нужно ставить несколько ... переходников подряд (при этом число ... переходников, естественно, равно разности диаметров). ... Переходников наши снабженцы не закупили. Посчитай мне, пожалуйста, сколько ... переходников надо заказать.

Входные данные

В первой строке задано число труб N ($0 < N \leq 10\,000$). В следующих N строках по одному положительному целому числу – диаметры труб (не превышающие 1 000 000 000).

Выходные данные

В единственной строке одно целое число – общее число необходимых переходников.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
2 100 1	99

Задача 10. Инвентаризация на складе.

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Вы работаете на складе и проводите инвентаризацию (сверяете наклейки на коробках с ведомостью). Коробки стоят высоко, и наклейки верхних коробок Вам не видно. Поэтому Вы решили поставить 2 коробки друг на друга и забраться на них, чтобы прочитать надписи. Коробки нужно поставить так, чтобы они стояли устойчиво, то есть, чтобы основание верхней коробки не выходило за границы нижней. Коробки – прямоугольные параллелепипеды. Коробки тяжелые, перемещать их можно только автопогрузчиком. Автопогрузчик не может работать, если коробка стоит криво, поэтому все стороны двух коробок должны быть либо параллельны, либо перпендикулярны. Напишите программу, которая для двух заданных коробок определяет, какую максимальную суммарную высоту можно из них составить.

Входные данные

В первой строке заданы 6 положительных целых чисел, разделённых пробелами: 3 измерения первой коробки и 3 измерения второй коробки. Все числа не превышают 1 000 000 000.

Выходные данные

В первой и единственной строке одно целое положительное число – искомая максимальная высота.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
2 3 4 1 2 3	7

Задача 11. Пульт

Время на тест: 1 секунда

Ограничение по памяти 256Мб

В некотором городе было N светофоров, работающих по следующей стандартной схеме: КРАСНЫЙ (1) – ЖЕЛТЫЙ (2) – ЗЕЛЕНый (3) – ЖЕЛТЫЙ (4) – КРАСНЫЙ (1) и т.д. В скобках указан код состояния светофора. Для управления этими светофорами город заказал единый пульт. Техники подошли к процессу творчески и вместо N -кнопочного пульта сделали пульт с M кнопками. Каждая из этих кнопок была привязана к определенному набору светофоров и переводила каждый светофор из этого набора в следующее состояние. На резонный вопрос, зачем они так сделали, техники сказали, что так получилось, техническое задание выполнено и что за работу надо рассчитаться.

Открыв техническое задание, администрация города с удивлением обнаружила, что там был всего один пункт - «должно быть возможно, чтобы любое состояние всех светофоров можно было перевести в их любое состояние». Того, кто это написал, найти не удалось, потому деньги на оплату с него взыскать невозможно. Чтобы не платить за это чудо-устройство, остается один шанс — показать, что техническое задание не выполнено. Задача состоит в том, чтобы проверить, можно ли одно состояние системы светофоров перевести в другое с помощью сконструированного пульта.

Входные данные

В первой строке два натуральных числа N и M – количество светофоров и количество кнопок ($1 \leq N \leq 1000$, $1 \leq M \leq 1000$). Далее M строк, описывающих наборы светофоров для каждой кнопки. В каждой строке первое число K – количество светофоров в наборе, $1 \leq K \leq N$, далее K чисел через пробел – номера светофоров, подключенных к данной кнопке.

Далее две строки по N цифр без пробелов, описывающие стартовое и требуемое состояние всех светофоров. Состояния нумеруются от 1 до 4.

Выходные данные

В первой и единственной строке YES, если светофоры можно перевести из стартового состояния в требуемое, и NO в противном случае.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
3 3 2 1 3 2 2 3 2 1 2 0 0 0 1 2 1	YES

Задача 12. Типографская бухгалтерия

Время на тест: 1 секунда

Ограничение по памяти 256Мб

В небольшом городке приближается празднование дня города. Мэрия решила повесить на главной улице поздравительную растяжку. Каждый желающий житель города прислал свой проект текста, который он хотел бы видеть на этой растяжке: одну строку. Споры о том, какую же именно строку использовать, затянулись.

В последний день перед днём города мэр ударил кулаком по столу и сказал: «Из-за споров у нас не осталось времени на изготовление растяжки. Поэтому теперь мы будем выбирать ту строку, которую быстрее всего может напечатать типография, а если таких строк окажется несколько, то выберем ту, что дешевле всего».

В типографии мэру объяснили, что быстрее напечатать не самую короткую строку, а строку, в которой меньше различных символов. Кроме того, выяснилось, что не для всех символов стоимость печати одинакова: на некоторые символы уходит больше краски, а на некоторые меньше. Типография представила прайс-лист, в котором содержится таблица стоимостей печати некоторых особых символов и стоимость печати всех остальных, не попавших в таблицу. Мэр дал задание бухгалтерии оплатить работу. Главный бухгалтер вызвал главного (и единственного) программиста, т.е. Вас, и попросил разобраться с проблемой выбора строки и оплаты.

Входные данные

В первой строке задано N – число строк, из которых требуется выбрать текст растяжки ($1 \leq N \leq 1000$). Далее заданы эти N строк (каждая строка состоит из не более чем 100 символов: латинских букв и цифр). В следующей строке заданы два целых числа m ($0 \leq m \leq 62$) и p , где m – число строк в таблице стоимостей, а p стоимость печати символа, не попавшего в таблицу. В следующих m строках задана таблица стоимости печати символов. Каждая строка состоит из символа и, через 1 пробел, стоимости его печати. Стоимость печати любого символа – целое положительное число, не превышающее 1 000 000.

Выходные данные

В первой (и единственной) строке файла вывести одно число – стоимость печати растяжки.

Пример.

<i>Входные данные</i>	<i>Выходные данные</i>
3 aaab bbbs abcd 1 10 s 100	40

Задача 13. Найди машинку

Время на тест: 1 секунда

Ограничение по памяти 256Мб

Вашему младшему брату подарили на день рождения (1 год) радиоуправляемую машинку. На следующий день, когда все ушли на работу и Вы остались с братом дома вдвоём, Вы прочли инструкцию, вставили батарейки и показали брату, как пользоваться управлением. Тут к Вам пришли друзья, и Вы решили оставить ребёнка ненадолго одного в комнате. Чтобы с ним ничего не случилось, Вы посадили ребёнка на детский стульчик, из которого он сам не мог вылезти, дали ему в руки управление, поставили машинку в центр комнаты, направив её в сторону окна, и ушли. Пару минут Вы слышали, как машинка ездил и брат смеялся, а потом перестали обращать на шум внимание.

Поговорив с друзьями, Вы вернулись в комнату и обнаружили, что брат заснул, управление от машинки лежит рядом на полу, а машинки нет. Когда брат проснулся, он начал плакать и просить поиграть машинку. Вы искали машинку, но не смогли её найти. Тогда Вам пришла в голову идея почитать инструкцию внимательно – а нет ли у машинки команды подать какой-то звуковой сигнал? Такой команды не оказалось, но выяснилось, что у управления есть память, в которую записываются все отправленные машинке команды. Ура! По командам можно восстановить весь путь машинки и узнать, где она сейчас! Однако всё не так просто. Команд много, кодировка их не очень удобная, поэтому нужно написать специальную программу для определения местоположения машинки. Поторопитесь, к приходу родителей машинка должна была найдена.

Выдержки из инструкции: Машинка рассматривает всё пространство как клеточное поле. Машинка может исполнять следующий набор команд (команды кодируются цифрами от 1 до 4):

- Проехать вперёд на одну клетку (1);
- Развернуться на месте и проехать одну клетку (2);
- Повернуть налево и проехать одну клетку (3);
- Повернуть направо и проехать одну клетку (4).

Входные данные

В начальный момент машинка ориентирована в направлении возрастания второй координаты. В первой строке заданы целые числа p и q – начальные координаты машинки на клеточном поле. Во второй строке одно целое положительное число n – число команд в последовательности. В следующих n строках заданы команды (по одной цифре в строке). Числа n , p и q по абсолютной величине не превосходят 1 000.

Выходные данные

В первой (и единственной) строке файла вывести два целых числа, разделенные пробелом – координаты машинки после выполнения последовательности команд.

Пример

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