

Задача А. Я буду долго гнать велосипед...

Имя входного файла: `bicycle.in`
Имя выходного файла: `bicycle.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Наконец-то, и город Нью-Васюки приблизился к цивилизации - его власти построили целых две велодорожки, по каждой из которых можно кататься в двух направлениях! Каждая из них имеет форму прямоугольника, стороны которого параллельны координатам, нанесённым на официальную карту города. Велосипедистам разрешается сворачивать с одной дорожки на другую, но категорически запрещено выезжать за их пределы (например, внутрь прямоугольника, который ограничивается дорожкой). Дорожки могут пересекать или касаться друг друга, и даже иметь общие участки ненулевой длины.

Вам необходимо проехать как можно быстрее из пункта **А** в пункт **В**. Известно, что оба этих пункта находятся на одной из велодорожек, а единицу расстояния Вы проезжаете за единицу времени.

Формат входных данных

Первая и вторая строка файла содержит координаты северо-западного и юго-восточного углов прямоугольника, соответствующего одной из велодорожек. В третьей и четвёртой строках записаны соответственно координаты пунктов **А** и **В**. Все координаты — целые числа, не превосходящие по модулю 10^9 . Прямоугольники, которые описывают каждую из дорожек, имеют ненулевую высоту и ширину.

Формат выходных данных

Вычислите минимальное время, за которое Вы сможете доехать из пункта **А** в пункт **В**, соблюдая приведенные выше условия. Если такая поездка невозможна, выведите -1.

Примеры

bicycle.in	bicycle.out
0 10 10 0 10 20 20 10 0 0 20 15	35
0 10 10 0 2 8 8 2 0 0 2 2	-1

Задача В. Двадцать одно (Division 1 Only!)

Имя входного файла: `blackjack.in`
Имя выходного файла: `blackjack.out`
Ограничение по времени: 2 seconds
Ограничение по памяти: 64 megabytes

Петя любит играть в «двадцать одно» на деньги. Но выигрывает не всегда... Родители знают, что с вероятностью p он проигрывает отдельную партию, а ничьих у него не бывает. При этом за каждый выигрыш Петя получает 1 тугрик, за каждый проигрыш — теряет.

И вот однажды он взял A тугриков и пошёл в гости к своему другу Васе. Вася достал свои B тугриков, и они решили играть до полной победы, то есть пока у одного из них не закончатся деньги.

Родители Пети всё это знают, потому что он от них ничего не скрывает. Они не переживают, что Петя может проиграть все свои деньги. Ведь тогда они перейдут в хорошие руки. Если Петя обыграет Васю, тоже ничего страшного не произойдёт, ведь они друзья и Петя при необходимости поделится. Родителей интересует только одно — в какое время приготовить Пете вкусный ужин. Но вкусный ужин готовить долго, а остывает он быстро. Поэтому им надо определить, когда ждать Петю домой, то есть ожидаемое количество партий.

Формат входных данных

В первой строке одно целое число p ($0 \leq p \leq 100$) — вероятность проигрыша в процентах. Во второй строке содержатся два целых числа A и B ($1 \leq A, B \leq 10^9$), разделенные пробелом.

Формат выходных данных

Вывести одно число — математическое ожидание количества партий с абсолютной или относительной погрешностью не более 10^{-9} .

Примеры

<code>blackjack.in</code>	<code>blackjack.out</code>
100	10.0000000000000000
10 20	
25	2.153846153846154
1 2	

Задача С. По кругу (Division 1 Only!)

Имя входного файла: `circular.in`
Имя выходного файла: `curcular.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дан граф из N вершин и M ребер. Коля хочет вершины расположить на окружности радиуса R (центр окружности в точке $(0, 0)$) так, чтобы ребра соответствовали отрезкам соединяющим вершины, и при этом могли пересекаться только в вершинах. Евклидово расстояние между любыми вершинами должно быть не меньше 1.

Возможно ли такое расположение?

Формат входных данных

В первой строке записаны три числа N , M и R ($1 \leq N, R \leq 1000$, $0 \leq M \leq 5000$). Далее в M строках описываются ребра графа (по одному в строке): два различных целых числа a_i и b_i ($1 \leq a_i, b_i \leq N$). В графе нет мультиребер.

Формат выходных данных

В первой строке выведите единственное слово **Yes**, если такое расположение возможно, иначе **No**. В случае положительного ответа далее в N строках выведите координаты вершин графа x_i , y_i (относительная погрешность при проверке принадлежности точки окружности не должна превышать 10^{-8}).

Примеры

<code>circular.in</code>	<code>curcular.out</code>
3 3 1 1 2 1 3 3 2	Yes 1.0000000000 0.0000000000 -0.5000000000 -0.8660254038 -0.5000000000 0.8660254038
4 6 100 1 2 1 3 1 4 2 3 2 4 3 4	No

Задача D. Угадай тяжелую мелодию (Division 1 Only!)

Имя входного файла: `entropy.in`
Имя выходного файла: `entropy.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Популярное телешоу «Угадай тяжелую мелодию» готовится к очередному эфиру. В это трудно поверить, но мелодия, которую нужно угадать, в самом деле всегда выбирается случайно. Но, безусловно, не все мелодии должны быть равновероятны. Хит ушедшего лета, песня «Heavy Heavy Metal» группы «Ocean of Metal», должна выпадать чаще, чем слегка позабытый тяжелый рок 70-ых.

Вставшая перед организаторами задача заключается в следующем: имеется N мелодий, для мелодии каждой мелодии m_i известна минимальная и максимальная вероятность ее выпадения, обозначаемые l_i и r_i соответственно. Необходимо выбрать распределение вероятностей $p(m_i) = p_i$ ($l_i \leq p_i \leq r_i$), при котором каждая песня будет выпадать с вероятностью p_i . Чтобы шоу было как можно более захватывающим и непредсказуемым, организаторы хотят выбрать распределение с максимальной возможной при данных условиях энтропией.

Помогите им решить эту задачу.

Формат входных данных

Первая строка ввода содержит единственное натуральное число N ($1 \leq N \leq 10^5$). Вторая строка ввода содержит N вещественных чисел l_i , каждое в формате с фиксированной точкой, не более чем с 9-ю знаками после запятой. Третья строка ввода содержит числа r_i , в том же формате, что и вторая строка. Дополнительно известно, что $\sum_{i=1}^N l_i \leq 1$, $\sum_{i=1}^N r_i \geq 1$ и $l_i \leq r_i$ для всех $1 \leq i \leq N$.

Формат выходных данных

Выведите N вещественных чисел p_i , разделенных пробелами, где p_i — вероятность выпадения i -ой песни, $l_i \leq p_i \leq r_i$, и при соблюдении этих условий условия энтропия p максимальна. Ваш ответ будет засчитан, если $|\sum_{i=1}^N p_i - 1|$ будет меньше 10^{-9} и для энтропии H предложенного вами распределения будет верно следующее неравенство: $1 - \frac{H}{H_{max}} \leq 10^{-9}$, где H_{max} — максимальная возможная энтропия при данных условиях.

Примеры

entropy.in	entropy.out
3	0.3333333333333333
0.0 0.0 0.0	0.3333333333333333
1.0 1.0 1.0	0.3333333333333333
4	0.5000000000000000
0.5 0.1 0.1 0.1	0.1000000000000000
0.5 0.1 0.2 0.2	0.2000000000000000
	0.2000000000000000

Примечание

Распределением вероятности на конечном множестве событий $X = \{x_1, \dots, x_N\}$ называется

функция $p(x_i) = p_i$, где $0 \leq p_i \leq 1$ и $\sum_{i=1}^N p_i = 1$. Энтропия распределения p вычисляется по формуле $\sum_{i=1}^N p_i \log_2 1/p_i$.

Задача Е. Факториал (Division 1 Only!)

Имя входного файла: `factorial.in`
Имя выходного файла: `factorial.out`
Ограничение по времени: 1 секунда (2 секунды для Java)
Ограничение по памяти: 256 мегабайт

Посчитайте $N! = 1 \cdot 2 \cdot \dots \cdot N$.

Формат входных данных

Единственная строка содержит целое число N ($1 \leq N \leq 40\,000$).

Формат выходных данных

Единственная строка должна содержать $N!$ без незначащих нулей.

Примеры

<code>factorial.in</code>	<code>factorial.out</code>
1	1
2	2

Задача F. И будет вам щасце!

Имя входного файла: `gluck.in`
Имя выходного файла: `gluck.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Земная колония на далёкой планете Новый Китай процветает (в отличие от перенаселённой метрополии), и ей нужны новые жители. . . Руководство колонии прислало заявку на N супружеских пар, которые согласились бы навсегда поселиться в Новом Китае. Увы, столько желающих не нашлось. . . Зато власти Земли смогли отобрать N незамужних женщин и столько же холостых мужчин, согласившихся променять земные трущобы на «рай» в колонии. К концу межпланетного путешествия все они должны составить супружеские пары, причём подбор супругов выполняют прикомандированные психологи, а у переселенцев нет права изменить этот выбор. Во время долгого перелёта психологи, наблюдая за контактами между переселенцами, смогли определить т.н. взаимный уровень симпатии s_{ij} между i -м мужчиной и j -й женщиной из них. Психологи считают, что чем больше величина взаимной симпатии, тем более прочным и счастливым получится брак. Помогите психологам, рассчитав такой вариант формирования из переселенцев супружеских пар, при котором суммарная величина взаимной симпатии в парах будет максимальной.

Формат входных данных

Первая строка файла содержит величину N ($1 \leq N \leq 100$). Далее следуют N строк из N чисел каждая — матрица взаимных симпатий s_{ij} . Величины s_{ij} — целые, неотрицательные, не превосходящие 100.

Формат выходных данных

Первая строка этого файла должна содержать рассчитанную суммарную величину взаимной симпатии. Каждая из последующих N строк соответствует одной супружеской паре и содержит номера мужчины и женщины, которые эту пару составят (нумерация начинается с единицы). Если задача допускает несколько решений, выведите любое из них.

Примеры

<code>gluck.in</code>	<code>gluck.out</code>
3 3 2 4 2 1 2 2 2 5	9 1 1 2 2 3 3
3 1 2 3 3 2 1 2 2 2	8 1 3 2 1 3 2

Задача G. Фонари у дороги 2

Имя входного файла: `lantern2.in`
Имя выходного файла: `lantern2.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Прямой отрезок дороги из Нью-Васюков в Старую Москву снискал дурную славу из-за большого количества аварий на нём, особенно в ночное время. Дорожный департамент в конце концов решил установить фонари и осветить весь этот отрезок. И тут чиновники обратили внимание, что территория вокруг дороги, возможно, включая отдельные её участки, уже освещается N фонарями, стоящими у заправочных станций, придорожных кафе и т.д., так что выделенные деньги можно сэкономить... В частности, фонарь с номером i ($1 \leq i \leq N$) освещает круг с центром в точке (a_i, b_i) с радиусом R_i . Дорога проходит из точки (x_1, y_1) в точку (x_2, y_2) , и её шириной можно пренебречь.

Участок дороги считается безопасным, если он освещен хотя бы одним фонарём.

По информации о расположении освещённых участков дороги определите суммарную длину безопасных участков дороги.

Формат входных данных

Первая строка содержит одно целое число: N — количество фонарей ($1 \leq N \leq 10^5$). Во второй строке задаются координаты x_1, y_1, x_2, y_2 — целые числа, не превосходящие по модулю 10^6 .

Каждая из последующих N строк задаёт информацию об одном фонаре и содержит три целых числа a_i, b_i и R_i . Диапазоны значений для этих чисел: $-10^6 \leq a_i, b_i \leq 10^6$, $1 \leq R_i \leq 10^6$.

Формат выходных данных

Выведите единственное число — суммарную длину безопасных участков дороги с относительной или абсолютной погрешностью не более 10^{-6} .

Примеры

lantern2.in	lantern2.out
1 0 0 9 0 3 2 3	4.4721359550
2 -1 -4 5 4 6 -3 5 3 3 3	4.8284271247

Задача Н. Палиндромы

Имя входного файла: palindroming.in
Имя выходного файла: palindroming.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

According to Guinness World Records, the Finnish word saippuakivikauppias (soapstone vendor), a 19-letter word, is claimed to be the world's longest palindromic word in everyday use.

© Wikipedia

Вам дана строка s из маленьких латинских букв. Вы можете за один ход вычеркнуть какую-то букву в ней или поменять две буквы местами.

Сколько разных палиндромов вы можете получить, сделав несколько ходов (возможно ноль)?

Формат входных данных

Во входном файле содержится одна непустая строка s (длиной до ста символов).

Формат выходных данных

Выведите единственное число — количество различных палиндромов, которые можно получить. Так как это число может оказаться слишком большим, то выведите только остаток от деления этого числа на $10^9 + 2012$.

Примеры

palindroming.in	palindroming.out
acb	3
abracadabra	157

Примечание

Палиндромом называется непустая строка, которая одинаково читается в обоих направлениях.

Задача I. Серии

Имя входного файла: `runs.in`
Имя выходного файла: `runs.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Для заданной перестановки π чисел от 1 до N *серией* назовём максимальный (по включению) интервал индексов $[i; j]$, для которого выполняется $\pi(k) < \pi(k+1)$ при всех $i \leq k < j$ или $\pi(k) > \pi(k+1)$ при всех $i \leq k < j$. Например, перестановка $\pi = (\begin{smallmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 5 & 2 & 3 & 4 \end{smallmatrix})$ содержит 3 серии: $[1; 2]$, $[2; 3]$, $[3; 5]$, поскольку $\pi(1) < \pi(2)$, $\pi(2) > \pi(3)$ и $\pi(3) < \pi(4) < \pi(5)$.

Пусть при заданном N выбирается случайная перестановка из N чисел (все возможные перестановки имеют равные вероятности). Требуется посчитать вероятность, что выбранная перестановка будет содержать ровно R серий.

Формат входных данных

Единственная строка содержит два целых числа N и R ($1 \leq N, R \leq 500$) — количество элементов в перестановке и требуемое количество серий.

Формат выходных данных

Единственная строка должна содержать искомую вероятность с абсолютной погрешностью не более 10^{-9} .

Примеры

<code>runs.in</code>	<code>runs.out</code>
3 1	0.333333333333
4 2	0.500000000000

Задача J. По спирали 2. . .

Имя входного файла: `spiral2.in`
Имя выходного файла: `spiral2.out`
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 64 мегабайта

Матрица, состоящая из N строк и N столбцов, заполняется натуральными числами по спирали: первый столбец сверху вниз, оставшиеся позиции последней строки — слева направо, оставшиеся элементы последнего столбца — снизу вверх, оставшиеся элементы первой строки — справа налево, и т. д.

Так, матрица из 4 строк и 4 столбцов имеет вид

1	12	11	10
2	13	16	9
3	14	15	8
4	5	6	7

Найдите сумму значений элементов прямоугольной подматрицы между строками r_1 и r_2 и столбцами c_1 и c_2 .

Формат входных данных

В первой строке записано одно число N ($1 \leq N \leq 10^9$). Во второй — четыре числа r_1, r_2, c_1, c_2 ($1 \leq r_1 \leq r_2 \leq N, 1 \leq c_1 \leq c_2 \leq N$).

Формат выходных данных

Выведите единственное число — сумму элементов в подматрице по модулю $10^9 + 2012$.

Примеры

<code>spiral2.in</code>	<code>spiral2.out</code>
4 2 3 1 3	63

Задача К. Фонари над дорогой (Division 2 Only!)

Имя входного файла: `lantern.in`
Имя выходного файла: `lantern.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Прямой отрезок дороги длины M снискал дурную славу из-за большого количества аварий на нём, особенно в ночное время. Дорожный департамент в конце концов решил установить фонари и осветить весь этот отрезок. И тут чиновники обратили внимание, что отдельные участки дороги уже освещаются N фонарями, стоящими у заправочных станций, придорожных кафе и т.д., так что выделенные деньги можно сэкономить... В частности, фонарь с номером i ($1 \leq i \leq N$) освещает участок дороги $[a_i, b_i]$, где $0 \leq a_i < b_i \leq M$.

Участок дороги считается безопасным, если он освещен хотя бы одним фонарём.

По информации о расположении освещённых участков дороги определите суммарную длину безопасных участков дороги и количество неосвещённых участков.

Формат входных данных

Первая строка содержит два целых числа: M — длину дороги и N — количество фонарей ($1 \leq M \leq 10^6, 1 \leq N \leq 10^5$). Каждая из последующих N строк задаёт информацию об одном фонаре и содержит два целых числа a_i и b_i .

Формат выходных данных

Выведите два числа, разделённые пробелом — суммарную длину безопасных участков дороги и количество неосвещённых участков.

Примеры

<code>lantern.in</code>	<code>lantern.out</code>
8 2 1 5 4 8	7 1
8 2 0 6 6 8	8 0

Задача L. Прогрессия (Division 2 Only!)

Имя входного файла: `progression.in`
Имя выходного файла: `progression.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Задана последовательность из N целых чисел. Выделите из нее подпоследовательность стоящих подряд чисел максимальной длины, которая образовывала бы арифметическую прогрессию.

Напомним, что последовательность a_1, a_2, \dots, a_k является арифметической прогрессией, если разность $a_i - a_{i-1}$ постоянна для $i = 2, \dots, k$.

Формат входных данных

Первая строка содержит величину N ($2 \leq N \leq 10^6$). Каждая из последующих N строк содержит одно число, не превосходящее по модулю 10^9 — очередной элемент исходной последовательности.

Формат выходных данных

Выведите единственное число — длину найденной прогрессии.

Примеры

<code>progression.in</code>	<code>progression.out</code>
7 1 2 3 4 5 4 3	5
5 3 1 8 -4 0	2

Задача М. По спирали... (Division 2 Only!)

Имя входного файла: `spiral.in`
Имя выходного файла: `spiral.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Матрица, состоящая из M строк и N столбцов, заполняется натуральными числами по спирали: первый столбец сверху вниз, оставшиеся позиции последней строки — слева направо, оставшиеся элементы последнего столбца — снизу вверх, оставшиеся элементы первой строки — справа налево, и т. д.

Так, матрица из 4 строк и 5 столбцов имеет вид

1	12	11	10
2	13	16	9
3	14	15	8
4	5	6	7

Найдите значение элемента матрицы, стоящего в i -й строке и j -м столбце.

Формат входных данных

Первая строка входного файла содержит величины M и N ($1 \leq M, N \leq 10^9$).

Вторая строка содержит значения i и j ($1 \leq i \leq M, 1 \leq j \leq N$).

Формат выходных данных

Единственная строка выходного файла должна содержать искомое значение элемента матрицы.

Примеры

<code>spiral.in</code>	<code>spiral.out</code>
4 5 2 2	15
20 30 1 2	96