

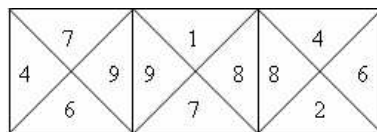
Задача А. Разноцветные плитки

Имя входного файла: `bricks.in`
Имя выходного файла: `bricks.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 Mb

Детектив Элайдж Бейли с Земли вместе со своим напарником, Р. Дэниелом Оливо, взялись за расследование очередного громкого дела на планете Аврора: кем-то был уничтожен уникальный робот. Расследование привело Элайджа и Дэниела в старинный замок, владельцы которого не особо были склонны общаться с землянами и роботами (да, буква Р. в имени Дэниела обозначала слово «робот», хотя, в отличие от основной массы роботов, отличить его от человека было крайне сложно). И тогда Дэниел предложил план: так как в замке идёт ремонт, то он проникает на территорию замка под видом инженера-строителя.

В этот момент в замке шла укладка плитки в одном из главных коридоров. Коридор имел размер $1 \times M$, и его требовалось замостить керамическими плитками. В распоряжении бригады роботов имеется K плиток, каждая из которых представляет собой квадрат размером 1×1 . Диагонали делят этот квадрат на четыре треугольника, каждый из которых окрашен в свой цвет. Цвета треугольников на каждой плитке различны. Схемы раскраски (цвета и их последовательность) могут различаться для разных плиток. В процессе укладки плиток их можно поворачивать на 90, 180 или 270 градусов, но нельзя переворачивать!

Заказчик потребовал от Дэниела уложить плитку так, что цвета треугольников, прилегающих друг к другу, на соседних плитках должны совпадать. При этом цвета треугольников на сторонах первой и последней плитки, соответствующих началу и концу коридора, также определяются заказчиком.



Можно ли найти последовательность плиток, удовлетворяющую этим требованиям?

Формат входного файла

Первая строка входного файла содержит два целых числа M и K ($1 \leq M \leq K \leq 10$). Каждая из последующих K строк описывает схему раскраски одной плитки и содержит четыре натуральных числа, не превосходящих 100 — коды цветов, в которые окрашены треугольники этой плитки, в порядке обхода треугольников по часовой стрелке. Наконец, последняя строка содержит два кода цветов, которыми должны быть окрашены треугольники, повернутые к началу и концу коридора.

Формат выходного файла

Первая строка выходного файла должна содержать строку **Yes**, если коридор можно выложить плитками согласно требованиям заказчика, и **No** в противном случае. В случае положительного ответа вторая строка должна описывать один из возможных вариантов укладки и должна содержать последовательность номеров плиток, расположенных в направлении от начала к концу коридора. Нумерация плиток начинается с единицы и соответствует их описанию во входном файле. Числа в строке выходного файла разделяются одним или несколькими пробелами.

Пример

bricks.in	bricks.out
3 4 1 8 7 9 6 2 8 4 3 2 7 6 9 6 4 7 4 6	Yes 4 1 2
3 5 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 1	No

Задача В. Прокладка канализации

Имя входного файла: `drain.in`
Имя выходного файла: `drain.out`
Ограничение по времени: 10 секунд
Ограничение по памяти: 256 мегабайт

Освоившись в замке, Дэниел быстро выяснил, что некая секретная лаборатория находится глубоко в подвалах замка. Добраться туда было практически невозможно. Если только не изменить план работ так, чтобы подвалы также не избежали реконструкции. Например, реконструировать канализационную систему замка.

Бейли, под видом представителя строительной компании, убедил хозяина замка перестроить в замке систему канализации. Для этого хозяину необходимо переоборудовать одну из комнат на каждом этаже под санузел (и установить в этой комнате канализационный стояк), на каждом из этажей, кроме первого, уложить горизонтальные трубы, соединяющие стояки, а также соединить стояк первого этажа с общественной канализацией, проходящей возле замка.

Замок насчитывает N этажей, на каждом этаже имеется M комнат. Стоимость переоборудования i -й комнаты на j -м этаже составляет $A_{i,j}$, стоимость соединения стояков в комнатах (i, j) и $(k, j-1)$ рассчитывается по формуле $|k-i| \times B_j$. Наконец, стоимость подсоединения к общественной канализации равна $\min(i, M-i+1) \times B_1$, если стояк первого этажа находится в комнате $(i, 1)$.

Рассчитайте минимальную стоимость прокладки канализации.

Формат входного файла

Первая строка входного файла содержит два целых числа M и N ($1 \leq M, N \leq 500$). Каждая из последующих N строк содержит M неотрицательных целых чисел, не превосходящих 10000 — величины $A_{i,j}$. Наконец, последняя строка содержит N чисел из того же интервала — величины B_j .

Формат выходного файла

Единственная строка выходного файла должна содержать одно число — рассчитанную стоимость прокладки канализации в замке.

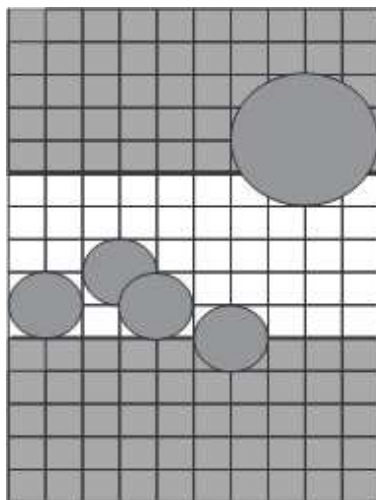
Пример

<code>drain.in</code>	<code>drain.out</code>
4 3 10 7 9 4 2 3 5 8 12 14 6 5 2 4 3	19

Задача С. Подвал

Имя входного файла:	<code>basement.in</code>
Имя выходного файла:	<code>basement.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Оказавшись в подвале, Дэниел осмотрелся. Прямоугольный коридор, по которому ему необходимо было пройти, имеет размеры m метров в ширину и n метров в длину и ограничен с двух сторон вертикальными стенами. В коридоре Дэниел заметил K круглых светящихся пятен. Проведённый им анализ этих пятен показал, что это — особый вид мха с одной из дальних планет, выделяющий очень сильную кислоту, которая опасна не только для людей, но даже для роботов. Пятна располагаются на полу и стенах коридора и могут частично или полностью перекрывать друг друга. Если центр пятна находится на полу рядом со стеной, пятно «наползает» на стену, сохраняя свою круглую форму. Аналогично, пятно, центр которого находится на стене, может частично «спускаться» на пол. Пятна могут частично выходить за пределы коридора. Пример расположения пятен показан на рисунке, приведённом ниже, причём стены обозначены серым цветом, пол коридора — белым.



Дэниел установил, что коридор проходим, если существует такой путь (непрерывная линия) по полу от одного его конца до другого, что минимальное расстояние от любой точки этой линии до пятна мха либо до стены не меньше S .

Вам задан коридор, размеры и расположение пятен мха. Требуется установить, является ли этот коридор проходимым с точки зрения Дэниела, если измерение расстояний между пятнами или от пятна до стены ведётся с округлением до 0.01 (в силу конструктивных особенностей встроенного дальномера).

Формат входного файла

Первая строка входного файла содержит четыре числа m, n, K, S ($1 \leq m, n \leq 100, 0 \leq K \leq 100, 0 < S \leq 5$). Первые три числа целые, четвертое — действительное. Каждая из последующих K строк описывает одно пятно мха и содержит три действительных числа x, y, r — координаты центра пятна и его радиус ($0 \leq x \leq n, -m \leq y \leq 2m, 0 < r \leq 100$). Если $y < 0$ или $y > m$, центр пятна находится на стене. Запись действительных чисел содержит не более двух десятичных цифр в дробной части.

Формат выходного файла

Единственная строка выходного файла должна содержать строку **Yes**, если коридор можно пройти, и **No** в противном случае.

Пример

basement.in	basement.out
5 10 5 1 1 1 1 8 6 2 3 2 1 4 1 1 6 0 1	Yes
5 10 5 1.5 1 1 1 8 6 2 3 2 1 4 1 1 6 0 1	No

Задача D. Древние языки

Имя входного файла: lang.in
Имя выходного файла: lang.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Прошли тысячелетия. Эпоха переселенческих планет сменилась новой экспансией, затем периодом упадка, затем образовалась огромная Империя с центром на Транторе. О роботах уже практически все забыли — развитие техники пошло по другому пути. Но и на Империю надвигался кризис, о котором мало кто подозревал. Но премьер-министр Империи Эдо Демерзель понимал ситуацию прекрасно. Именно поэтому он активно поддержал проект Гэри Селдона.

Математик Гэри Селдон занялся разработкой нового приложения, названного им «психоистория». Он предположил, что исторические события подчиняются определённым закономерностям, которые вполне могут быть вычислены. Для сбора фактического материала, позволяющего провести подобные вычисления, на одну из заброшенных переселенческих планет была отправлена экспедиция.

На языке аборигенов этой планеты числа записываются в позиционной системе счисления. Все используемые аборигенам числа меньше 1000000. 0 и отрицательных чисел они не знали. Есть три слова, обозначающие десятки, сотни, тысячи. В языке использовались ещё девять слов, обозначающие различные цифры. В записи чисел аборигены использовали только именительный падеж и единственное число. Цифра 1 при указании количества десятков, сотен, тысяч не записывается. Например, число 107201 записывается в виде СОТНЯ СЕМЬ ТЫСЯЧА ДВА СОТНЯ ОДИН. Максимальное число (999999) имеет вид: ДЕВЯТЬ СОТНЯ ДЕВЯТЬ ДЕСЯТОК ДЕВЯТЬ ТЫСЯЧА ДЕВЯТЬ СОТНЯ ДЕВЯТЬ ДЕСЯТОК ДЕВЯТЬ.

Экспедиции посчастливилось найти документ, описывающий бюджет одного из местных королевств. Смета содержала записи нескольких чисел. Записи всех чисел корректны. Известно, что документ описывает возрастающую последовательность чисел. Все записи цифр и разрядов выполнены прописными латинскими буквами, и разделены одним или несколькими пробелами.

Вам необходимо помочь экспедиции расшифровать запись, т.е. поставить в соответствие словам цифры и разряды, а строкам — числа так, чтобы последнее записанное число было минимально возможным.

Формат входного файла

В первой строке входного файла записано N — количество чисел (строк) на языке аборигенов. Далее, в каждой из следующих N строк, записаны сами числа по одному в строке. Слова в записи числа разделяются произвольным числом пробелов. Длина строки не более 250 символов, количество строк не больше 100.

Формат выходного файла

В первой строке выходного файла записывается десятичное представление для минимального из возможных значений последнего числа. В следующих строках записано соответствие названий цифр словам в форме СЛОВО=ЦИФРА или СЛОВО=РАЗРЯД, обеспечивающее получение минимального результата. Слова указываются в порядке их появления в тексте задания.

Пример

lang.in	lang.out
8	1211
EIN	EIN=1
ZWEI	ZWEI=10
ZWEI EIN	DREI=100
DREI	FIER=2
DREI ZWEI EIN	FUNF=1000
FIER DREI ZWEI EIN	
FUNF	
FUNF FIER DREI ZWEI EIN	

Задача Е. Имперские налоги

Имя входного файла: `rent.in`
Имя выходного файла: `rent.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Империя Трантора распространила свою власть на многие планеты, при этом оставляя в силе местные законы и обычаи — естественно, обращая их себе на пользу. Так, на одной из планет была принята следующая система земельных налогов. Сельскохозяйственные площади на этой планете располагались на искусственном острове в виде прямоугольника. Территория острова разбита на N горизонталей и M вертикалей так, что получается $M \times N$ участков. Все «граничные» участки, примыкающие к морю, и некоторое количество «внутренних» участков принадлежат центральному правительству, и на них налог не начисляется. На остальных же участках налог равен квадрату расстояния до ближайшего участка, принадлежащего правительству. Более точно, размер налога для участка с координатами (x, y) равен минимуму на всех правительственных участках с координатами (a, b) функции $(x - a)^2 + (y - b)^2$.

К сожалению, экономика этой планеты находилась в упадке, и владельцы участков выплачивали налоги крайне нерегулярно. Да и правительству от его собственности перепало не так много... Имперский наместник сдал некоторые внутренние (не находящиеся на периметре) участки в аренду, взимая по описанным выше правилам налог с арендаторов (если арендован правительственный участок, арендатор за него налога не платит).

	1	2	3	4	5	6	7	8	9	10
1	*	*	*	*	*	*	*	*	*	*
2	*									*
3	*									*
4	*	4					2			*
5	*									*
6	*									*
7	*			5				3		*
8	*									*
9	*	1								*
10	*	*	*	*	*	*	*	*	*	*

На рисунке серые клетки — участки, находящиеся в частном владении, белые — участки, находящиеся в собственности правительства, цифрами обозначены арендованные участки; ближайшие клетки для участков, арендованных у частных владельцев, обозначены штриховкой; граничные клетки обозначены '*'.
Вам требуется написать программу, которая рассчитывает налог с каждого из арендованных участков.

Формат входного файла

Первая строка входного файла содержит два целых числа N и M ($3 \leq N, M \leq 2007$). Далее следуют N строк по M символов в каждой, описывающих правительственные и частные участки. Символ «точка» означает, что участок принадлежит правительству, а символ # соответствует участку, находящемуся в частной собственности. В следующей строке записывается величина K — количество арендованных участков ($1 \leq K \leq 20007$). Наконец, следующие K строк описывают координаты арендованных участков и содержат номер горизонтали и вертикали для каждого из участков.

Формат выходного файла

В выходной файл выведите K строк. В каждой строке должно быть одно число — налог с соответствующего арендатора. Учтите, что за аренду правительственного участка налог равен нулю.

Пример

rent.in	rent.out
10 10	1
.....	2
.#####.	4
..##.##..	0
.....##.	0
.....##.	
.#.#####.	
.#...###.	
.#...###.	
.#####.	
.....	
5	
9 2	
4 7	
7 8	
4 2	
7 4	

Задача F. Красивые последовательности

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

Исследования Гэри Селдона по психоистории показали, что годы, в которых Империи угрожают кризисные ситуации, образуют «красивую последовательность».

Последовательность целых чисел $1 \leq a_1 < a_2 < \dots < a_k \leq n$ Селдон назвал «красивой», если для любого i ($1 \leq i \leq k$) величина $|a_i - i|$ будет чётным числом. Премьер-министр Эдо Демерзель для вычисления количества возможных вариантов развития Империи поручил Вам определить количество различных «красивых» последовательностей для заданных величин k и n ($1 \leq k \leq n \leq 10000$). Так, например, в первом примере такая последовательность одна — (1, 2), а во втором их две: (1) и (3).

Формат входного файла

Единственная строка входного файла содержит величины k и n .

Формат выходного файла

Единственная строка выходного файла должна содержать десятичную запись искомого числа «красивых» последовательностей.

Пример

<code>sequence.in</code>	<code>sequence.out</code>
2 3	1
1 3	2

Задача G. Освоение Терминуса

Имя входного файла: `terminus.in`
Имя выходного файла: `terminus.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Прогнозы по поводу кризиса Империи оказались верными. Однако, согласно плану Гэри Селдона, сохранялся шанс сохранить достижения цивилизации и достаточно быстро восстановить транспортное и экономическое единство Галактики. Основной частью этого плана было создание на планете Терминус, находящейся на периферии старой Империи, Первого Основания, вокруг которого спустя столетия будет сформирована Вторая империя.

Первоначально колонизация Терминуса было обосновано проектом Галактической Энциклопедии — было объявлено, что учёные, которым было поручено составление Энциклопедии, во избежание вовлечения в различные внутриимперские конфликты будут работать в поселении на Терминусе.

В процессе строительства поселения вездеход вышел на участок поверхности, представляющий собой прямоугольник размерами $m \times n$, разделённый вдоль стороны длины m на три равные части $m/3 \times n$. Ближняя части участка (которому принадлежит точка $(0,0)$ — начальное положение вездехода) покрыта песком, в средней части расположен водоём со стоячей водой, дальняя от вездехода часть представляет собой участок местной степи, иссечённый рытвинами. Вездеходу необходимо оказаться в точке с координатами (m, n) как можно скорее. Маршрут вездехода выглядит следующим образом: он стартует в точке с координатами $(0,0)$, доезжает по песку до любой точки, расположенной на ближнем берегу водоёма, переплывает водную преграду так, чтобы выйти в любой точке противоположной стороны, и едет по степи так, чтобы оказаться в точке (m, n) .

Требуется найти минимальное время, за которое вездеход окажется в точке (m, n) , если известна скорость его движения по песку, вплавь и по пересечённой местности соответственно.

Формат входного файла

Первая строка входного файла содержит целые величины n и m ($1 \leq n, m \leq 10000$), при этом m делится на 3. Вторая строка содержит скорости вездехода по песку, по воде и по пересечённой местности — три положительных действительных числа, каждое из которых не превосходит 10000.

Формат выходного файла

Выходной файл содержит единственное число — искомое время, рассчитанное с точностью до 10^{-3} .

Пример

<code>terminus.in</code>	<code>terminus.out</code>
1 9 0.001 10000 0.001	6000.000316

Задача Н. Торговые операции

Имя входного файла: `numbers.in`
Имя выходного файла: `numbers.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

С течением времени поселение на Терминусе развивалось, и в какой-то момент существенное влияние приобрели торговые кланы. Их конкуренция оказывала существенное влияние на развитие Первого Основания, приводя к экспансии на окрестные планеты.

Для своих записей торговцы использовали «сокращённую» систему записи. В ней информация о некотором множестве S целых неотрицательных чисел (индексов товаров), каждое из которых меньше 100000, записывается в одну строку по следующим правилам:

- строка содержит одно или несколько описаний групп элементов, разделенных запятой;
- описание группы содержит от 1 до 5 символов — десятичных цифр или символа ‘?’ (вопросительный знак). Если вопросительного знака нет, группа соответствует одному числу — элементу множества. В противном случае группа соответствует всем числам, которые можно получить заменой каждого из вопросительных знаков на любую цифру.
- в каждом описании могут быть лидирующие нули, как записанные явно, так и образующиеся в результате замены вопросительного знака (например, описание «?6» соответствует числам 6, 16, 26, ..., 86, 96).

Вам в руки попала такая запись, описывающая некую сделку. Чтобы оценить объём сделки, Таможенная служба Терминуса просит Вас подсчитать мощность множества S , записанного таким образом.

Формат входного файла

Единственная строка входного файла содержит описание множества S согласно приведённым выше правилам. Длина строки не превосходит 100 символов.

Формат выходного файла

Единственная строка выходного файла должна содержать искомую мощность множества S .

Пример

<code>numbers.in</code>	<code>numbers.out</code>
?6,9?,22	20
???	1000
0,00,0?	10

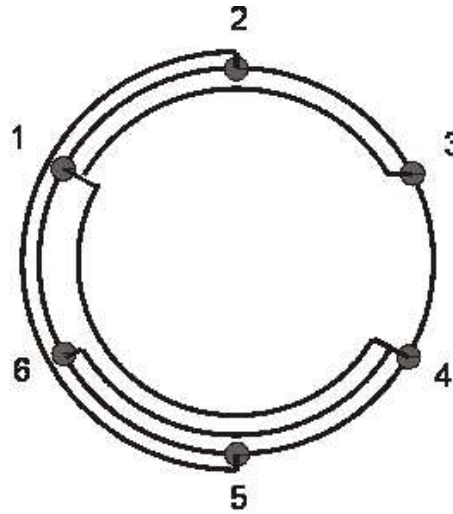
Задача I. Метро на Транторе

Имя входного файла: metro.in
Имя выходного файла: metro.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Для Второго Основания Гэри Селдоном была выбрана бывшая столица Империи Трантор. В отличие от Первого Основания, Второе представляло собой нечто типа «мозгового центра», состоящего из учёных, которые бы продолжали исследования Селдона и незаметно контролировали развитие Первого Основания. Для Второго Основания была выбрана часть Трантора, представляющая собой небольшой город, стоящий на N холмах. Город, по сути, является совокупностью из N кварталов, отделённых друг от друга оврагами. Линия метро, построенная в этом городе, была кольцевой и соединяла станции, которые расположены по одной в каждом квартале. Схема этой линии представляла собой круг.

В эпоху упадка Империи из-за малочисленности ресурсов развитие метро шло не за счёт строительства новых станций, а за счёт прокладки так называемых «хордовых» линий. Каждая хордовая линия соединяет напрямую две станции, которые не являются соседними по кольцевой линии. Между двумя конкретными станциями может быть проведено не более одной хордовой линии.

Психологи, входящие в состав Второго Основания, заявили, что круг — это идеальный для восприятия способ отобразить на схеме метро кольцевую линию. Более того, требуется, чтобы изображения хордовых линий на схеме не пересекались между собой. Правда, изображения отдельных хордовых линий не обязаны быть отрезками прямой и находиться внутри круга, обозначающего кольцевую линию. (В частности, для первого примера к задаче решением является схема, показанная на рисунке).



Ваша задача — определить, можно ли для заданной конфигурации метро нарисовать подобную схему?

Формат входного файла

Первая строка входного файла содержит N — число кварталов и K — число построенных хордовых линий ($4 \leq N \leq 10000, 0 \leq K \leq 10000$). Каждая из последующих K строк описывает одну хордовую линию и содержит два числа a и b ($1 \leq a, b \leq N$) — номера станций, которые соединяет эта линия. Кольцевая линия метро проходит через станции $1, 2, \dots, N-1, N, 1$.

Формат выходного файла

Единственная строка выходного файла должна содержать текст **Yes**, если требуемую схему можно построить, и **No** в противном случае.

Пример

metro.in	metro.out
6 4 3 1 2 5 4 1 4 6	Yes
6 4 3 6 2 5 4 1 4 6	No