**Submit a solution for A**

|  |  |
| --- | --- |
| **Full score:** | 100 |
| **Input file name:** | a.in |
| **Output file name:** | a.out |
| **Time limit:** | 100 ms |
| **Real time limit:** | 5 s |
| **Memory limit:** | 256M |
| **Stack limit:** | 64M |

**Зимовий мотив - Snow**

Хтось із вас бачив сніжинку, яка не має осей симетрії? Я до сьогодні твердо переконаний, що таких сніжинок просто не існує! У когось є інша думка? Що ж, щоб довести це, достатньо принести і показати мені таку сніжинку :) Тепер по суті. У нас є закодована інформація про … Ми не знаємо про що, але підозрюємо, що перед Новим роком нам могли б прислати зображення ексклюзивної сніжинки. Можна вважати, що сніжинка має 4 осі симетрії: вертикальну, горизонтальну та дві діагональних. Отримана інформація - це квадратна таблиця заповнена цілими числами. Однаковим узорам сніжинки мають відповідати однакові числа. Вам треба написати програму, яка визначить чи може бути в таблиці приховане зображення сніжинки.

Формат вхідних даних: У першому рядку вхідного файлу знаходиться одне ціле число N (0<N<=100) . У наступних N рядках знаходиться по N цілих чисел не більших 100. Це і є закодована інформація.

Формат вихідних даних: У вихідний файл вивести «YES» у випадку, коли розшифрована інформація є сніжинкою. Якщо ж дана таблиця не є прихованою сніжинкою, то у першому рядку вивести « NO », а в другому – одне число від 0 до 3 – кількість наявних осей симетрії.

*Пояснення. У першому прикладі є одна вісь симетрії - горизонтальна, яка проходить через другий рядок. У випадку непарної кількості рядків (стовбців) середній рядок (стовбець) симетричний сам собі.*

**Examples**

|  |  |
| --- | --- |
| **Input in a.in** | **Output in a.out** |
| 3  1 2 3  1 2 3  1 2 3 | NO  1 |
| 3  1 2 1  2 1 2  1 2 1 | YES |

**Submit a solution for B**

|  |  |
| --- | --- |
| **Full score:** | 100 |
| **Input file name:** | b.in |
| **Output file name:** | b.out |
| **Time limit:** | 1000 ms |
| **Real time limit:** | 5 s |
| **Memory limit:** | 256M |
| **Stack limit:** | 64M |

**Зимовий мотив - NewYear**

Діди Морози хочуть зустрітися для святкування Нового року. Виявили бажання бути разом у новорічну ніч K (1<=K<=100) Дідів Морозів. Кожен з них в цю ніч може бути в одному із N (1<=N<=1000) населених пунктів. Між цими пунктами є M (1<=M<=10000) доріг. Оскільки життя Дідів Морозів пов’язане з плином часу, а час незворотній, то всі дороги для них мають лише один напрям. Це накладає певні обмеження на вибір місця для святкового зібрання. Допоможіть Дідам Морозам визначити кількість населених пунктів, що можуть взяти участь у конкурсі на місце проведення святкування, тобто таких, що є досяжними для них усіх.

Формат вхідних даних: У першому рядку вхідного файлу є три цілих числа K , N та M , розділених пропусками. Далі у K рядках по одному числу – номеру населеного пункту, де перебувають Діди Морози у новорічну ніч. Наступні M рядків містять по два числа i , j (1<=i , j<=N), що вказують дороги від пункту і до j .

Формат вихідних даних: У вихідний файл вивести одне число – кількість пунктів, що можуть стати місцем для святкування.

*Перелік населених пунктів, які є досяжними для всіх Дідів Морозів: 1, 2, 3.*

**Examples**

|  |  |
| --- | --- |
| **Input in b.in** | **Output in b.out** |
| 2 4 4  2  4  1 3  2 1  4 2  4 3 | 3 |

**Submit a solution for C**

|  |  |
| --- | --- |
| **Full score:** | 100 |
| **Input file name:** | c.in |
| **Output file name:** | c.out |
| **Time limit:** | 3000 ms |
| **Real time limit:** | 5 s |
| **Memory limit:** | 256M |
| **Stack limit:** | 64M |

**Зимовий мотив - Business**

Василько не мріє бути програмістом – він хоче стати мільйонером. І не десь там у далекому майбутньому, а вже цього року. Для втілення своєї мрії Василько організував фірму, яка під Новий рік буде займатися поставкою новорічних ялинок. Оскільки мільйон треба заробити уже цього року, Василько вирішив обійтися без посередників. Бізнес-план вийшов дивовижно простий: сам рубаю, сам доставляю і сам продаю. Василько сміливо взявся за справу, але коли отримав угоду на вирубку ялинок, то задумався. За один раз йому дозволили вивозити рівно K (0<= K <=10000) ялинок і партія Зелених домоглася певних обмежень на вирубку: ялинки можна рубати тільки рухаючись на схід або на південь; починати вирубку слід в одному кутку і закінчувати у будь-якому іншому. Ділянки мають форму прямокутника із сторонами N , M (1<= N , M <=100) , сторони якого були орієнтовані чітко на захід, схід, північ та південь. Василько зразу ж помітив, що забезпечити собі гарний результат вирубки можна лише тоді, коли починаєш рубати із північно-західного кутка і закінчуєш у південно-східному :) «Зелені» добре подумали над тим, щоб виконати угоду по вирубці йому було складно. Василько це побачив і пошкодував, що не став спочатку програмістом. Допоможіть Василькові правильно провести вирубку і стати мільйонером, або повідомте, що на цій ділянці краще не рубати.

Формат вхідних даних: У першому рядку вхідного файлу дано три цілих числа N , M , K , розділених пропусками. Дальше у N рядках міститься по M символів: «.» - це вільне місце, а « # » - позначає місце знаходження ялинки.

Формат вихідних даних: У вихідний файл у випадку наявності правильного варіанту вирубки вивести послідовність координат, які виводити по одній у кожному рядку. Перша координата завжди (1,1), остання – ( N , M ). В іншому випадку вивести повідомлення « To not cut !»

**Examples**

|  |  |
| --- | --- |
| **Input in c.in** | **Output in c.out** |
| 4 4 3  ....  #...  ###.  #### | 1 1  2 1  2 2  2 3  3 3  3 4  4 4 |
| 3 4 1  #...  ..#.  #.## | To not cut! |