

## Alienígenas

Nuestro satélite acaba de descubrir una civilización alienígena en un planeta remoto. Ya hemos obtenido una foto en baja resolución de un área de forma cuadrada del planeta. La foto muestra muchos signos de vida inteligente. Nuestros expertos han identificado  $n$  puntos de interés en la foto. Los puntos de interés se numeran de  $0$  a  $n - 1$ . Ahora queremos tomar fotos en alta resolución que contengan todos esos  $n$  puntos.

Internamente, el satélite ha dividido el área de la foto en baja resolución en una cuadrícula de  $m$  por  $m$  celdas. Tanto filas como columnas de la cuadrícula son consecutivamente numeradas a partir de  $0$  y hasta  $m - 1$  (desde la parte superior y desde la izquierda, respectivamente). Utilizamos el par  $(s, t)$  para denotar la celda en la fila  $s$  y la columna  $t$ . El punto de interés  $i$  se encuentra en la celda  $(r_i, c_i)$ . Cada celda puede contener un número arbitrario de estos puntos.

Nuestro satélite está en una órbita estable que pasa directamente sobre la diagonal *principal* de la cuadrícula. La diagonal principal es el segmento de recta que conecta la esquina superior izquierda y la esquina inferior derecha de la cuadrícula. El satélite puede tomar una foto en alta resolución de cualquier área que satisfaga las siguientes limitaciones:

- La forma del área es un cuadrado,
- Dos esquinas opuestas del cuadrado se encuentran ambas en la diagonal principal de la cuadrícula,
- Cada celda de la cuadrícula esta completamente dentro o completamente fuera del área fotografiada.

El satélite es capaz de tomar como máximo  $k$  fotos en alta resolución.

Una vez que el satélite ha tomado las fotos, transmitirá la foto en alta resolución de cada celda fotografiada a nuestra base de operaciones (independientemente de si la celda contiene algunos puntos de interés).

Los datos para cada celda fotografiada sólo se transmitirá *una vez*, incluso si fue fotografiada varias veces.

Por lo tanto, tenemos que elegir como máximo  $k$  áreas cuadradas que se pueden fotografiar, asegurando que:

- Cada celda que contiene por lo menos un punto de interés sea fotografiada al menos una vez, y
- El número de celdas que se fotografió al menos una vez se reduzca al mínimo.

Su tarea es encontrar el número total más pequeño posible de celdas fotografiadas.

### Detalle de la implementación

Usted debe implementar la siguiente función (método):

- `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 columnas) en la cuadrícula,
  - `k`: el número máximo de fotos que el satélite puede tomar,
  - `r` y `c`: dos arreglos de longitud `n` que describen las coordenadas de las celdas de la cuadrícula que contienen los puntos de interés. Para  $0 \leq i \leq n - 1$ , el  $i$ -ésimo punto de interés se encuentra en la celda  $(r[i], c[i])$ ,
  - La función debe devolver el número total posible más pequeño de celdas que se fotografió al menos una vez (respetando que la foto debe cubrir todos los puntos de interés).

Por favor utilice los archivos de ejemplos provistos para los detalles de implementación en su lenguaje de programación.

## Ejemplos

### Ejemplo 1

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

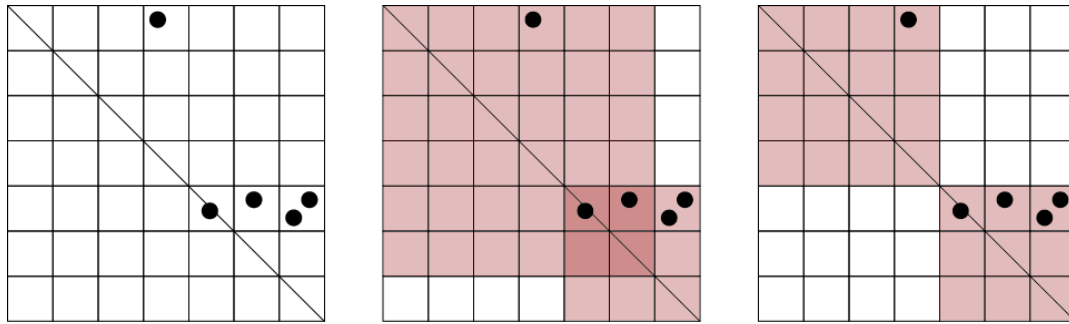
En este ejemplo, tenemos una cuadrícula de  $7 \times 7$  con 5 puntos de interés. Los puntos de interés se ubican en cuatro celdas diferentes:  $(0, 3)$ ,  $(4, 4)$ ,  $(4, 5)$  y  $(4, 6)$ . Puede tomar como máximo 2 fotos de alta resolución.

Una forma de capturar los cinco puntos de interés es hacer dos fotos: una con las celdas  $(0, 0)$  y  $(5, 5)$  en las esquinas opuestas, y la otra con celdas  $(4, 4)$  y  $(6, 6)$  en las esquinas opuestas. Si tomamos estas dos fotos, el satélite transmitirá las fotos de 41 celdas. Esta cantidad no es óptima.

La solución óptima utiliza una fotografía para capturar un cuadrado de  $4 \times 4$  que contiene las celdas  $(0, 0)$  y  $(3, 3)$  y otra foto para capturar un cuadrado de  $3 \times 3$  que contiene las celdas  $(4, 4)$  y  $(6, 6)$ . Esto da como resultado solamente 25 celdas fotografiadas, que es óptimo, por lo que `take_photos` debería devolver 25.

Observe que es suficiente fotografiar la celda  $(4, 6)$  una vez, a pesar de que contiene dos puntos de interés.

Este ejemplo se muestra en las siguientes figuras. La figura de la izquierda muestra la cuadrícula que corresponde a este ejemplo. La figura del medio muestra la solución subóptima en el que 41 celdas fueron fotografiadas. La figura de la derecha muestra la solución óptima.

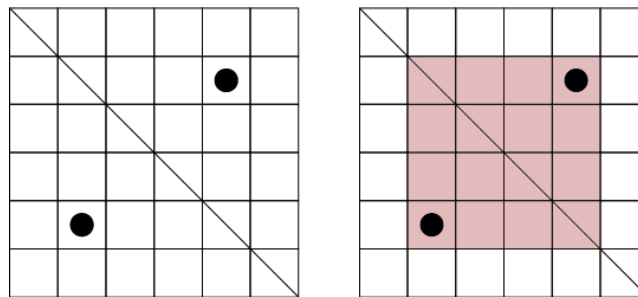


## Ejemplo 2

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

Aquí tenemos 2 puntos de interés situados simétricamente en las celdas:  $(1, 4)$  y  $(4, 1)$ . Cualquier foto válida que contiene uno de ellos contiene el otro también. Por lo tanto, es suficiente utilizar una sola foto.

Las figuras siguientes muestran este ejemplo y su solución óptima. En esta solución, el satélite capta 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 todo  $i$  tal 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)  $1 \leq n \leq 100\,000$ ,  $1 \leq m \leq 1\,000\,000$ .

## Evaluador de ejemplo

El evaluador de ejemplo lee la entrada en el siguiente formato:

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