

Задача А: Лабиринт знаний

В Летней Компьютерной Школе (ЛКШ) построили аттракцион "Лабиринт знаний". Лабиринт представляет собой N комнат, занумерованных от 1 до N , между некоторыми из которых есть двери. Когда человек проходит через дверь, показатель его знаний изменяется на определенную величину, фиксированную для данной двери. Вход в лабиринт находится в комнате 1, выход - в комнате N . Каждый ученик проходит лабиринт ровно один раз и попадает в ту или иную учебную группу в зависимости от количества набранных знаний (при входе в лабиринт этот показатель равен нулю). Ваша задача показать наилучший результат.

Формат входных данных

Первая строка ввода содержит целые числа N ($1 \leq N \leq 2000$) - количество комнат и M ($1 \leq M \leq 10000$) - количество дверей. В каждой из следующих M строк содержится описание двери - номера комнат, из которой она ведет и в которую она ведет (через дверь можно ходить только в одном направлении), а также целое число, которое прибавляется к количеству знаний при прохождении через дверь (это число по модулю не превышает 10000). Двери могут вести из комнаты в нее саму, между двумя комнатами может быть более одной двери.

Формат выходных данных

Выведите ":" - если можно получить неограниченно большой запас знаний, ":((" - если лабиринт пройти нельзя, и максимальное количество набранных знаний в противном случае.

Пример

Пример ввода	Пример вывода
2 2 1 2 5 1 2 -5	5

Задача В: Побег с космической станции

Представьте, что вы состоите на службе во внешней разведке Межгалактического Альянса Республиканских Сил (МАРС). Одному из агентов разведки крупно не повезло, и он был захвачен на засекреченной космической базе. К счастью, внешней разведке МАРС удалось заполучить план этой базы. И вот теперь вам поручено разработать план побега.

База представляет собой прямоугольник размером $N \times M$, со всех сторон окружённый стенами, и состоящий из квадратных отсеков единичной площади. База снабжена K выходами, до одного из которых агенту необходимо добраться. В некоторых отсеках базы находятся стены. Ваш агент может перемещаться из отсека в любой из четырех соседних с ним, если в том отсеке, куда он хочет переместиться, нет стены.

Кроме того, база снабжена системой гипертуннелей, способных перемещать агента из одного отсека базы (вход в гипертуннель) в другой (выход из гипертуннеля). Когда агент находится в отсеке, где есть вход в гипертуннель, он может (но не обязан) им воспользоваться.

Начальное положение вашего агента известно. Вам необходимо найти кратчайший путь побега (то есть путь, проходящий через минимальное количество отсеков).

Входные данные

В первой строке входного файла INPUT.TXT записаны числа N и M ($2 \leq N \leq 100$, $2 \leq M \leq 100$), задающие размеры базы: N — количество строк в плане базы, M — количество столбцов. Во второй строке записаны начальные координаты агента X_A, Y_A ($1 \leq X_A \leq N$, $1 \leq Y_A \leq M$). Первая координата задает номер строки, вторая — номер столбца. Строки нумеруются сверху вниз, столбцы слева направо. Далее следуют N строк по M чисел, задающих описание стен внутри базы: 1 соответствует стенке, 0 — её отсутствию. Далее в отдельной строке записано число H ($0 \leq H \leq 1000$) — количество гипертуннелей. В последующих H строках идут описания гипертуннелей. Каждый гипертуннель задается 4 числами: X_1, Y_1, X_2, Y_2 ($1 \leq X_1, X_2 \leq N$; $1 \leq Y_1, Y_2 \leq M$) — координатами входа и выхода гипертуннеля. Никакие два гипертуннеля не имеют общего входа. После этого в отдельной строке следует число K ($1 \leq K \leq 10$) — количество выходов с базы. В последующих K строках идут описания выходов с базы. Каждый выход задается двумя координатами X и Y ($1 \leq X \leq N$; $1 \leq Y \leq M$).

Гарантируется, что начальные координаты агента не совпадают ни с одним из выходов и он не стоит в отсеке, занятом стеной. Никакие входы и выходы гипертуннелей, а также выходы с базы не находятся в отсеках, занятых стенами. Никакой вход в гипертуннель не совпадает с выходом с базы.

Выходные данные

Если побег невозможен, выведите в выходной файл OUTPUT.TXT "Impossible". В противном случае в первой строке выведите число L - количество отсеков в кратчайшем пути побега. В последующих L строках последовательно выведите координаты отсеков кратчайшего пути побега. Если решений несколько, то выведите любое из них.

Пример

Пример ввода

```
4 5
2 1
0 0 0 0 0
0 1 0 0 0
0 0 0 0 0
0 0 0 0 0
2
1 2 1 4
3 1 1 4
1
2 4
```

Пример вывода

```
4
2 1
3 1
1 4
2 4
```

Задача С: Заправки

В стране N городов, некоторые из которых соединены между собой дорогами. Для того, чтобы проехать по одной дороге требуется один бак бензина. В каждом городе бак бензина имеет разную стоимость. Вам требуется добраться из первого города в N -ый, потратив как можно меньшее количество денег.

Формат входных данных

В вводе записано сначала число N ($1 \leq N \leq 100$), затем идет N чисел, i -ое из которых задает стоимость бензина в i -ом городе (все числа целые из диапазона от 0 до 100). Затем идет число M - количество дорог в стране, далее идет описание самих дорог. Каждая дорога задается двумя числами - номерами городов, которые она соединяет. Все дороги двухсторонние (то есть по ним можно ездить как в одну, так и в другую сторону); между двумя городами всегда существует не более одной дороги; не существует дорог, ведущих из города в себя.

Формат выходных данных

Выведите одно число - суммарную стоимость маршрута или -1, если добраться невозможно.

Пример

Пример ввода	Пример вывода
4 1 10 2 15 4 1 2 1 3 4 2 4 3	3
4 1 10 2 15 0	-1

Задача D: Игра в города

Всем известны правила игры «в города»: первый игрок называет произвольный город, следующий – город, название которого начинается на ту же букву, на которую заканчивается название предыдущего города, и т.д. Аналогичным образом можно играть не в названия городов, а, например, в названия животных. Задан список допустимых для описанной игры слов, слова в нем могут повторяться. Напишите программу, определяющую, в каком порядке в процессе игры должны быть названы слова из списка, чтобы каждое слово было использовано ровно столько раз, сколько оно в нем встречается.

Входные данные

В первой строке входа записано целое число N – количество слов в списке ($1 \leq N \leq 1000$), а в последующих N строках – сами слова. Каждое из них является последовательностью не более чем из 10 строчных английских букв.

Выходные данные

Выведите «YES» если такой порядок существует, либо «NO», если такого порядка не существует.

Пример ввода	Пример вывода	Коментарии
4 b ab bc bb	YES	ab bb b bc

Задача E: Флойд - 1

Полный ориентированный взвешенный граф задан матрицей смежности. Постройте матрицу кратчайших путей между его вершинами. Гарантируется, что в графе нет циклов отрицательного веса.

Формат входных данных

В первой строке входа записано единственное число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках по N чисел - матрица смежности графа (j -ое число в i -ой строке соответствует весу ребра из вершины i в вершину j). Все числа по модулю не превышают 100. На главной диагонали матрицы - всегда нули.

Формат выходных данных

Выведите N строк по N чисел - матрицу кратчайших расстояний между парами вершин. j -ое число в i -ой строке должно быть равно весу кратчайшего пути из вершины i в вершину j .

Пример

Пример ввода	Пример вывода
4 0 5 9 100 100 0 2 8 100 100 0 7 4 100 100 0	0 5 7 13 12 0 2 8 11 16 0 7 4 9 11 0

Задача F: Существование

Дан ориентированный взвешенный граф. По его матрице смежности нужно для каждой пары вершин определить, существует кратчайший путь между ними или нет.

Комментарий:

Кратчайший путь может не существовать по двум причинам:

- Нет ни одного пути
- Есть путь сколь угодно маленького веса

Формат входных данных

В первой строке входа записано единственное число N ($1 \leq N \leq 100$) - количество вершин графа. В следующих N строках по N чисел - матрица смежности графа (j -ое число в i -ой строке соответствует весу ребра из вершины i в вершину j), в которой число 0 обозначает отсутствие ребра, а любое другое число - наличие ребра соответствующего веса. Все числа по модулю не превышают 100.

Формат выходных данных

Выведите N строк по N чисел: j -ое число в i -ой строке должно быть равно 0, если путь из i в j не существует, 1 - если существует кратчайший путь, и 2 - если существует путь сколь угодно маленького веса.

Пример

Пример ввода	Пример вывода
5 0 1 2 0 0 1 0 3 0 0 2 3 0 0 0 0 0 0 0 -1 0 0 0 -1 0	1 1 1 0 0 1 1 1 0 0 1 1 1 0 0 0 0 0 2 2 0 0 0 2 2

Задача G: Минимальный каркас

От вас требуется определить вес минимального остовного дерева для неориентированного взвешенного связного графа.

Формат входных данных

В первой строке входа находятся числа N и M ($1 \leq N \leq 100$; $1 \leq M \leq 6000$), где N - количество вершин в графе, а M - количество рёбер. В каждой из последующих M строк записано по тройке чисел A , B , C , где A и B - номера вершин, соединённых ребром, а C - вес ребра (натуральное число, не превышающее 30000)

Формат выходных данных

Вывести одно число - искомый вес.

Пример

Пример ввода	Пример вывода
3 3 1 2 1 2 3 2 3 1 3	3

Задача Н: Дерево?

Неориентированный граф без петель и кратных ребер задан матрицей смежности. Определить, является ли этот граф деревом.

Формат входных данных

В вводе записано сначала число N - количество вершин графа (от 1 до 100). Далее записана матрица смежности размером $N \times N$, в которой 1 обозначает наличие ребра, 0 - его отсутствие. Матрица симметрична относительно главной диагонали.

Формат выходных данных

Выведите сообщение YES, если граф является деревом, и NO в противном случае

Пример

Пример ввода	Пример вывода
3 0 1 0 1 0 1 0 1 0	YES

Задача I: Каркас-разминка 1

Формат входных данных

В вводе записано сначала число N ($1 \leq N \leq 100$), а затем N чисел от 1 до 100 - элементы массива $A[i]$. Далее записаны два числа q и w (от 1 до N , не обязательно различные).

Требуется все элементы, которые равны $A[q]$, сделать равными $A[w]$. Постарайтесь сначала считать данные, потом сделать то, что требуется, и только потом вывести результат (а не делать преобразование на этапе вывода). Постарайтесь не пользоваться дополнительными массивами.

Формат выходных данных

Выведите N чисел - элементы массива $A[i]$ после преобразования.

Пример

Пример ввода	Пример вывода
5 1 4 2 2 5 3 2	1 4 4 4 5

Задача J: Цветной дождь

В Банановой республике очень много холмов, соединенных мостами. На химическом заводе произошла авария, в результате чего испарилось экспериментальное удобрение "зован". На следующий день выпал цветной дождь, причем он прошел только над холмами. В некоторых местах падали красные капли, в некоторых - синие, а в остальных - зеленые, в результате чего холмы стали соответствующего цвета. Президенту Банановой республики это понравилось, но ему захотелось покрасить мосты между вершинами холмов так, чтобы мосты были покрашены в цвет холмов, которые они соединяют. К сожалению, если холмы разного цвета, то покрасить мост таким образом не удастся. Посчитайте количество таких "плохих" мостов.

Формат входных данных

В первой строке записано N ($0 < N \leq 100$) - число холмов. Далее идет матрица смежности, описывающая наличие мостов между холмами (1 - мост есть, 0 - нет). В последней строке записано N чисел, обозначающих цвет холмов: 1 - красный; 2 - синий; 3 - зеленый.

Формат выходных данных

Вывести количество "плохих" мостов.

Пример

Пример ввода	Пример вывода
7 0 1 0 0 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 1 1 1 1 1 3 3	4

Задача К: Получи дерево

Дан связный неориентированный граф без петель и кратных ребер. Разрешается удалять из него ребра. Требуется получить дерево.

Формат входных данных

В вводе заданы два числа - N (от 1 до 100) и M - количество вершин и ребер графа соответственно. Далее идет M пар чисел, задающих ребра. Гарантируется, что граф связный.

Формат выходных данных

Выведите $N-1$ пару чисел - ребра, которые войдут в дерево. Ребра можно выводить в любом порядке.

Пример

Пример ввода	Пример вывода
4 4 1 2 2 3 3 4 4 1	1 2 2 3 3 4