

Задача А. Карта из сундука мертвеца

Имя входного файла: map.in
Имя выходного файла: map.out
Ограничение по времени: 3 seconds
Ограничение по памяти: 256 Mebibytes

Перед участниками экспедиции была полная карта Острова Сокровищ, а также обрывок карты, найденный в сундуке покойного Билли Бонса. Доктор Ливси, пристально изучая карту, заметил, что масштаб изображения совпадает. Правда, он ещё уточнил, что обрывок карты из сундука, равно как и большая карта, не являются абсолютно точными, так что полного совпадения добиться не удалось. Значит, надо искать похожие фрагменты.

Как карта всего острова, так и найденный фрагмент представляют собой матрицы размера $M \times M$ и $N \times N$ соответственно ($1 \leq M, N \leq 800, N \leq M$). Каждый элемент матрицы соответствует одному пикселю карты и содержит целое число от 32 до 127 — интенсивность белого цвета в этом пикселе. Будем считать, что изображения на найденном фрагменте и совпадающей по размеру части карты похожи, если соответствующие им матрицы различаются только в одном элементе, и значения интенсивности в этом элементе различаются только на единицу. Повороты как всей карты, так и найденного фрагмента недопустимы!

Сколько различных частей карты, похожих на найденный фрагмент, можно найти? Части карты считаются различными, если координаты их северо-западного угла различны.

Формат входного файла

Первая строка входного файла содержит значения величин M и N , Далее следуют M строк из M символов каждая — описание всей карты. Код i -го символа j -й строки равен интенсивности белого цвета в пикселе (i, j) . Следующие N строк из N символов каждая соответствуют описанию найденного фрагмента и имеют тот же формат.

Формат выходного файла

Единственная строка выходного файла должна содержать найденное количество похожих частей карты.

Пример

map.in	map.out
5 3 bbbab ababa bbbab ababa bbabb bab aba bbb	3

Задача В. Диагональ крышки

Имя входного файла: `diag.in`
Имя выходного файла: `diag.out`
Ограничение по времени: `2 seconds`
Ограничение по памяти: `256 Mebibytes`

При осмотре «Эспаньолы» доктор Ливси внимательно изучал размеры погрузочного люка. «Зачем это Вам, доктор?» — поинтересовался сквайр Трелони. Оказывается, в вещах Билли Бонса было найдено описание сундука с сокровищами. Учитывая, что описание было изрядно зашифровано, доктору Ливси было известно только, что площадь прямоугольной крышки сундука задается выражением $a \times P^2 + b \times P + c$, где a, b, c — заданные коэффициенты, P — периметр прямоугольника. Доктор пытался понять, можно ли погрузить данный сундук в люк. Для этого ему надо, в частности, определить максимально возможную длину диагонали его крышки.

Ваша задача — написать программу, находящую вышеуказанную длину.

Формат входного файла

Единственная строка этого файла содержит величины a, b, c — вещественные числа, не превосходящие по модулю 10^6 и записанные не более, чем с 3 знаками в дробной части. Числа во входном файле разделены одним или несколькими пробелами.

Формат выходного файла

Единственная строка выходного файла должна содержать искомую длину диагонали, вычисленную с точностью до 0.0001. Если длина диагонали может быть сколь угодно большой, следует вывести `Infinity`, если же такого прямоугольника не существует, следует вывести `Impossible`.

Пример

<code>diag.in</code>	<code>diag.out</code>
<code>1 1 1</code>	<code>Impossible</code>
<code>0 -3 7</code>	<code>1.6667</code>
<code>0 0 0</code>	<code>Infinity</code>

Задача С. Ферзи

Имя входного файла: `queens.in`
Имя выходного файла: `queens.out`
Ограничение по времени: 6 seconds
Ограничение по памяти: 256 Mebibytes

В пути доктор Ливси пытался обучить Джима игре в шахматы. Несмотря на то, что он получал в качестве форы то слона, а то и ладью, Джим часто проигрывал. Как-то раз он решил предложить сыграть в модифицированные правила, в которых ферзь доктора будет иметь неполную силу. Джим назвал силой шахматного ферзя максимальное количество клеток, которые ферзь может держать под боем в одном направлении по горизонтали, вертикали или одной из диагоналей (включая клетку, на которой стоит сам ферзь). Так, обычный ферзь имеет силу 8, а ферзь силы 3, стоящий на поле $d4$ обычной шахматной доски, держит под боем 17 полей.

Идея доктору понравилась, но перед тем, как сыграть первую партию, доктор нарисовал шахматную доску размером $N \times N$, расставил на ней M ферзей заданной силы (не более чем по одному ферзю в клетке), после чего спросил Джима, сколько клеток доски не находятся под боем ни одного из ферзей?

Попробуйте и Вы решить эту задачу.

Формат входного файла

Первая строка входного файла содержит два целых числа M и N ($0 \leq M \leq 10000, 1 \leq N \leq 10000, M \times N \leq 20000000, M \leq N^2$). Каждая из последующих M строк описывает одного ферзя и содержит три целых положительных числа, не превосходящих N — номер горизонтали, номер вертикали (нумерация начинается с единицы) и силу этого ферзя.

Формат выходного файла

Единственная строка выходного файла должна содержать одно число — рассчитанное количество клеток, не находящихся под боем.

Пример

<code>queens.in</code>	<code>queens.out</code>
3 8 4 4 3 8 8 2 2 1 2	38

Задача D. Уголки

Имя входного файла:	<code>corners.in</code>
Имя выходного файла:	<code>corners.out</code>
Ограничение по времени:	2 seconds
Ограничение по памяти:	256 Mebibytes

Добравшись до шахматной доски, в спешке забытой доктором, пираты решили использовать её по назначению — для игры, тем более, что игра в кости уже многим (особенно тем, кто в основном проигрывает) надоела. Только вот правил шахмат и даже шашек они не знали, так что играли шашками в игру, у нас называемую «уголки».

Первоначально на доске некоторым образом расставлено по 12 шашек белого и чёрного цвета так, что каждая шашка находится ровно в одной клетке, и ни в одной клетке не находится более одной шашки. Игроки делают ходы по очереди, передвигая шашки своего цвета (игру начинают белые). Каждый ход состоит либо в передвижении своей шашки на свободную клетку по вертикали или горизонтали, либо в выполнении серии прыжков.

При прыжке шашка «перелетает» через соседнюю по горизонтали или вертикали клетку, занятую шашкой (неважно, своей или противника) и опускается на следующую клетку в этом же направлении, которая должна быть свободной. Во время выполнения серии прыжков можно изменять направление прыжков, но нельзя опускаться дважды на одну и ту же клетку, а также на исходную клетку.

Собравшиеся вокруг доски зрители живо обсуждали происходящее и мешали игрокам, пока хитрый Джон Сильвер, поднаторевший в таких играх, не предложил пари — каждый из зрителей должен, взглянув на текущую позицию, придумать самую длинную серию прыжков, которую может совершить одна из шашек (любого цвета). Под длиной серии понимается количество прыжков в ней. Джон уверен, что он легко найдёт самую длинную серию. А сможете ли Вы сделать это?

Формат входного файла

Входной файл содержит восемь строк по восемь символов. Каждая строка файла соответствует одной строке игрового поля (сверху вниз) и содержит символы W, B или точку, что соответствует белой, черной шашке или пустой клетке соответственно

Формат выходного файла

Первая строка выходного файла должна содержать позицию клетки, из которой можно совершить самую длинную серию прыжков, в стандартной шахматной нотации. Вторая строка содержит единственное число — количество прыжков в такой серии. Если задача допускает несколько решений, выведите то из них, для которого обозначение начальной клетки минимальное в лексикографическом порядке. Если ни одного прыжка сделать нельзя, единственная строка выходного файла должна содержать строку `Impossible`.

Пример

corners.in	corners.out
BBB..... BBB..... BBB..... BBB.....WWWWWWWWWWWW	a6 1
B.B.B.B. BB.B.B.. B.B.B.B. ...W....W.W.WW WW.W.W.. ..W.W.W.	h3 7

Задача Е. Треугольный жребий

Имя входного файла:	<code>triangles.in</code>
Имя выходного файла:	<code>triangles.out</code>
Ограничение по времени:	3 seconds
Ограничение по памяти:	256 Mebibytes

«Кто пойдёт на поиск беглецов?» — грозно спросил Джон Сильвер. Пираты не выражали особого желания: всех привлекало участие в поиске сокровищ, да и с учётом того, что беглецы могли быть вооружены, охота не была особо безопасной. И тут Сильвер решил прибегнуть к жеребьёвке. Вытащив откуда-то N спичек, возможно, с разными длинами, он передал их одному из пиратов. «Делаем так. Сначала ты выбираешь какое-то количество спичек. Если вот он» — Сильвер указал на второго пирата — «сделает из них треугольник, ты отправляешься на поиски беглецов. Нет — отправляется он. Потом я возвращаю выбранные спички в кучу, и тот из вас, кому не выпал жребий идти на поиск, снова выбирает какое-то количество спичек. И если вот этот долговязый тип» — Джон выбрал ещё одного — «не сможет выбрать из нового набора три спички, из которых треугольник сделать нельзя, то вторым пойдёт он. Нет — пойдёт оставшийся из вас!».

Первый пират понял, что если он сумеет выбрать в первом случае такое подмножество B изначального множества спичек A , что никакие три спички из этого подмножества не образуют треугольник, а во втором случае — такое подмножество C того же множества A , что любые три спички из этого подмножества, наоборот, образуют треугольник, то идти ловить беглецов ему не придётся. При этом, чтобы не быть уличённым в жульничестве, пират хочет выбрать B и C так, чтобы количество спичек в каждом из них при сохранении указанных выше свойств было максимальным.

Попробуйте и Вы найти подобное разделение. Учтите, что Сильвер имел в виду невырожденные треугольники. Треугольник считается невырожденным, если каждый из его углов больше 0 и меньше π радиан.

Формат входного файла

Первая строка входного файла содержит значение N ($3 \leq N \leq 10^6$) — количество спичек в изначальном множестве A . Далее следуют N целых положительных чисел, разделенных пробелами и/или переводом строки — длины спичек. Каждое из этих значений не превосходит 10^9 .

Формат выходного файла

Первая строка выходного файла должна содержать количество спичек в подмножестве B , вторая — длины спичек из этого подмножества, разделённые пробелами. Последующие строки выходного файла содержат информацию о подмножестве C в том же формате. Если одно из подмножеств содержит менее трёх элементов, в качестве информации об этом множестве необходимо вывести строку с нулём.

Если задача допускает несколько решений, достаточно вывести любое из них.

Пример

triangles.in	triangles.out
4 7 2 3 2	3 7 2 3 3 2 3 2
4 17 12 13 12	0 4 17 12 13 12
3 17 2 63	3 17 2 63 0

Задача F. Эллиптические заграждения

Имя входного файла: ellipses.in
Имя выходного файла: ellipses.out
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 Mebibytes

Для организации обороны форта капитан Смоллетт приказал соорудить две линии заграждений: внешнюю в виде эллипса, описанного вокруг форта, и внутреннюю, в виде эллипса, вписанного в контур, образованный стенами форта. Доктор Ливси заметил, что для того, чтобы пираты на внешней линии заграждений оказались уязвимы для ружейного огня из форта, площадь внешнего эллипса должна быть минимальной, а для того, чтобы как можно дольше задержать пиратов, сумевших ворваться в форт, площадь внутреннего эллипса должна быть максимальной. Сквайр Трелони заметил, что оси этих эллипсов не обязательно будут параллельны осям координат, так что вычислить все параметры в походных условиях будет затруднительно. Так что от Вас требуется найти оба необходимых эллипса, если форт представляет собой невырожденный треугольник, заданный своими вершинами.

Формат входного файла

Входной файл состоит из трёх строк. Каждая строка этого файла соответствует одной вершине треугольника, образуемого стенами форта, и содержит два целых числа, не превосходящих по абсолютной величине 10^4 — координаты x и y этой вершины.

Формат выходного файла

Первая строка выходного файла должна содержать описание описанного эллипса в виде пяти чисел — координат фокусов и суммы расстояний от фокусов до любой точки эллипса. Фокусы могут быть перечислены в любом порядке. Вторая строка содержит описание вписанного эллипса в том же формате. Числа в выходном файле должны быть заданы с точностью до 0.001.

Пример

ellipses.in	ellipses.out
0 0	2.0000 0.0000 2.0000 2.6667 5.3333
2 4	2.0000 2.0000 2.0000 0.6667 2.6667
4 0	

Задача G. Выпуклые оболочки

Имя входного файла: convex.in
Имя выходного файла: convex.out
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 Mebibytes

Рассматривая карту окрестностей форта, сквайр Трелони заметил, что N высоких деревьев располагаются в точках с целочисленными координатами $(X_1, Y_1), \dots, (X_N, Y_N)$ так, что для произвольных целых чисел i и j ($1 \leq i < j \leq N$), выполняется по крайней мере одно из следующих свойств: первое — $X_i < X_j$, или второе — $X_i = X_j$, при этом $Y_i < Y_j$.

Тогда «зоной контроля» для i деревьев является $CH(i)$ — выпуклая оболочка точек с номерами от 1 до i . Определим $S(i)$ как площадь $CH(i)$. В процессе планирования контрастступления против пиратов капитану Смоллетту потребовалось значение величины $S = \sum_{i=1}^N S(i)$.

От Вас требуется вычислить это значение.

Формат входного файла

В первой строке входного файла записано одно число N ($1 \leq N \leq 256000$). В каждой из последующих N строк записаны два целых числа X_i и Y_i ($-32000 \leq X_i, Y_i \leq 32000$) — координаты точки с номером i .

Формат выходного файла

Единственная строка выходного файла должна содержать искомое значение S , вычисленное с точностью до 0.0001.

Пример

convex.in	convex.out
4	1.5
0 0	
1 0	
1 1	
2 0	

Задача Н. Последовательность из кода Флин-та

Имя входного файла: `sequence.in`
Имя выходного файла: `sequence.out`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 Mebibytes

Осматривая окрестности форта, Джим нашёл вкопанный в землю скелет. В руке он держал закупоренную бутылку с зашифрованным описанием места, в котором и закопан клад.

Использованный Флинтом шифр не был секретом для доктора Ливси. При составлении шифра использовалась следующая последовательность, зависящая от целочисленного параметра x :

$a_1 = x$, $a_{n+1} = a_n/2$, если a_n четное, $a_{n+1} = 3a_n + 1$, если a_n нечетное.

Пусть $f(x)$ — индекс первого элемента последовательности, равного 1. Для расшифровки текста, в частности, необходимо найти сумму, минимум и максимум $f(x)$ по всем значениям x , удовлетворяющим условию $A \leq x \leq B$, где $1 \leq A \leq B \leq 10^9$.

Например, в приведённом в примере случае если $x = 3$, то $a_1 = 3$, $a_2 = 10$, $a_3 = 5$, $a_4 = 16$, $a_5 = 8$, $a_6 = 4$, $a_7 = 2$, $a_8 = 1$, то есть $f(3) = 8$. Если $x = 4$, то $a_1 = 4$, $a_2 = 2$, $a_3 = 1$, то есть $f(4) = 3$. Аналогично, если $x = 5$, то $a_1 = 5$, $a_2 = 16$, $a_3 = 8$, $a_4 = 4$, $a_5 = 2$, $a_6 = 1$, то есть $f(5) = 6$.

Ваша задача — помочь в расшифровке и найти вышеуказанные параметры.

Формат входного файла

В единственной строке входного файла записаны величины A и B .

Формат выходного файла

Выходной файл состоит из трех чисел — суммы, минимума и максимума $f(x)$. Эти числа должны быть отделены друг от друга пробелами и/или символами перевода строки.

Пример

<code>sequence.in</code>	<code>sequence.out</code>
3 5	17 3 8

Задача I. За сокровищем

Имя входного файла:	treasure.in
Имя выходного файла:	treasure.out
Ограничение по времени:	3 seconds
Ограничение по памяти:	256 Mebibytes

Карта Острова Сокровищ представляет собой сетку, состоящую из квадратных клеток. Сетка состоит из M рядов с севера на юг и N — с запада на восток. Форт, в котором первоначально находится поисковая партия в составе сквайра Трелони, доктора Ливси и капитана Смоллетта, расположен так, что единственный выход из него ведёт в клетку, находящуюся в северо-западном углу карты, а сокровища зарыты в клетке, находящейся в юго-восточном углу. Некоторые клетки заполнены лесом, водой или болотом. Партия тратит 1 час на проход по территории, соответствующей одной клетке. Если ей нужно войти в клетку, заполненную лесом, водой или болотом, она должна дополнительно потратить t_1 , t_2 или t_3 часов на преодоление соответствующего препятствия. Наконец, на поиск сокровищ в финальной клетке требуется потратить t_0 часов.

Однако на поиск беглецов с запада отправлен превосходящий её в силах отряд пиратов, так что во избежание встречи с пиратами партия может переходить из той клетки, где она находится, только в одну из трёх соседних клеток на севере, востоке или юге (естественно, не выходя при этом за пределы острова).

Вам требуется определить, какое минимальное время пройдёт от момента выхода партии из форта до нахождения сокровищ.

Формат входного файла

Входной файл содержит несколько строк. В первой строке заданы значения M и N ($1 \leq M, N \leq 1000$). Далее следует строка со значениями t_0 , t_1 , t_2 и t_3 ($0 \leq t_0, t_1, t_2, t_3 \leq 1000$). Третья строка содержит величины k_1 , k_2 , k_3 — количество клеток, заполненных лесом, водой и болотом соответственно ($0 \leq k_1, k_2, k_3; k_1 + k_2 + k_3 \leq M \times N$). Далее следуют k_1 строк с координатами клеток, заполненных лесом, k_2 строк с координатами клеток, заполненных водой и k_3 строк с координатами клеток, заполненных болотом. Крайняя северо-западная клетка имеет координаты $(1,1)$, а юго-западная — (N, M) .

Формат выходного файла

Единственная строка выходного файла должна содержать искомое значение.

Пример

treasure.in	treasure.out
5 5	29
1 5 6 4	
5 6 1	
2 1	
2 2	
1 2	
5 4	
5 5	
5 1	
4 2	
5 2	
4 3	
4 4	
4 5	
1 1	