

Задача А Сад пчел

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

В саду Бена расположено n ульев. Некоторые из них соединены друг с другом или с домом Бена прямыми дорожками. Бен гуляет только по дорожкам, переходя с одной на другую только возле очередного улья. Ни дом Бена, ни какой-либо улей не расположены на дорожке, соединяющей другие два улья. Дорожки организованы таким образом, что Бен может дойти от дома до любого улья только единственным способом.

Каждую неделю Бен делает обход ульев. Он начинает от своего дома и идет по дорожкам, посещая каждый улей по крайней мере один раз, минимизируя при этом общий путь. Сын Бена решил помочь своему стареющему отцу и проложить еще одну прямую дорожку между двумя ульями, так чтобы путь Бена от дома через все ульи с возвращением к дому стал как можно короче. Она, как и старые дорожки, проходить мимо дома или другого улья не должна.

Входные данные

На вход сначала подается число n — количество ульев ($1 \leq n \leq 200$).

Введем координаты так, что дом Бена будет располагаться в точке $(0, 0)$. В следующих n строках входных данных записаны координаты ульев в саду Бена. Они не превосходят 10000 по абсолютному значению. Никакие два улья не совпадают и нет ульев, расположенных в точке $(0, 0)$. Будем считать, что они пронумерованы от 1 до n .

Следующие n строк описывают дорожки — каждая дорожка описывается номерами объектов, которые она соединяет. Дом Бена имеет номер 0.

Выходные данные

Выведите два числа — номера объектов, которые надо соединить дополнительной дорожкой. Если требуемой прямой дорожки, сокращающей общий путь Бена не существует, то выведите -1 .

Пример

Входные данные	Выходные данные
3 1 0 1 1 0 1 0 2 1 2 1 3	0 3

Задача В Преобразование матрицы

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Введем следующие операции над прямоугольной таблицей символов. Пусть нам дана матрица A , состоящая из m строк (первый индекс) и n столбцов (второй индекс). Определим результирующую матрицу B для каждой из операций следующим образом:

- Транспозиция относительно главной диагонали (код операции '1'): $B_{j,i}=A_{i,j}$
- Транспозиция относительно другой диагонали (код операции '2'): $B_{n-j+1,m-i+1}=A_{i,j}$
- Горизонтальное отражение (код операции 'H'): $B_{m-i+1,j}=A_{i,j}$
- Горизонтальное отражение (код операции 'V'): $B_{i,n-j+1}=A_{i,j}$
- Поворот на 90 ('A'), 180 ('B'), или 270 ('C') градусов по часовой стрелке: $B_{j,m-i+1}=A_{i,j}$ (для 90°)
Поворот на 90 ('X'), 180 ('Y'), или 270 ('Z') градусов против часовой стрелки: $B_{n-j+1,i}=A_{i,j}$ (для 90°)

Вам дана последовательность не более 100000 операций над исходной матрицей. Выведите полученную в результате этих операций матрицу.

Входные данные

В первой строке входных данных находятся значения m и n ($0 < m, n \leq 300$). В каждой из следующих m строк находятся по n печатных символов (т.е. не используются символы с кодами от 33 до 126).

Последняя строка содержит описание операций, которые были выполнены над этой матрицей, путем записи без пробелов из кодов. Операции выполняются по порядку слева направо.

Выходные данные

Выведите сначала два целых числа — число строк и столбцов в результирующей матрице. Затем выведите саму матрицу в том же формате, что и во входных данных.

Пример

Входные данные	Выходные данные
3 4 0000 a0b0 cdef A1	3 4 cdef a0b0 0000

Задача С Крестный отец

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

В последние годы Чикагская мафия сильно активизировалась и шеф полиции решил арестовать лидеров мафиозного клана. К несчастью, структура Чикагской мафии устроена достаточно сложно. Известно, что к мафии относятся n человек. Полиция отследила, кто из них с кем контактирует. Основываясь на собранных данных, шеф полиции предположил, что иерархия в мафии представляет собой дерево. Крестный отец мафии находится в корне этого дерева. В целях конспирации мафиози контактируют только со своими непосредственными боссами (предок в дереве) или подчиненными (потомки в дереве).

Полиция знает все контакты, но не знает, кто босс, а кто подчиненный в каждой паре. Крестный отец также не известен. Шеф полиции сделал еще одно предположение, что Крестный отец — это такой член мафии, удаление которого из дерева коммуникаций, приведет к распаду ее на несколько связанных кусков, а максимальный размер оставшегося связным куска структуры будет как можно меньше. Помогите полиции вычислить потенциальных Крестных отцов.

Входные данные

Первая строка входных данных содержит число n членов мафии ($2 \leq n \leq 50000$). Они пронумерованы от 1 до n . Следующие $n - 1$ строка содержат описание замеченных контактов. Пара a_i, b_i означает, что гангстер a_i общается с гангстером b_i .

Гарантируется, что система контактов образует дерево.

Выходные данные

Выведите номера всех гангстеров, которые могут быть Крестным отцом мафии. Номера выводите в возрастающем порядке, разделяя их пробелом.

Пример

Входные данные	Выходные данные
6 1 2 2 3 2 5 3 4 3 6	2 3

Задача D Даша и кризис

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Даша имеет n ювелирных украшений. Каждое украшение имеет стоимость w_i и значимость для Даши v_i . В связи с финансовым кризисом Даша решила продать некоторые украшения и сохранить только k из имеющихся. Чтобы решить, что именно сохранить, Даша вводит параметр *важности* для набора из выбранных k украшений, который вычисляет по следующей формуле:

$$\frac{\sum_{j=1}^k v_j}{\sum_{j=1}^k w_j}$$

Даша решает сохранить такие k украшений, для которых параметр важности будет максимально возможным. Помогите ей правильно выбрать украшения.

Входные данные

Первая строка ввода содержит числа n – количество ювелирных изделий у Даши и k – количество ювелирных изделий, которые планируется оставить ($1 \leq k \leq n \leq 100\,000$).

В следующих n строках содержатся по два числа – v_i и w_i ($0 \leq v_i \leq 10^6$, $1 \leq w_i \leq 10^6$, обе суммы всех значений v_i и w_i не превосходят 10^7 каждая).

Выходные данные

Выведите k чисел – номера ювелирных украшений, которые следует оставить. Если существует несколько решений, то выведите любое из них.

Пример

Входные данные	Выходные данные
3 2 1 1 1 2 1 3	1 2

Задача Е Кольца в тарелке

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Требуется из бумажной тарелки радиуса P вырезать два кольца. Определите, можно ли это сделать.

Входные данные

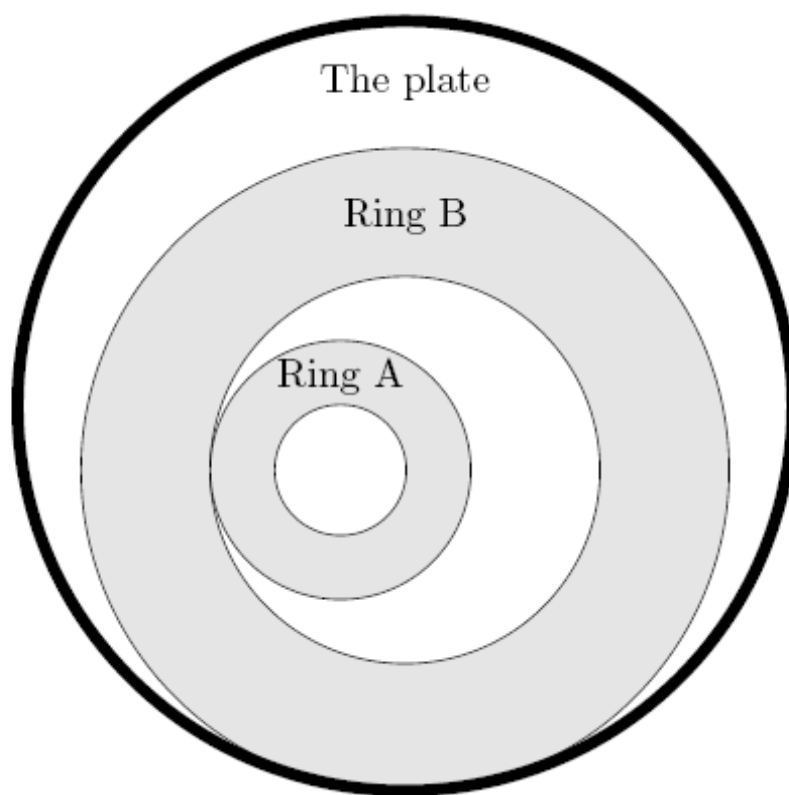
Входные данные содержат 5 целых чисел A, a, B, b, P ($0 < A; a; B; b; P \leq 1\,000\,000$, $a < A$ и $b < B$). Здесь A и B внешние радиусы колец, а a и b их внутренние радиусы соответственно, P – радиус тарелки.

Выходные данные

Выведите Yes или No в зависимости от ответа на вопрос задачи.

Примеры

Входные данные	Выходные данные
2 1 5 3 6	Yes



Задача F Кеннинги

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Кеннинг – это форма поэтической метафоры в древности, когда одно слово заменяется словосочетанием (двумя или более словами). Например, “giver of the gold” – это кеннинг для слова “warrior”. Причем нет разницы в использовании слова и его кеннинга: “poor giver of the gold” и “poor warrior” – это одно и то же. Кеннинги могут быть вложенными. Так, “serpent’s lair” означает “gold”, поэтому “giver of the serpent’s lair” означает “warrior”.

Для зачета вам надо создать достаточно длинный текст. Это можно сделать быстро, имея план текста и список кеннингов. Будем использовать следующий алгоритм. Если текст уже достаточно длинный, то задание выполнено. В противном случае одновременно заменяем все слова в тексте, имеющие кеннинги на соответствующие кеннинги, затем алгоритм повторяется снова.

Входные данные

В первой строке входных данных содержатся 3 числа: ширина результирующего текста w ($1 \leq w \leq 255$), минимальное число непробельных символов в тексте l ($1 \leq l \leq 3\,000$) и число кеннингов в списке n ($1 \leq n \leq 380$). Далее следует список кеннингов, по одному в строке. Каждая строка сначала содержит заменяемое слово, а за ним следует соответствующее словосочетание. В конце входных данных в одной или нескольких строках содержится план текста.

Каждый кеннинг содержит по крайней мере 2 слова (т.е. одна строка содержит не менее трех слов). Кеннинги могут быть рекурсивными. Например, кеннинг для слова “GNU” может быть таким “GNU is Not UNIX”. Кеннинги чувствительны к грамматическим формам и даже к регистру букв, так слова “warrior”, “Warrior” и “warriors” различны и могут иметь разные кеннинги.

Входной файл не превосходит 3 000 байт и содержит только английские буквы, подчеркивания, пробелы, переводы строки и, в первой строке, цифры. Одно слово имеет только один кеннинг. Все слова состоят не более чем из w символов.

Соседние слова разделены в точности одним пробелом или переводом строки. Ни в одной из строк нет ведущих или хвостовых пробелов.

Выходные данные

Если алгоритм не даст результата, то выведите слова “No result” в единственной строке.

В противном случае выведите полученный текст, размещая не более w символов (включая пробелы) в строке. Все переводы строки из первоначального плана должны быть сохранены. Перед очередным словом надо вставить перевод строки, если оно не уместится в текущей строке. Соседние слова должны быть разделены ровно одним пробелом. В строках не должно быть ведущих и хвостовых нулей. В корректном тексте не будет больше 10 000 байт.

Пример

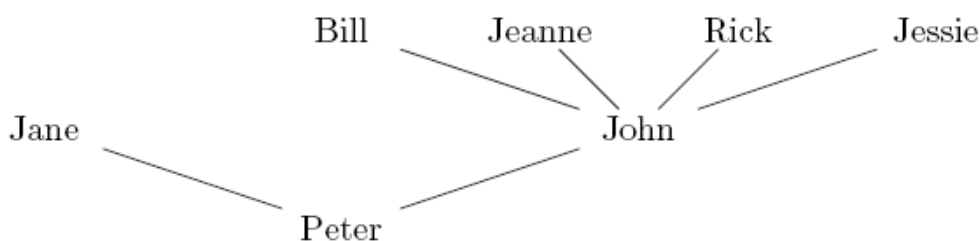
Входные данные	Выходные данные
21 103 7 king hosts leader vessel windless bay of horns horns bulls spears spears war needles Sudden Fate caught death of doomed to death Death It was the end Sudden death of Fjolner in the house of Frodi Death of the king in the vessel	Fate caught doomed to death Fjolner in the house of Frodi It was the end of the hosts leader in the windless bay of bulls spears

Задача G Генеалогическое дерево

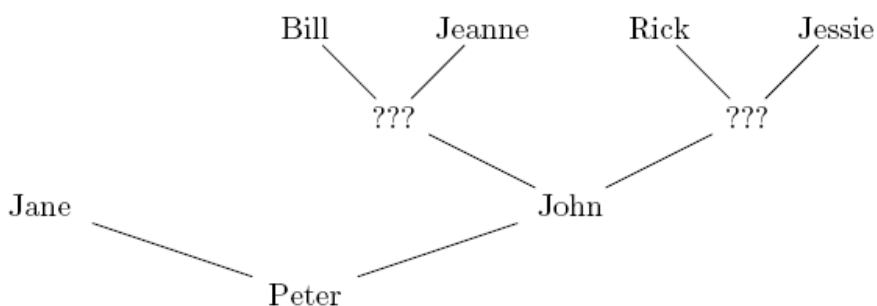
Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Инопланетянин Петр занялся составление своего генеалогического дерева. На его планете каждый происходит от d родителей. А в полученной бета-версии дерева, некоторые его предки имели больше родителей. Петя решил, что некоторые вершины дерева, имеющие отношение родитель–ребенок, на самом деле относятся как предок–потомок. Петя решил подсчитать, сколько предков в его дереве не хватает, чтобы сделать его правильным. Дерево называется правильным, если каждый инопланетянин в нем имеет не более d родителей и каждый инопланетянин упоминается в дереве только один раз..

Например, если $d = 2$, и в бета-версии генеалогическое дерево выглядит так



то в него надо добавить по крайней мере двух предков, чтобы оно стало правильным:



Входные данные

Обозначим Петиних предков из бета-версии номерами от 1 до n (а Петя будет иметь номер 0).

Первая строка входных данных содержит числа n и d ($2 \leq n \leq 100\,000$, $2 \leq d \leq n$). Следующая строка содержит n чисел: i -е число есть номер ребенка у i -го инопланетянина.

Выходные данные

Выведите минимальное число предков, которое надо добавить в дерево, чтобы оно стало правильным.

Примеры

Входные данные	Выходные данные
6 2 5 5 0 5 0 5	2

Задача Н Фен шуй

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Фен шуй – это древняя китайская практика размещения предметов в пространстве для достижения гармонии. В ней говорится, что пол не должен быть пустым, поэтому Фил разместил на полу два одинаковых круглых ковра (по фен шуй надо избегать прямых линий и острых углов).

К сожалению, ими нельзя покрыть пол комнаты полностью, так как она имеет форму выпуклого многоугольника. Фил хочет минимизировать непокрытую часть пола, располагая ковры оптимальным образом.

Помогите ему расположить два ковра в комнате так, чтобы общая покрытая площадь была максимальна. Ковры можно накладывать друг на друга, но их нельзя загибать и резать.

Входные данные

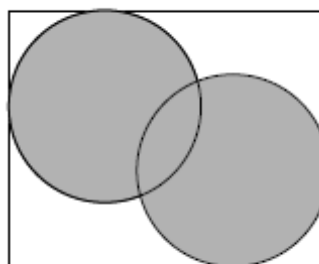
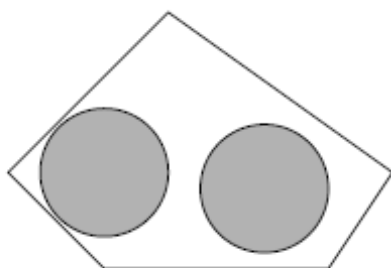
Первая строка входных данных содержит два целых числа n и r — число углов в комнате ($3 \leq n \leq 100$) и радиус ковров ($1 \leq r \leq 1000$, оба ковра имеют один и тот же радиус). Следующие n строк содержат по два числа x_i и y_i — координаты i -го угла ($-1000 \leq x_i, y_i \leq 1000$). Углы описаны в порядке обхода комнаты по часовой стрелке. Координаты углов различны и соседние стены не коллинеарны.

Выходные данные

Выведите x_1, y_1, x_2, y_2 , где (x_1, y_1) и (x_2, y_2) обозначают центры ковров. Координаты должны иметь точность не менее 4 цифр после точки. Выведите любое оптимальное расположение. Входные данные гарантируют, что решение существует (Фил не стал бы покупать ковер, который не помещается в комнату).

Пример

Входные данные	Выходные данные
5 2 -2 0 -5 3 0 8 7 3 5 0	-2 3 3 2.5
4 3 0 0 0 8 10 8 10 0	3 5 7 3



Задача I Форд Транкинкс и язык Велулу

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Мистер Форд Транкинкс, известный археолог, недавно обнаружил в самом сердце Африки остатки загадочного поселения. Через несколько недель исследования он и его коллеги поняли, что они нашли нечто необычное. Племена Велулу, которые жили там 20 000 лет назад, по-видимому, достигли высокого уровня развития. У них даже был алфавит! Но большинство из них стало жертвой ледникового периода и все их культурные достижения были утрачены. Только несколько счастливиц из племени выжили и попытались возродить исчезнувшую цивилизацию. Теперь они известны под именем грозных зулулов, но об этом в другой раз...

Нынешняя задача состоит в расшифровке текстов Велулу. Основная проблема заключается в том, что в их алфавите нет пробелов. Поэтому все тексты слипаются в одну последовательность букв, которые сложно разобрать.

К счастью, археологи уже разработали проект словаря велульского языка. Конечно же они знают о том, что можно обработать последовательность букв на компьютере и получить слова, имея в своем распоряжении словарь. Однако, воспользовавшись таким методом они обнаружили, что огромное количество таких последовательностей почти для любого текста разумного размера. Они не знают, является ли это проблемой метода обработки или особенностью велульского языка. Поэтому они придумали другой метод, который базируется не только на словаре, но и на порядке частей речи в предложении.

Теперь у них был не только словарь, но и предположение о том, как предложение строится в велульском языке. Вам предстоит определить, сколькими способами можно разбить текст на слова, следуя этим правилам, и вывести один из примеров разбиения.

Входные данные

Вам будет дан словарь, правила построения предложений и текст для расшифровки. Для каждого слова определено, какой частью речи оно может быть.

Первая строка входного файла содержит числа N , M и K , где $1 \leq N \leq 5000$ — количество слов в велульском словаре, $1 \leq M \leq 10$ — количество правил построения предложения, а $1 \leq K \leq 10$ — количество различных частей речи.

В велульском языке не так много букв. Так что археологи закодировали их маленькими латинскими буквами. В каждой из N следующих строк содержится слово (не длиннее 20 символов), затем число k_i ($1 \leq k_i \leq 10$) — количество частей речи, которыми может быть это слово, а затем k_i чисел a_{ij} обозначающие допустимые части речи. Все числа a_{ij} даны в возрастающем порядке. Слова в словаре даны в произвольном порядке (не успели упорядочить). Каждое слово встречается в словаре только один раз.

Следующие M строк содержат правила конструирования предложений. Каждое правило описывается количеством слов в предложении l_i ($1 \leq l_i \leq 10$) и, затем, перечислены l_i чисел, являющихся идентификаторами частей речи. Правила не повторяются.

Последняя строка ввода содержит текст, который надо расшифровать. Текст не пуст и его длина не превышает 1000 символов.

Выходные данные

Первая строка должна содержать количество вариантов расшифровки. Если их больше чем 10^{18} выведите строку «TOO MANY» вместо количества.

Если существует корректное разбиение текста на слова, то выведите одно из них в следующей строке. Исходный текст требуется разделить пробелами и точкой с пробелом для получения набора корректных предложений, каждое из которых удовлетворяет правилам, а все слова содержатся в словаре. Точка должна стоять сразу за окончанием предложения, а все слова должны быть разделены пробелами. Весь текст должен быть разбит на предложения. Внимательно посмотрите пример.

Если вариантов несколько — выведите любой из них.

Примеры

Входные данные	Выходные данные
5 2 2 ba 1 2 za 2 1 2 a 2 1 2 caba 1 1 ab 1 1 2 1 2 3 2 2 1 abazabacaba	2 ab a. za ba caba.

Задача J Японский кроссворд

Максимальное время работы на одном тесте:	2 секунды
Максимальный объем используемой памяти:	64 мегабайта

Наиболее известная игра, дошедшая до нас из Японии – это Судоку. Но новая игра должна затмить ее славу. Про нее известно следующее. Нам дан квадрат, разделенный сеткой на $n \times n$ клеток, а каждая клетка содержит картинку одного из k типов. Игрок должен переместить их, чтобы получить максимально возможное число одинаковых первых рядов (два ряда считаются одинаковыми, если оба заполнены одинаковыми картинками и в одинаковом порядке). По виду таблицы определите, сколько одинаковых рядов в ней можно сложить (если менять картинки как угодно).

Например, если нам дана такая таблица:

≡	≡	+
≡	*	*
Δ	*	Δ

одно из результирующих состояний в игре будет

≡	*	Δ
≡	*	Δ
+	≡	*

Входные данные

Первая строка входных данных содержит два числа n ($1 \leq n \leq 40\,000$) и k ($1 \leq k \leq 50\,000$). Каждая из следующих k строк содержит число картинок в таблице каждого из k типов. Все числа больше 0, их сумма в точности равно n^2 .

Выходные данные

Выведите в первой строке максимальное количество одинаковых рядов, которые можно построить из этих картинок. В следующих n строках выведите содержание таких рядов: в каждой строке должно находиться одно число – номер соответствующей картинке в порядке ее появления во входных данных. Если решений несколько, выведите любое из них.

Пример

Входные данные	Выходные данные
3 4	2
3	1
3	2
2	3
1	