

Задача А. Высокий маразм

Имя входного файла: `stdin`
 Имя выходного файла: `stdout`
 Ограничение по времени: 5 секунд
 Ограничение по памяти: 64 мегабайта

Город Нью-Васюки представляет собой таблицу N на N кварталов. Таким образом, в нем существует $(N + 1)(N + 1)$ перекрестков и $2N(N + 1)$ двусторонних дорог. Каждый перекресток имеет высоту. Известно, что левый верхний перекресток имеет высоту 0, а правый нижний – высоту 1. Для каждой дороги известно, сколько человек идет в каждом направлении по этой дороге. При этом, если дорога ведет от перекрестка i к перекрестку j , и разность высот $h = h_i - h_j$, то неудобство перемещения для каждого человека равно $\max(h, 0)$.

Для всех клеток, кроме двух угловых вы вправе выбирать любую высоту. Найдите распределение, при котором суммарное неудобство будет минимальным.

Формат входного файла

В первой строке входного файла содержится число N – размер таблицы, $1 \leq N \leq 500$. Далее следует $4N(N+1)$ чисел – искомое число людей для каждой дороги в естественном порядке. Сначала следует $N(N + 1)$ чисел для каждой дороги с запада на восток, затем столько же чисел с севера на юг, потом столько же чисел с востока на запад, а затем столько же чисел с юга на север. Для каждого направления дороги перечислены с севера на юг, а затем – с запада на восток. Обратите внимание на пример (там все числа специально различны).

Все количества являются целыми неотрицательными числами не превосходящими миллиона.

Формат выходного файла

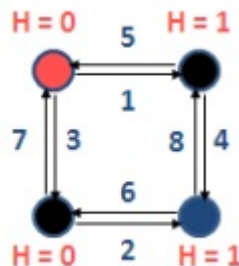
Выведите одно число – итоговое минимальное суммарное неудобство.

Примеры

	stdin	stdout
1		3
1		
2		
3		
4		
5		
6		
7		
8		

Note

Высоты могут быть произвольными вещественными числами! Картинка к примеру из условия:



Задача В. Пешки

Имя входного файла:	<code>pawns.in</code>
Имя выходного файла:	<code>pawns.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В первом классе Глеб увлекался шахматами. К тому моменту он знал только лишь как ходит пешка: она может бить по диагонали влево-вверх и вправо-вверх, и ходить на клетку вверх только если та клетка не занята другой фигурой. О том, что пешка может превращаться в ферзя Глеб не подозревает. Поэтому он придумал свой вариант шахмат.

Игра идёт на доске с N строками и M столбцами ($1 \leq N \leq 100$, $1 \leq M \leq 100$) по следующим правилам. В нижней строке, имеющей номер 1, стоят P белых пешек, белых фигур на доске больше нет. На остальной части доски стоят разные чёрные фигуры (их названия Глеб не знает). Ходят только белые, их цель — побить все чёрные фигуры.

Как и в настоящих шахматах, если пешка Глеба бьёт чёрную фигуру, то она становится на её место, а побитая фигура убирается с доски. Считается, что Глеб выиграл, если он сумел побить белыми пешками все чёрные фигуры, в противном случае он проиграл. Помогите ему по заданной конфигурации всех фигур определить, сможет ли он выиграть, и, в случае успеха, выведите правильную последовательность ходов белых пешек.

Формат входного файла

Сначала вводятся четыре целых числа N , M , P , K ($1 \leq N \leq 100$, $1 \leq M \leq 100$, $0 \leq P \leq M$, $1 \leq K \leq 1000$, $K \leq (M - 1)N$). Далее записано P различных чисел — номера столбцов p_j ($1 \leq p_j \leq M$), в которых стоят белые пешки. Далее идут K различных пар целых чисел — координаты (строки и столбцы) чёрных фигур r_i , c_i ($2 \leq r_i \leq N$, $1 \leq c_i \leq M$).

Формат выходного файла

Если пешки не смогут съесть все фигуры, выведите единственное слово NO.

В противном случае в первую строку выведите YES, вторая строка должна содержать суммарное число перемещений C , последующие C строк — описание ходов пешек, по одному ходу на каждую строку. Каждый ход задаётся двумя координатами r , c пешки (номерами строки и столбца), которая будет ходить, и символом m , принимающем три значения: L, R, F — побить вперед и влево, побить вперед и вправо, сделать шаг вперед соответственно. Данные о ходе следует выводить разделёнными одним пробелом, сначала координаты, потом тип хода.

Если последовательностей ходов несколько, выведите любой из них. Обратите внимание, что минимизировать количество перемещений не требуется.

Примеры

<code>pawns.in</code>	<code>pawns.out</code>
2 2 2 1 1 2 2 2	YES 1 1 1 R
3 3 2 2 1 3 3 1 3 3	NO

Задача С. Улиточки

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

Две улиточки Маша и Петя сейчас находятся в на лужайке с абрикосами и хотят добраться до своего домика. Лужайки пронумерованы числами от 1 до n и соединены дорожками (может быть несколько дорожек соединяющих две лужайки, могут быть дорожки, соединяющие лужайку с собой же). В виду соображений гигиены, если по дорожке проползла улиточка, то вторая по той же дорожке уже ползти не может. Помогите Пете и Маше добраться до домика.

Формат входного файла

В первой строке файла записаны четыре целых числа — n , m , a и h (количество лужаек, количество дорог, номер лужайки с абрикосами и номер домика).

В следующих m строках записаны пары чисел. Пара чисел (x, y) означает, что есть дорожка с лужайки x до лужайки y (из-за особенностей улиток и местности дорожки односторонние).

Ограничения: $2 \leq n \leq 10^5$, $0 \leq m \leq 10^5$, $s \neq t$

Формат выходного файла

Если существует решение, то выведите YES и на двух отдельных строчках сначала путь для Машеньки (т.к. дам нужно пропускать вперед), затем путь для Пети. Если решения не существует, выведите NO. Если решений несколько, выведите любое.

Примеры

<code>snails.in</code>	<code>snails.out</code>
3 3 1 3	YES
1 2	1 3
1 3	1 2 3
2 3	

Задача D. Химия!!!

Имя входного файла: `molecule.in`
 Имя выходного файла: `molecule.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Вася и Сережа играют в следующую игру. В некоторых клетках клетчатого листка Сережа рисует один из символов 'H', 'O', 'N' или 'C', после чего Вася должен провести между некоторыми находящимися в соседних клетках символами линии так, чтобы получилось корректное изображение химической молекулы. К сожалению, Сережа любит рисовать много символов, и Вася не может сразу определить, возможно ли вообще нарисовать линии нужным способом. Помогите ему написать программу, которая даст ответ на этот вопрос.

В этой задаче проведенные между символами химических элементов линии будем считать корректным изображением молекулы, если они удовлетворяют следующим условиям:

- каждая линия соединяет символы, нарисованные в соседних (по стороне) клетках,
- между каждой парой символов проведено не более одной линии,
- от каждого элемента отходит ровно столько линий, какова валентность этого элемента (1 для H, 2 для O, 3 для N, 4 для C),
- пустые клетки ни с чем не соединены, и
- хотя бы в одной клетке нарисован какой-то символ.

Формат входного файла

Первая строка входного файла содержит два натуральных числа n и m ($1 \leq n, m \leq 50$) — размеры листочка, на котором рисует Сережа. Далее следуют n строк по m символов в каждой, задающих конфигурацию химических элементов, которую нарисовал Сережа; пустые клетки задаются символом '.'.

Формат выходного файла

В выходной файл выведите одно слово: 'Valid', если линии провести требуемым образом можно, и 'Invalid', если нельзя.

Примеры

<code>molecule.in</code>	<code>molecule.out</code>
<pre>3 4 HOH. NCOH OO..</pre>	Valid
<pre>3 4 HOH. NCOH OONH</pre>	Invalid