

## Задача А. Максимальный поток

Имя входного файла: flow2.in  
Имя выходного файла: flow2.out  
Ограничение по времени: 0.5 секунда  
Ограничение по памяти: 256 мегабайт

Задан ориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Найдите максимальный поток из вершины с номером 1 в вершину с номером  $n$ .

### Формат входных данных

Первая строка входного файла содержит  $n$  и  $m$  — число вершин и ребер в графе ( $2 \leq n \leq 500$ ,  $1 \leq m \leq 10\,000$ ). Последующие строки описывают ребра. Каждое ребро задается тремя числами: начальная вершина ребра, конечная вершина ребра и пропускная способность ребра. Пропускные способности не превосходят  $10^9$ .

### Формат выходных данных

Выведите величину максимального потока между вершинами 1 и  $n$ .

### Примеры

flow2.in	flow2.out
4 5 1 2 1 1 3 2 3 2 1 2 4 2 3 4 1	3

## Задача В. 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  $h = h_j - h_i$ , 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 входных данных

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 выходных данных

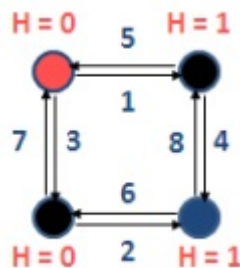
Выведите одно число – итоговое минимальное суммарное неудобство.

### Примеры

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

### Замечание

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



## Задача С. Живопись

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

В стране Олимпия очень развита живопись. Картиной считается любой прямоугольник, который состоит из черных и белых единичных квадратов. Художник Олимпус решил радикально улучшить свои картины. Для этого он планирует к белому и черному цветам добавить еще и серый оттенок. По его задумке, граница между каждым черным и белым квадратом должна содержать серую линию, чтобы образовался эффект плавного перехода.

Однако, перед началом работы, он обнаружил, что серая краска очень дорого стоит. Чтобы сэкономить деньги, художник решил оценить, не выгоднее ли сначала перекрасить некоторые белые квадраты в черные, а черные в белые для того, чтобы минимизировать расходы на краску.

Напишите программу, которая по информации о существующей картине определяет минимальную сумму денег, которая понадобится на улучшение картины.

### Формат входных данных

Первая строка входного файла содержит пять натуральных чисел  $N, M, w, b, g$ .  $1 \leq N, M \leq 70$  — высота и ширина картины,  $1 \leq w, b, g \leq 1000$  — цена рисования одного белого единичного квадрата, черного единичного квадрата и серой линии единичной длины, соответственно. Далее следует  $N$  строк, каждая из которых состоит из  $M$  литер. Литера `B` соответствует черному квадрату, а `W` — белому.

### Формат выходных данных

Единственная строка выходного файла должна содержать одно целое число, являющееся минимальной суммой затрат на улучшение картины.

### Примеры

<code>drawing.in</code>	<code>drawing.out</code>
3 2 10 12 1 BW WB BW	7