

А. А это обязательно?

1 секунда, 256 мегабайт

После неудачной стрелки с небезызвестным преподавателем Вы были отчислены и призваны в армию.

Вам, как новобранцу, полковник приказал покрасить траву на поле используя темно-зеленый и светло-зеленый цвета. Поле разбито на квадраты и имеет форму прямоугольника с n строками и m столбцами, каждый квадрат должен быть покрашен в один цвет. Для внесения разнообразия вам запрещено повторять раскраску на двух строках подряд.

Вы не уверены что это действительно полезно, но приказ есть приказ.

Входные данные

В единственной строке даны два целых числа n и m ($1 \leq n, m \leq 100$) — размеры поля в квадратах.

Выходные данные

Выведите любую раскраску удовлетворяющую условию. Светло-зеленый цвет обозначается символом «.» (ASCII код — 46), темно-зеленый обозначается символом «#» (ASCII код — 35)

входные данные
6 30
выходные данные
#####

В. Симулятор чёрных дыр

1 секунда, 256 мегабайт

Максим посмотрел много видео про космос и чёрные дыры. Он не смог позволить себе любительский телескоп, да и разглядеть в него чёрную дыру не получится, поэтому он решил запустить симулятор космических объектов на компьютере.

Максим называет чёрной дырой небесное тело настолько большой массы и плотности, что всё, что пересекает «горизонт событий», не может покинуть его, даже объект, движущийся со скоростью света. Чёрная дыра характеризуется своим радиусом Шварцшильда или просто размером. Сверхмассивная чёрная дыра — это чёрная дыра с массой от 10^5 до 10^{11} масс Солнца. Сверхмассивные чёрные дыры обнаружены в центре многих галактик, включая Млечный Путь.

Недавний снимок сверхмассивной чёрной дыры в центре галактики M 87:



Максим решил поиграться с чёрными дырами, сталкивая их. Как известно, при столкновении одна чёрная дыра поглощает другую, и образуется новая чёрная дыра, размер которой равен сумме размеров столкнувшихся черных дыр. Чёрная дыра может поглотить другую чёрную дыру, если её размер **строго больше** размера поглощаемой. Изначально в центре галактики находится сверхмассивная чёрная дыра массой Ω масс Солнца. По галактике разбросаны другие чёрные дыры, массы которых ω_i масс Солнца. Максим может перемещать центральную чёрную дыру в любую точку пространства и сталкивать с любой другой чёрной дырой строго меньшего размера. Максим хочет узнать, может ли сверхмассивная чёрная дыра в центре галактики поглотить все другие чёрные дыры.

Входные данные

В первой строке записаны два числа n и Ω ($1 \leq n \leq 2 \cdot 10^5, 1 \leq \Omega \leq 10^9$) — количество чёрных дыр в галактике и изначальная масса сверхмассивной чёрной дыры в центре галактики.

В следующей строке находится n чисел ($1 \leq \omega_i \leq 10^9$) — массы чёрных дыр, разбросанных по галактике.

Выходные данные

Если сверхмассивная чёрная дыра не может поглотить все чёрные дыры галактики, выведите «NO». Иначе выведите «YES» и порядок, в котором Максиму следует сталкивать сверхмассивную чёрную дыру с другими чёрными дырами.

входные данные
1 3 3
выходные данные
NO

входные данные
1 5 4
выходные данные
YES 1

входные данные
3 2 4 1 2

выходные данные
YES 2 3 1

С. Поиск подперестановки

1 секунда, 256 мегабайт

Вам даны массивы a и b . Их длины — n и m соответственно. Назовём подотрезок массива $a[l \dots r]$ *подперестановкой*, если существует способ переупорядочить его элементы таким образом, что он совпадёт с массивом b .

Например, $a = [1, 2, 3, 4, 2]$, $b = [2, 4, 3]$. Подотрезки $a[2 \dots 4] = [2, 3, 4]$ и $a[3 \dots 5] = [3, 4, 2]$ являются подперестановками, так как можно переупорядочить числа в них так, чтобы в точности получить массив b .

Ваша задача — найти все позиции, начиная с которых подотрезок массива a будет являться подперестановкой массива b .

Входные данные

В первой строке заданы два целых числа n и m ($1 \leq n, m \leq 2 \cdot 10^5$) — количество элементов в первом и втором массиве соответственно.

Во второй строке заданы n целых чисел ($1 \leq a_i \leq 100$) — элементы массива a .

Во третьей строке заданы m целых чисел ($1 \leq b_i \leq 100$) — элементы массива b .

Выходные данные

В первой строке выведите целое число k — количество позиций, начиная с которых подотрезок массива a будет подперестановкой массива b . В следующей строке выведите k целых чисел — искомые позиции в порядке возрастания.

входные данные
5 3 1 2 3 4 2 2 4 3
выходные данные
2 2 3

входные данные
7 1 1 2 3 1 2 3 1 1
выходные данные
3 1 4 7

входные данные
8 3 2 3 4 1 3 5 2 3 1 2 3
выходные данные
0

входные данные
6 2 4 3 4 3 3 4 4 3
выходные данные
4 1 2 3 5

D. Обитатель морских глубин

1 секунда, 256 мегабайт

Исследование морских глубин всегда связано с опасностями, например, атакой Кракена. В один день n кораблей отправились исследовать глубины океана. Капитаны кораблей знали об атаке Кракена, поэтому i -й корабль был способен выдержать a_i атак Кракена, прежде чем он будет потоплен. На борту одного из кораблей так же был опытный кракенолог, который прекрасно знал поведение Кракена.

Кракен атакует корабли на протяжении k дней и он делает это в определённом порядке. Для простоты можно считать, что он атакует корабли слева направо, а после атаки крайнего правого корабля он снова атакует крайний левый корабль. Если корабль был потоплен, то Кракен его пропускает, оставляя моряков в свободном плавании. Кроме того, кракенолог знает наперёд, что Кракен будет атаковать b_i кораблей в i -й день. Причём если Кракен закончил на j -м корабле, то на следующий день он продолжит атаковать начиная с $j + 1$ -го корабля.

Вам даны характеристики кораблей и хронология атак Кракена, восстановите сражение и определите, сколько кораблей потопил Кракен!

Входные данные

В первой строке записано целое число t ($1 \leq t \leq 100$) — количество наборов входных данных. Далее следуют наборы входных данных.

В первой строке каждого набора заданы два целых числа n и k ($1 \leq n \leq 10^3, 1 \leq k \leq 10^3$) — количество кораблей в море и количество дней, когда Кракен будет атаковать корабли.

Во второй строке каждого набора находится n целых чисел ($1 \leq a_i \leq 10^4$) — показатель прочности i -го корабля.

В третьей строке каждого набора находится k целых чисел ($1 \leq b_i \leq 10^4$) — количество кораблей, атакованных Кракеном в i -й день.

Гарантируется, что суммы n и k не превосходят 10^3 по всем наборам входных данных.

Выходные данные

Для каждого набора входных данных выведите единственное число — количество потопленных Кракеном кораблей.

входные данные
2 4 3 4 1 2 3 5 2 3 4 2 4 1 2 3 5 4
выходные данные
4 3

В первом тестовом наборе изначально корабли имеют следующие показатели прочности: $[4, 1, 2, 3]$

Кракен ударил 5 кораблей. Так как кораблей на плаву 4, то первый корабль будет атакован дважды: $[2, 0, 1, 2]$

Затем Кракен атакует два корабля: $[2, 0, 0, 1]$. Последним ударом Кракен топит все оставшиеся корабли $[0, 0, 0, 0]$.

Во втором тестовом наборе остался не потоплен только первый корабль, которой был атакован Кракеном трижды.

E. Никита и маркеры

1 секунда, 256 мегабайт

Никита работает в институте M^* преподавателем по алгоритмам и структурам данных. Для объяснения он использует доску и маркеры, однако приходится так много писать, что маркера хватает **ровно на одну лекцию**. Сейчас только начало учебного года, поэтому Никита хочет оценить, насколько ему хватит маркеров для проведения занятий.

В лекционной аудитории есть два ящика с маркерами. В каждом ящике находится N одинаковых маркеров. Каждую лекцию Никита равновероятно выбирает ящик и достаёт из него маркер для объяснения очередного алгоритма или структуры данных.

Никита интересуется, какова вероятность того, что когда он в первый раз откроет пустой ящик, в другом будет ровно k маркеров.

Входные данные
В первой строке заданы два целых числа n и k ($1 \leq k \leq n \leq 2000$) — количество маркеров в ящиках и интересующее Никиту число оставшихся маркеров.

Выходные данные
Выведите единственное число — ответ на задачу. Ответ требуется вывести в следующем виде: пусть искомая вероятность представима в виде несократимой дроби $P = \frac{a}{b}$, тогда ответом будет число $a \cdot b^{-1} \bmod (10^9 + 7)$.

входные данные
1 1
выходные данные
500000004

входные данные
2 1
выходные данные
375000003

входные данные
2 2
выходные данные
250000002

В первом тесте искомая вероятность $P = 2 \cdot \frac{1}{4} = \frac{1}{2}$, тогда ответ равен $1 \cdot 2^{-1} \bmod (10^9 + 7) \equiv 500000004 \bmod (10^9 + 7)$.

Во втором тесте вероятность $P = \frac{3}{8}$.

В третьем тесте вероятность $P = \frac{1}{4}$.

Г. Егор и куб

1 секунда, 256 мегабайт

Егор поступил на первый курс, на котором преподают математический анализ, дискретную математику и, конечно же, линейную алгебру и аналитическую геометрию. Будучи прилежным студентом Егор всегда выполняет всю домашнюю работу, но сегодня задание оказалось сложнее предыдущих. Егор просит Вас помочь ему.

Дан куб со стороной a и три точки A, B и C , лежащие на рёбрах куба, требуется построить сечение куба, содержащее эти три точки и сказать, какая фигура получится в сечении. Егор интересуется, сколько вершин содержит получившийся многоугольник. Гарантируется, что плоскость сечения не совпадает с гранью куба.

Вершины куба имеют координаты $(0, 0, 0), (0, a, 0), (a, a, 0), (a, 0, 0), (0, 0, a), (0, a, a), (a, a, a), (a, 0, a)$.

Входные данные

В первой строке находится целое число a ($1 \leq a \leq 1000$) — сторона куба. В следующих трёх строках находится по три целых числа — координаты точек A, B и C . Гарантируется, что точки не лежат на одной прямой.

Выходные данные
Выведите единственное число — количество вершин многоугольника в сечении куба плоскостью

входные данные
10 0 10 10 10 0 10 10 10 0
выходные данные
3

входные данные
10 0 0 0 10 10 10 0 10 5
выходные данные
4

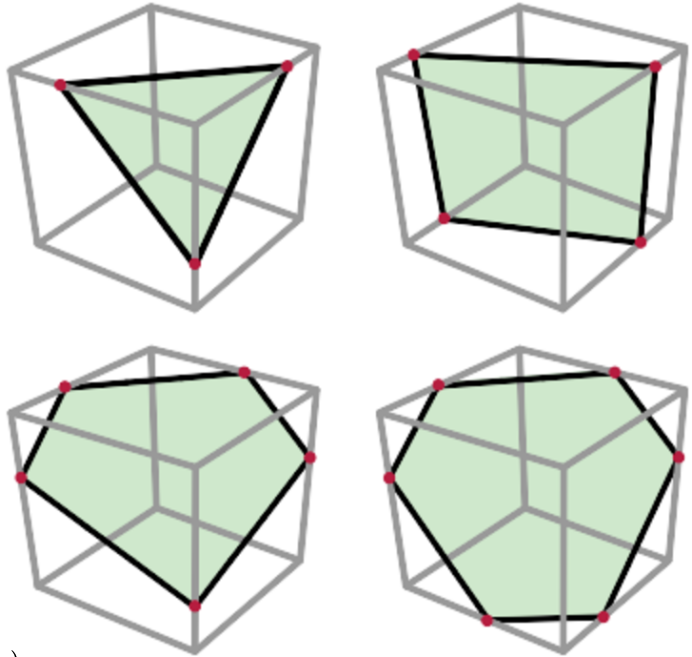
входные данные
10 0 0 0 10 8 10 0 10 4
выходные данные
5

входные данные
10 0 0 0 10 8 10 0 10 4
выходные данные
5

входные данные
10 0 5 0 5 0 0 10 5 10
выходные данные
6

входные данные
10 0 5 0 5 0 0 10 5 10
выходные данные
6

Ниже приведены иллюстрации возможных сечений куба:



Г. Голодные игры

1 секунда, 256 мегабайт

Это интерактивная задача. Не забывайте о том, что ваша программа должна каждый раз после вывода запроса сбрасывать буфер вывода.

Голодные студенты Лёлик и Болик играют с цепочкой из N сосисок. Правила просты. За ход студент может выбрать одну из оставшихся цепочек и вырезать из нее в любом месте от 1 до K подряд идущих сосисок.

Проигрывает игрок который не может сделать ход. Лёлик ходит первым.

Болик мастер этой игры и всегда играет оптимально. Цена ошибки велика, ведь, как известно, победитель забирает все сосиски себе. Как лучший друг Лёлика вы согласились ему помочь.

Победите Болика в этой непростой и важной игре.

Входные данные

Первая строка входных данных содержит два целых числа N и K ($3 \leq N \leq 2000, 2 \leq K < N$) — изначальный размер цепочки и максимальное количество сосисок которое можно взять за один ход.

Протокол взаимодействия

Чтобы сделать ход, выведите два целых числа в отдельной строке pos, len ($1 \leq pos \leq N, 1 \leq len \leq K$) — номер первой сосиски и количество сосисок которые вы возьмете. После совершения операции, нумерация сосисок не меняется.

В случае если вы выигрываете после этого хода и противник сделать ход не может выведите в конце строку WIN .

После вашего хода, в том же формате описывается ход противника: pos, len ($1 \leq pos \leq N, 1 \leq len \leq K$), если $pos = -1$ и $len = -1$, это означает, что вы сделали некорректный ход. Ваша программа должна немедленно завершиться (например, вызовом `exit(0)`). Вы получите вердикт «Неправильный ответ», и это будет означать, что вы сделали некорректный ход. Если вы проигнорируете это, то можете получить любой вердикт, так как ваша программа продолжит читать из закрытого потока ввода.

Ваше решение получит вердикт «Решение зависло», если вы не будете ничего выводить или забудете сделать операцию `flush` после вывода вопроса или ответа.

входные данные
5 2 1 1
выходные данные
2 2 4 2 WIN

Пояснение к первому примеру.

В начале игры имеем одну цепочку [1, 5]. На первом ходу забираем [2, 3], остается две цепочки [1, 1] и [4, 5] На втором ходу противник забирает [1, 1], остается одна цепочка [4, 5]. На третьем ходу забираем [4, 5]. Больше сосисок не осталось, противник не может сделать ход, значит мы выиграли. Выводим WIN

Н. Китайский вирус против настольной игры

1 секунда, 256 мегабайт

На днях вышла новая настольная игра, состоящая из n уровней. Каждый уровень описывается интересностью уровня a_i . Компания друзей хочет вместе пройти игру. Садиться играть можно только один раз, то есть один игрок может пройти последовательно уровни с l по r ровно один раз. Каждый уровень должен быть пройден, нельзя несколько раз проходить один и тот же уровень.

Если суммарная интересность пройденных игроком уровней больше k , то остальные друзья его осуждают, ведь они тоже хотели получить удовольствие от игры. Разумеется, дружба важнее игры и это условие нарушаться не будет.

Из-за ограничений на массовые мероприятия, вы хотите, чтобы играло минимальное количество Ваших друзей. Определите минимальное количество игроков, которое понадобится чтобы полностью пройти игру.

Входные данные

В первой строке записано целое число t ($1 \leq t \leq 10^3$) — количество наборов входных данных. Далее следуют наборы входных данных.

В первой строке каждого набора заданы два целых числа n и k ($1 \leq n \leq 10^4, 1 \leq k \leq 10^5$) — количество уровней в игре и максимальная интересность уровней, которые может пройти один игрок.

В следующей строке находится n чисел ($-10^3 \leq a_i \leq 10^3$) — интересность i -го уровня.

Гарантируется, что сумма n не превосходит 10^4 по всем наборам входных данных.

Выходные данные

Для каждого набора входных данных выведите единственное число — минимально возможное число игроков, которые могут пройти игру.

Если пройти игру с данными ограничениями невозможно, выведите «-1» без кавычек.

входные данные
5 5 6 1 2 3 4 5 5 5 1 2 3 4 5 5 5 1 -3 12 -4 5 5 5 1 -3 13 -4 5 5 5 1 5 -3 2 1
выходные данные
3 4 3 -1 2

I. НОД на клетчатом поле

2 секунды, 256 мегабайт

Совсем недавно Василий узнал про алгоритм Евклида нахождения наибольшего общего делителя двух чисел. Наибольшим общим делителем двух чисел a и b называется наибольшее число, на которое a и b делятся без остатка. С помощью этих знаний Василий может решить задачу, которую он когда-то не решил.

У Василия есть клетчатое поле n строк на m столбцов, на пересечении i строки и j столбца находится число a_{ij} . Василий хочет попасть из левого верхнего угла в правый нижний угол и узнать, какой НОД у всех чисел на пути. Разрешается двигаться только вниз и вправо. Василий выписал несколько путей и получил разные значения НОД, ему стало интересно, какой максимальный НОД можно получить.

К сожалению, Василий устал считать НОД чисел и поэтому просит Вас помочь ему найти максимальный НОД чисел на пути из левого верхнего угла в правый нижний угол клетчатого поля.

Входные данные

В первой строке записано целое число t ($1 \leq t \leq 100$) — количество наборов входных данных. Далее следуют наборы входных данных.

В первой строке каждого набора заданы два целых числа n и m ($1 \leq n, m \leq 100$) — количество строк и столбцов клетчатого поля соответственно.

В следующих n строках находятся по m чисел ($1 \leq a_{ij} \leq 10^6$) — числа, записанные в i строке и j столбце клетчатого поля.

Гарантируется, что суммы n и m не превосходят 1000 по всем наборам входных данных.

Выходные данные

Для каждого набора входных данных выведите единственное число — максимально возможный gcd на пути из клетки левой верхней клетки в правую нижнюю клетку.

входные данные
3 1 3 2 4 6 3 1 3 6 9 2 2 10 30 15 20
выходные данные
2 3 10

Ж. Самый модный на районе

3 секунды, 256 мегабайт

Сергей недавно закупился в магазине одежды и теперь ему предстоит разобраться в этом обилии покупок и выбрать среди них самые красивые. Каждая вещь в гардеробе Сергея характеризуется показателем красоты. Сергей хочет надеть кепку, футболку, джинсы и кроссовки. Красота надетых вещей равна суммарной красоте кепки, футболки, джинс и кроссовок, которые надел Сергей. Однако, некоторые кепки не подходят каким-то футболкам, некоторые футболки каким-то джинсам и некоторые джинсы каким-то парам кроссовок.

У Сергея есть n_1 кепок, n_2 футболок, n_3 пар джинс и n_4 пар кроссовок. Так же Сергей знает, что есть m_1, m_2 и m_3 пар несовместимых между собой пар кепок и футболок, футболок и джинс, джинс и кроссовок соответственно.

Сергей хочет одеться максимально красиво, учитывая то, что некоторая одежда не подходит друг другу. Сам он не смог справиться с выбором из такого количества одежды в гардеробе, поэтому попросил Вас помочь ему.

Входные данные

В первой строке записано целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных. Далее следуют наборы входных данных.

В первой строке каждого набора заданы четыре целых числа n_1, n_2, n_3 и n_4 ($1 \leq n_1, n_2, n_3, n_4 \leq 10^5$) — количество кепок, футболок, пар джинс и кроссовок соответственно.

В следующих четырёх строках находятся по n_1, n_2, n_3 и n_4 чисел ($1 \leq a_{1i}, a_{2i}, a_{3i}, a_{4i} \leq 10^8$) — описания красоты кепок, футболок, пар джинс и кроссовок соответственно.

В следующей строке задано число m_1 ($0 \leq m_1 \leq 10^5$) — количество пар несовместимых кепок и футболок. В каждой из следующих m_1 строк заданы пары чисел u и v — номера несовместимых кепок и футболок.

Описания для несовместимых пар футболок и джинс, джинс и кроссовок аналогичные.

Гарантируется, что суммы $n_1, n_2, n_3, n_4, m_1, m_2$ и m_3 не превосходят 10^5 по всем наборам входных данных.

Выходные данные

Для каждого набора входных данных выведите единственное число — максимально возможную итоговую красоту одежды, которую Сергей может надеть. Если вся одежда несовместима, то выведите «-1» без кавычек.

входные данные
3 1 2 3 4 6 3 4 8 1 2 5 10 7 9 2 1 1 1 2 3 2 2 2 3 1 3 4 1 1 3 1 2 3 2 4 1 2 3 4 6 3 4 8 1 2 5 10 7 9 1 1 2 4 2 2 2 3 1 1 1 2 7 2 2 2 4 3 4 3 2 1 4 1 2 1 3 1 1 1 1 100000000 100000000 100000000 100000000 0 0 0
выходные данные
-1 18 400000000

К. На какой стул сядешь?

1 секунда, 256 мегабайт

Судьба дала второй шанс великому комбинатору Остапу Блендеру. Он обнаружил N стульев стоящих в ряд, на каждом стуле было записано положительное число, а рядом лежал обрывок записки «... Я спрятала две шкатулки драгоценностей в стульях с номерами ...».

«Все не может быть так просто», — подумал товарищ Блендер. Он решил что сокровища непременно лежат в двух различных стульях, таких что НОД чисел, написанных на стульях между ними (включая их самих), не равен 1.

Остап принялся считать количество пар удовлетворяющих условию, но число было слишком велико и поэтому он обратился к Вам за помощью.

Входные данные

В первой строке задано единственное целое число N ($1 \leq N \leq 2 \cdot 10^5$) — количество стульев.

Во второй строке находятся N значений a_i ($1 \leq a_i \leq 10^7$) — числа написанные на стульях.

Выходные данные

В единственной строке выведите одно число — ответ на задачу.

входные данные
8 22 4 6 15 20 1 30 7
выходные данные
5

Подходящие пары стульев в примере:
(1, 2), (2, 3), (3, 4), (4, 5), (1, 3)

Z. Усердная работа

2 секунды, 256 мегабайт

Как говорится: «Поспешишь - людей насмешишь»

Подумайте подольше и не спешите говорить, иначе это может привести вас к неверному ответу...

Входные данные

В первой строке дано целое число n ($30 \leq n \leq 60$).

Выходные данные

Выведите единственное число 2^n . На всякий случай не забудьте сбросить вывод при помощи `flush`.

входные данные
42
выходные данные
4398046511104