

Problem A. Путь

Input file: path.in
Output file: path.out
Time limit: 2 секунды
Memory limit: 256 мегабайт

В руки Тёмы, сотрудника Института Деления и Отрезания (ИДиОт), попал путь на карте. *Путь* — это упорядоченная последовательность точек карты, последовательно соединённых направленными дугами. Путь называется *простым*, если в нём нет повторяющихся точек. Тёме требуется выбрать подпоследовательность дуг этого пути так, чтобы она образовывала путь (при этом дуги будут идти в том же порядке, что и в исходном пути), который ведёт из той же начальной точки и заканчивается в той же конечной точке, а также является простым.

Разрешается не вырезать ни одной дуги, если путь и так является простым.

Input

Первое число входного файла — N ($2 \leq N \leq 100\,000$), количество точек в пути. Затем следует описание пути: точки на карте в порядке обхода. Для простоты, точки задаются своими уникальными идентификаторами — натуральными числами, не превышающими 100 000. Гарантируется, что начало и конец пути не совпадают. Дуга может соединять точку с самой собой. Между двумя точками может быть несколько дуг в разных местах пути.

Все числа во входном файле разделяются переводами строк.

Output

Выведите количество точек в получившемся пути и самый простой путь в том же формате, что и во входном файле. Если простой путь можно сделать несколькими способами, разрешается выводить любой из них.

Example

path.in	path.out
6	4
1	1
3	3
8	8
5	6
8	
6	

Problem B. Компоненты связности

Input file: connect.in
Output file: connect.out
Time limit: 2 секунды
Memory limit: 256 мегабайт

Вам задан неориентированный граф с N вершинами и M рёбрами ($1 \leq N \leq 20\,000$, $1 \leq M \leq 200\,000$). В графе отсутствуют петли и кратные рёбра.

Определите компоненты связности заданного графа.

Input

Граф задан во входном файле следующим образом: первая строка содержит числа N и M . Каждая из следующих M строк содержит описание ребра — два целых числа из диапазона от 1 до N — номера концов ребра.

Output

На первой строке выходного файла выведите число L — количество компонент связности заданного графа. На следующей строке выведите N чисел из диапазона от 1 до L — номера компонент связности, которым принадлежат соответствующие вершины. Компоненты связности следует занумеровать от 1 до L произвольным образом.

Example

connect.in	connect.out
4 2	2
1 2	1 1 2 2
3 4	

Problem C. Поиск цикла

Input file: cycle.in
Output file: cycle.out
Time limit: 2 секунды
Memory limit: 64 мегабайта

Дан ориентированный невзвешенный граф. Необходимо определить есть ли в нём циклы, и если есть, то вывести любой из них.

Input

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N \leq 100\,000$, $M \leq 100\,000$) — количество вершин и рёбер в графе соответственно. Далее в M строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

Output

Если в графе нет цикла, то вывести «NO», иначе — «YES» и затем перечислить все вершины в порядке обхода цикла.

Examples

cycle.in	cycle.out
2 2	YES
1 2	1 2
2 1	
2 2	NO
1 2	
1 2	

Problem D. Кракозябрики

Input file: bfs.in
Output file: bfs.out
Time limit: 1 секунда
Memory limit: 16 мегабайт

... мораль сей басни такова — у нас
хорошая трава.

Парад параллелей ЛКШ.Зима

Кракозябрик Пушистик живёт в столице Кракозябрии — Финденляндии, а его лучший друг детства — в деревне Менделеевка. Кракозябрики очень дружелюбные существа, поэтому в большие праздники у них принято поздравлять старыхдобрых друзей лично.

Наступает День Рождения Короля Кракозябрии — Мирикундина Первого, праздник будет отмечаться по всей стране, а Пушистик, возможно, даже не встретится со своим другом. Всё дело в том, что Пушистик — известный музыкант, он играет на волшебных доремифасольчиках, и его график гастролей расписан буквально по часам. Но он никогда себе не простит, если он не встретится с другом детства в такой день!

Пушистик внимательно изучил, какие самолёты летают в Кракозябрии и теперь на основании этих данных хочет узнать, какое минимальное количество раз ему придется взлетать на самолёте, чтобы в итоге увидеться с лучшим другом детства. Пожалуйста, помогите ему в этом.

Input

В первой строке входного файла содержится три натуральных числа N , S и F ($1 \leq S, F \leq N \leq 100$) — количество городов в Кракозябрии и номера, которыми обозначены Финденляндия и Менделеевка. Далее в N строках заданы все самолётные рейсы страны. Если значение в j -м элементе i -й строки равно 1, то в Пушистик может напрямую лететь из города номер i в город номер j .

Output

Выведите одно целое число — минимальное количество самолётов при полёте из Финденляндии в Менделеевка. Если пути не существует и старым добрым друзьям так и не суждено увидеться, выведите 0.

Example

bfs.in	bfs.out
4 4 3 0 1 1 1 1 0 1 0 1 1 0 0 1 0 0 0	2

Problem E. Снова про коней

Input file: knight3.in
Output file: knight3.out
Time limit: 2 секунды
Memory limit: 64 мегабайта

На шахматной доске 8×8 указаны две несовпадающие клетки. Найдите кратчайший маршрут коня из первой клетки во вторую.

Input

Во входном файле записаны координаты двух клеток. Каждая координата представлена двумя символами, где

сначала указана одна строчная буква от **a** до **h**, а после буквы (без пробела) цифра от 1 до 8, например **h8**. Каждая клетка записана в отдельной строке.

Output

Программа должна вывести последовательность клеток, первая из которых совпадает с первой данной, а последняя совпадает со второй данной. Две соседние клетки должны быть соединены ходом коня, при этом количество клеток в последовательности должно быть минимально возможным.

Examples

knight3.in	knight3.out
a1 b1	a1 b3 d2 b1