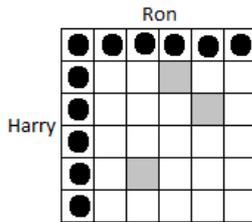


## Dementores a la puerta

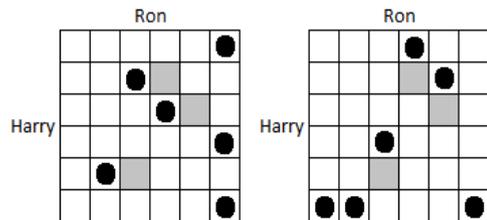
¡Los temidos dementores están a las puertas de Hogwarts! Como de costumbre, será tarea de los amigos Harry Potter y Ron Weasley alejarlos de ahí. Para alejarlos ambos saben invocar el hechizo **patronus**. Cuando un mago invoca un patronus los dementores se alejan en dirección contraria al mago lo más posible. Por ejemplo, si el mago está del lado oeste, los dementores se alejarán hacia el este hasta chocar con algún obstáculo o pared. Si está del lado norte, los dementores se alejarán hacia el sur.

El patio de Hogwarts se representa como un cuadrado de lado **N**, Harry y Ron están en el lado oeste y norte del patio. El patio tiene **O** obstáculos internos. Inicialmente toda la fila norte (la fila superior) y la columna oeste (la columna hasta la izquierda) están llenas de dementores.



La primer figura muestra el patio de Hogwarts, Harry está del lado oeste y Ron del lado norte. Los puntos negros son las casillas en donde hay dementores (en una casilla pueden haber múltiples dementores). Los cuadros grises son obstáculos y los dementores no pueden pasar a través de ellos, sin embargo, los *patronus* si atraviesan los obstáculos.

Aquí se muestra cómo quedan los dementores si Harry o Ron invocan su patronus. La izquierda muestra el resultado de Harry y derecha el resultado de Ron.



El objetivo de Harry y de Ron es llevar a todos los dementores a la esquina inferior derecha del patio.

Como no son buenos para calcular, te pidieron que les digas cuál es el número mínimo de patronus que deben invocar entre los dos (cada uno de ellos puede invocar varios) para lograrlo.

### Problema

Escribe un programa que dados los obstáculos del patio determine el número mínimo de patronus que se requieren para llevar a los dementores a la esquina inferior derecha.

### Restricciones

$1 < N \leq 1,000$

Dimensiones del patio

$0 \leq O \leq 1,000,000$

Número de obstáculos en el patio

### Entrada

Tu programa debe leer del teclado los siguientes datos:

- En la primera línea los números **N** y **O**, dimensiones del patio y número de obstáculos.
- En las siguientes **O** líneas habrá **2** números enteros separados por un espacio en cada una que indican la columna y la fila de cada uno de los obstáculos respectivamente. La esquina superior izquierda se representa como la posición (0, 0).

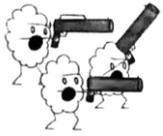
### Salida

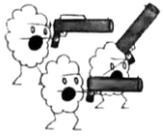
Tu programa debe escribir a la pantalla un único número entero que indique la cantidad mínima de patronus necesarios para llevar a todos los dementores a la esquina inferior derecha.

Te aseguramos que para todos los casos de prueba es posible lograr el objetivo.

Entrada	Salida	Explicación
6 3 3 1 4 2 2 4	3	El ejemplo representa el patio de la figura. Una manera de resolverlo es que los patronus sean invocados en el orden Ron-Harry-Ron llevando a todos los dementores a la esquina deseada en 3 invocaciones.







## Capturando al gato

Homero Simpson y su equipo de *zorros* (agentes secretos) se han dado a la tarea de capturar al criminal internacional conocido como *el gato*. El plan de Homero es crear un cerco de vigilancia alrededor del Museo de Springfield, próximo objetivo del criminal.

Si se traza un círculo alrededor del museo existen  $N$  lugares adecuados para que un *zorro* ponga su puesto de vigilancia. Homero quiere poner el máximo número de *zorros* posibles alrededor del Museo, sin embargo hay una limitante. El método de comunicación entre los *zorros* (inventado por el mismo Homero) es el *teléfono de hilo*. Este teléfono consta de dos vasos unidos por un hilo el cual debe estar tenso para que funcione. El *círculo* de *zorros* estará comunicado por teléfonos de la siguiente forma: cada *zorro* tendrá dos teléfonos, uno para comunicarse con el *zorro* a su derecha y otro para comunicarse con el *zorro* a su izquierda. Para que la comunicación sea funcional, el *círculo* deberá estar cerrado.

Homero no es un buen ingeniero y aunque puede hacer muchos teléfonos, requiere que el hilo de todos tenga la misma longitud.

Ayuda a Homero a determinar cuál es el máximo número de *zorros* que puede colocar alrededor del Museo de modo que todos estén comunicados por teléfonos cuyo hilo sea de la misma longitud.

### Problema

Escribe un programa que conociendo el número de posiciones adecuadas en el círculo y las distancias entre cada una de estas posiciones, determine el máximo número de *zorros* que se pueden colocar de modo que la longitud de los hilos de sus teléfonos sea la misma.

### Restricciones

$1 < N \leq 5,000$

Cantidad de posiciones para puestos de vigilancia.

$1 < p \leq 2,000,000,000$

Perímetro total del círculo de vigilancia.

### Entrada

Tu programa debe leer del teclado la siguiente información

- En la primera línea el número  $N$ , la cantidad de posiciones posibles para *zorros*
- En las siguientes  $N$  líneas un entero  $d_i$  que representa la distancia que hay entre el puesto  $i$  y el puesto  $i+1$ . El entero en la última de estas líneas representa la distancia del último puesto hacia el primero para cerrar el círculo.

### Salida

Tu programa debe escribir a la pantalla un único número entero que representa la cantidad máxima de *zorros* que se pueden colocar.

Entrada	Salida	Imagen	Explicación
6 3 2 1 4 5 5	4		Hay 6 posiciones posibles las cuales se muestran con un punto negro. Entre puntos se muestran las distancias de un punto al siguiente avanzando en sentido horario.  El máximo número de <i>zorros</i> que se pueden poner en el círculo es 4. Utilizando las posiciones 1, 3, 5 y 6. La distancia entre todos ellos es de 5.



### Evaluación

Para este problema, todos los casos cuya  $N$  sea mayor a 1,000 estarán agrupados, es decir, para obtener los puntos del grupo es necesario que tu programa resuelva correctamente todos los casos. El valor total de este grupo será **25 puntos**.



## La guardia negra

Según *la canción del fuego y el hielo*, la guardia de los centinelas negros ha defendido al reino de Westeros de las amenazas del norte durante los últimos 8,000 años. A últimas fechas un gran número de huestes enemigas se ha venido concentrando en la frontera norte de Westeros. El comandante de la guardia, Lord Snow, ha mandado llamar por ti para que le ayudes a planear la defensa.

La guardia tiene representado el campamento enemigo como una cuadrícula de  $M$  filas por  $N$  columnas, en cada casilla de la cuadrícula hay un número  $s_{i,j}$  que representa la cantidad de soldados enemigos en esa casilla.

La guardia cuenta con una catapulta capaz de lanzar un proyectil explosivo a cualquier lugar de esta cuadrícula. Al explotar, el proyectil eliminará a todos los soldados enemigos que se encuentren a una *distancia* menor o igual a un número  $d_i$  que depende del proyectil en cuestión. La *distancia* entre dos casillas se define como la suma del valor absoluto de la diferencia de sus columnas mas el valor absoluto de la diferencia de sus filas.

El comandante de la guardia quiere que desarrolles un programa para calcular la cantidad de soldados que eliminaría un proyectil específico si fuera lanzado en una cierta casilla.

### Problema

Escribe un programa que dada la cuadrícula del campamento, el número de soldados en cada casilla, el número de proyectiles, la distancia de alcance de cada uno y el lugar donde quieres lanzarlo, calcule el número de enemigos que serán eliminados con cada proyectil.

### Restricciones

$1 \leq M, N \leq 1,000$	Filas y de columnas del campamento respectivamente.
$1 \leq P \leq 100,000$	Número de proyectiles con los que cuenta la guardia.
$0 \leq s_{i,j} \leq 1,000,000,000$	Número de soldados en la casilla $(i,j)$ .
$0 \leq d_i \leq 500$	Alcance del proyectil $i$ .

### Entrada

Tu programa debe leer del teclado la siguiente información:

- En la primer línea los números  $M$  y  $N$  que representan el número de filas y columnas.
- En cada una de las siguientes  $M$  líneas hay  $N$  números enteros separados por un espacio que representan los soldados en cada casilla del campamento.
- En la siguiente línea el número  $P$  de proyectiles que tiene la guardia.
- En las últimas  $P$  líneas hay  $3$  enteros separados por espacios en cada una que representan la fila y columna donde se lanzará un proyectil y la distancia de alcance del mismo.

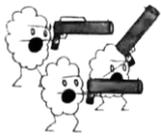
Tanto las filas como las columnas inician numeradas a partir del  $0$  y hasta  $M-1$  y  $N-1$  respectivamente. La fila  $0$  es la fila superior y la columna  $0$  es la columna hasta la izquierda.

Puedes estar seguro de que los proyectiles serán lanzados de tal forma que su alcance no exceda el campamento, es decir, ninguna explosión llegará a una casilla que este fuera de la cuadrícula.

### Salida

Tu programa deberá escribir a la pantalla  $P$  líneas con un número cada una. La  $i$ -ésima línea debe contener la cantidad de enemigos que serían eliminados al lanzar el proyectil  $i$ .





**Ejemplo**

Entrada	Salida	Imagen	Explicación																																																		
5 5 1 2 3 4 5 5 4 3 2 1 1 1 1 1 1 2 3 4 3 0 7 8 9 6 5 3 1 0 0 2 2 2 3 1 1	5 36 18	<table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>2</td><td>3</td><td>4</td><td>3</td><td>0</td></tr> <tr><td>7</td><td>8</td><td>9</td><td>6</td><td>5</td></tr> </table> <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>2</td><td>3</td><td>4</td><td>3</td><td>0</td></tr> <tr><td>7</td><td>8</td><td>9</td><td>6</td><td>5</td></tr> </table>	1	2	3	4	5	5	4	3	2	1	1	1	1	1	1	2	3	4	3	0	7	8	9	6	5	1	2	3	4	5	5	4	3	2	1	1	1	1	1	1	2	3	4	3	0	7	8	9	6	5	<p>La imagen superior muestra los primeros 2 proyectiles.</p> <p>El primer proyectil, lanzado en la fila 1, columna 0 y con alcance 0, sólo elimina a los soldados de esa casilla.</p> <p>El proyectil 2 se lanza en la casilla 2,2 y elimina todos los soldados a distancia menor o igual que 2.</p> <p>La imagen inferior muestra el tercer proyectil.</p> <p><b>NOTA:</b> Observa que los proyectiles no se lanzan realmente, de modo que al calcular la cantidad de enemigos que podrías eliminar, debes considerar todas las casillas sin importar si se intersectan con el alcance de algún proyectil previo.</p>
1	2	3	4	5																																																	
5	4	3	2	1																																																	
1	1	1	1	1																																																	
2	3	4	3	0																																																	
7	8	9	6	5																																																	
1	2	3	4	5																																																	
5	4	3	2	1																																																	
1	1	1	1	1																																																	
2	3	4	3	0																																																	
7	8	9	6	5																																																	

**Evaluación**

Para este problema los casos de prueba se agruparán en 3 grupos distintos. Para obtener los puntos de un grupo es necesario que tu programa resuelva correctamente **TODO**s los casos de ese grupo, de otra forma obtendrás 0 puntos por ese grupo aun cuando hayas resuelto correctamente algunos de los casos.

**RESTRICCIONES ESPECIALES PARA EL GRUPO 1 (Valor del grupo, 25 puntos)**

- $1 \leq M, N \leq 100$  Filas y de columnas del campamento respectivamente.
- $1 \leq P \leq 100$  Número de proyectiles con los que cuenta la guardia.
- $0 \leq s_{i,j} \leq 100,000$  Número de soldados en la casilla  $(i,j)$ .
- $0 \leq d_i \leq 50$  Alcance del proyectil  $i$ .

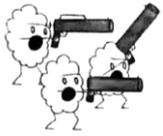
**RESTRICCIONES ESPECIALES PARA EL GRUPO 2 (Valor del grupo, 50 puntos)**

- $1 \leq M, N \leq 1,000$  Filas y de columnas del campamento respectivamente.
- $1 \leq P \leq 1,000$  Número de proyectiles con los que cuenta la guardia.
- $0 \leq s_{i,j} \leq 1,000,000$  Número de soldados en la casilla  $(i,j)$ .
- $0 \leq d_i \leq 500$  Alcance del proyectil  $i$ .

**RESTRICCIONES ESPECIALES PARA EL GRUPO 3 (Valor del grupo, 25 puntos)**

Las restricciones para el grupo 3 son las especificadas en la sección **Restricciones** del problema.

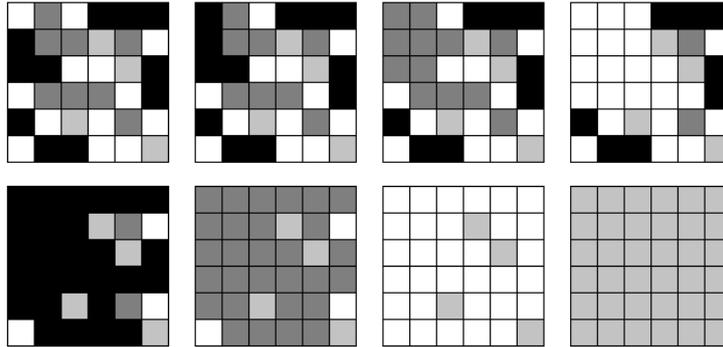




## Coloréalo

*Coloréalo* es un juego de tablero para una persona. El juego se desarrolla sobre una cuadrícula de  $M$  filas por  $N$  columnas. Cada casilla del tablero está pintada de uno de  $C$  colores distintos. Si dos casillas contiguas, ya sea de manera vertical u horizontal tienen el mismo color, se considera que ambas casillas están *unidas*. El objetivo del juego es colorear todo el tablero del mismo color.

En cada turno el jugador selecciona un color, cualquiera que él desee, y colorea de ese color la casilla superior izquierda y todas las que estén *unidas* a ella.



La figura muestra un juego completo de *Coloréalo*. Inicialmente el tablero tiene 4 colores. El jugador termina con la siguiente secuencia de siete turnos: 

### Problema

Este problema es distinto de los tres anteriores, en este problema debes descargar del sistema de evaluación los 10 casos de entrada del problema. Cada uno de estos casos contiene un tablero de coloréalo en el formato que se describe más abajo.

Para cada caso deberás generar un archivo, utilizando cualquier método que desees, que contenga la secuencia de colores que se deben usar en cada turno para colorear completamente el tablero de un color.

Mientras menos turnos uses para lograr el objetivo del juego, mejor será tu calificación.

### Evaluación

Para cada uno de los casos tu resultado será calificado con el siguiente criterio:

- Si tu secuencia de colores no colorea el tablero completamente de un color, obtendrá 0 puntos para ese caso.
- Si tu secuencia colorea correctamente el tablero pero usa un número de turnos mayor a **la peor** solución del comité científico, obtendrás 0 puntos para ese caso. Del sistema de evaluación podrás descargar un archivo con el número de turnos que usa la peor solución del comité científico para cada caso.
- Si tu secuencia no cae en los puntos anteriores, se comparará con las del resto de los concursantes que tengan una respuesta válida y contra las respuestas del comité procediendo como sigue:
  - Por tener una respuesta válida que no cae en los puntos anteriores ganas el 20% de los puntos del caso.
  - El 80% restante dependerá de cuántos turnos uses. La mejor solución (aquella que use menos turnos) obtendrá el 100% de los puntos del caso.

### Restricciones

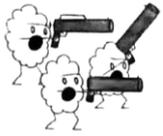
$1 \leq M, N \leq 100$

$1 \leq C \leq 10$

Número de filas y columnas del tablero

Número de colores distintos en el tablero





### Entrada

Los archivos de casos contienen los siguientes datos:

- En la primer línea los números **M**, **N** y **C** que indican respectivamente el número de filas y columnas del tablero y el número de colores distintos disponibles.
- En cada una de las siguientes **M** líneas hay **N** enteros entre **0** y **C-1** separados por un espacio que indican el color del que está coloreada esa casilla.

### Salida

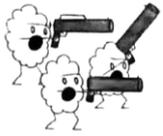
Para cada caso deberás entregar un archivo que contenga una línea con enteros entre **0** y **C-1** separados por un espacio que indique la secuencia de colores para colorear el tablero.

Si tu secuencia tiene más de **M\*N** números será rechazada por el evaluador.

### Ejemplo

Entrada	Salida	Explicación
6 6 4 0 1 0 2 2 2 2 1 1 3 1 0 2 2 0 0 3 2 0 1 1 1 0 2 2 0 3 0 1 0 0 2 2 0 0 3	2 1 0 2 1 0 3	El ejemplo representa el tablero mostrado en la figura.





<b>Nombre</b>	<b>Límite de memoria</b>	<b>Límite de tiempo</b>	<b>Número de casos</b>
Dementores	256MB	1 seg	20
Capturando al gato	256MB	1 seg	20
Guardia negra	256MB	1 seg	20
Coloréalo	No aplica	No aplica	10

