



Aliens

Uno de los satelites nuestros ha descubierto una civilizacion de aliens de un planeta lejano. De alguna manera ya se ha obtenido una foto de baja resolución de una area cuadrada del planeta. La foto obtenida muestra muchos signos de vida inteligente. Nuestros expertos han identificado n puntos de interes en la foto. Los puntos se encuentran enumerados de 0 a $n - 1$. Ahora se desea obtener fotos de mas alta resolución que contenga aquellos n puntos de interés.

Internamente, el satélite ha dividido el área de baja resolución de la foto en una matriz de m por m unidades de celdas. Ambas filas y columnas de la matriz estan consecutivamente enumeradas de 0 to $m - 1$ (desde arriba abajo y de izquierda a derecha respectivamente). Utilizaremos (s, t) para denotar las celdas en la fila s y la columna t . El numero de punto i se encuentra localizado en la celda (r_i, c_i) . Cada celda puede contener un numero arbitrario de puntos.

Nuestro satelite se encuentra sobre una orbita estable que pasa directamente por sobre la diagonal *principal* de la matriz. La diagonal principal se considera como una linea de segmento que conecta la esquina superior izquierda y la esquina inferior derecha de la matriz. El satélite puede tomar fotos de alta resolución de cualquier área que satisfaga las siguientes restricciones:

- la forma del área debe ser un cuadrado,
- una diagonal del cuadrado se encuentra totalmente contenida en la diagonal principal de la matriz,
- cada celda de la grilla esta completamente dentro o completamente fuera del area fotografiada.

Tome en cuenta que el satélite puede solamente tomar k fotos de alta resolución.

Una vez que el satélite ha terminado de tomar las fotos, las celdas capturadas serán transmitidas a nuestra base por medio de las fotos de alta resolución (independientemente de que si las celdas contienen algunos puntos de interés). Los datos para cada celda fotografiada serán transmitidos *una única vez*, pese a que haya sido fotografiada varias veces.

De esa manera, debemos escoger a lo más k areas cuadradas que serán fotografiadas, asumiendo que:

- cada celda que contiene al menos un punto de interes es fotografiada al menos una vez, y
- el número de celdas que son fotografiadas al menos una vez debe ser minimizado.

Tu tarea es encontrar el numero total de celdas fotografiadas más pequeña posible

Detalles de implementación

Debes implementar la siguiente función (method):

- `int64 take_photos(int n, int m, int k, int[] r, int[] c)`
 - `n`: El número de puntos de interés,
 - `m`: el número de filas(y también el número de columnas) en la matriz,
 - `k`: el número máximo de fotos que el satélite puede tomar,
 - `r` y `c`: dos vectores de longitud `n` que describen las coordenadas de las celdas de la matriz que contienen los puntos de interés. Para cada $0 \leq i \leq n - 1$, el punto de interés `i`-th se encuentra localizado en la celda `(r[i], c[i])`,
 - la función debe retornar la cantidad más pequeña posible del total de celdas que son fotografiadas al menos una vez(Teniendo en cuenta que la foto debe cubrir la mayor cantidad de puntos de interés).

Porfavor usa los ejemplos de archivos plantillas provistos para los detalles de implementación en tu lenguaje de programación

Ejemplos

Ejemplo 1

```
take_photos(5, 7, 2, [0, 4, 4, 4, 4], [3, 4, 6, 5, 6])
```

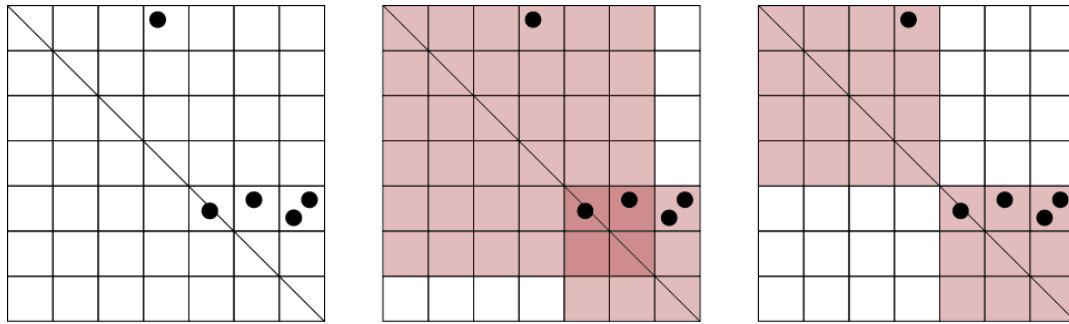
En este ejemplo tenemos una matriz de 7×7 con 5 puntos de interés. Los puntos de interés se encuentran localizados en cuatro diferentes celdas: `(0, 3)`, `(4, 4)`, `(4, 5)` y `(4, 6)`. Y recuerda que debes tomar al menos 2 fotos de alta resolución para este ejemplo.

Una manera de capturar todos los cinco puntos de interés, es realizar la toma de dos fotos: una con las celdas `(0, 0)` y `(5, 5)` en la esquina opuesta, y otra con las celdas `(4, 4)` y `(6, 6)` en la otra esquina opuesta. Si tomamos estas dos fotos, el satélite transmitiría los datos de 41 celdas. Esta selección nos es la más optima.

La solución más optima, utiliza una foto para capturar un cuadrado de 4×4 que contiene las celdas `(0, 0)` y `(3, 3)` y otra foto para capturar un cuadrado de 3×3 que contiene las celdas `(4, 4)` and `(6, 6)`. De esta manera las celdas fotografiadas serían solamente 25, lo cual es optimo, entonces la función `take_photos` debería retornar 25.

Nota que es suficiente fotografiar la celda `(4, 6)` una vez, pese a que esta contiene dos puntos de interés.

Este ejemplo se encuentra visualmente explicado en la figura de abajo. La figura que se encuentra más a la izquierda muestra la matriz que corresponde al ejemplo. La que esta al medio muestra la solución que no es tan optima en donde 41 celdas son fotografiadas. La figura que se encuentr más a la Derecha muestra la solución óptima.

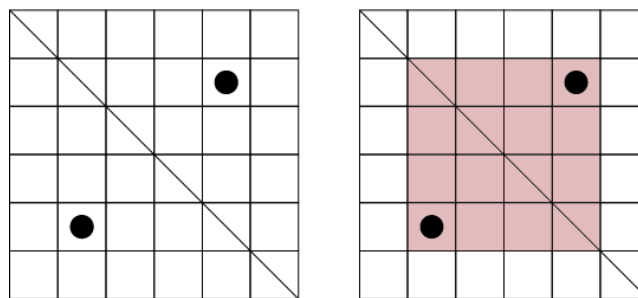


Ejemplo 2

`take_photos(2, 6, 2, [1, 4], [4, 1])`

Para este ejemplo se tiene 2 puntos de interes localizados simetricamente: en las celdas (1,4) y (4,1). Cualquier foto valida que contenga uno de ellos contiene a la otra tambien. Por lo tanto, es suficiente utilizar una simple foto.

La figura de abajo muestra de forma gráfica el ejemplo y su solución óptima. En esta solución el satélite captura una sola foto de 16 celdas.



Subtareas

Para todas las Subtareas, $1 \leq k \leq n$.

1. (4 puntos) $1 \leq n \leq 50$, $1 \leq m \leq 100$, $k = n$,
2. (12 puntos) $1 \leq n \leq 500$, $1 \leq m \leq 1000$, para cada i de tal forma que $0 \leq i \leq n - 1$, $r_i = c_i$,
3. (9 puntos) $1 \leq n \leq 500$, $1 \leq m \leq 1000$,
4. (16 puntos) $1 \leq n \leq 4000$, $1 \leq m \leq 1\,000\,000$,
5. (19 puntos) $1 \leq n \leq 50\,000$, $1 \leq k \leq 100$, $1 \leq m \leq 1\,000\,000$,
6. (40 puntos) $1 \leq n \leq 100\,000$, $1 \leq m \leq 1\,000\,000$.

Ejemplo de "Grader"

El Grader de muestra lee las entradas con el siguiente formato:

- línea 1: integers n , m y k ,
- línea $2 + i$ ($0 \leq i \leq n - 1$): enteros r_i y c_i .