



Aliens

El nostre satèl·lit acaba de descobrir una civilització alienígena en un planeta llunyà. Ja hem aconseguit una fotografia en baixa resolució d'una àrea quadrada del planeta. La fotografia mostra signes de vida intel·ligent. Els nostres experts han identificat n punts d'interès en la fotografia. Els punts estan numerats del 0 al $n - 1$. Ara volem fer fotografies d'alta resolució que continguin tots aquests n punts.

Internament, el satèl·lit ha dividit l'àrea de la fotografia de baixa resolució en una quadrícula de m per m unitats. Tant les files com les columnes de la quadrícula estan numerades consecutivament del 0 al $m - 1$ (començant per dalt a l'esquerra). Utilitzem (s, t) per denotar la casella a la fila s i columna t . El punt número i està situat a la casella (r_i, c_i) . Cada casella pot contenir un nombre arbitrari de punts.

El nostre satèl·lit és a una òrbita estable que travessa la diagonal *principal* de la quadrícula. La diagonal principal és el segment que connecta les cantonades superior esquerra i inferior dreta de la quadrícula. El satèl·lit pot fer una fotografia en alta resolució de qualsevol àrea que satisfaci les restriccions següents:

- la forma de l'àrea és quadrada,
- dues cantonades oposades del quadrat estan dins de la diagonal principal de la quadrícula,
- cada casella de la quadrícula està totalment dins o totalment fora de l'àrea fotografiada.

El satèl·lit pot fer un màxim de k fotografies en alta resolució.

Un cop el satèl·lit acabi de fer fotografies, transmetrà la fotografia en alta resolució de cada casella fotografiada a la nostra base (independentment de si conté o no punts d'interès). Les dades de cada casella fotografia només es transmeten *una* vegada, fins i tot si la casella s'ha fotografiat més d'una vegada.

Per tant, hem de triar un màxim de k àrees quadrades per fotografiar, de manera que:

- cada casella que contingui almenys un punt d'interès es fotografiï com a mínim una vegada, i
- el nombre de caselles que es fotografiïn almenys una vegada sigui tan petit com sigui possible.

La vostra tasca consisteix a trobar el nombre més petit possible de caselles fotografiades.

Detalls d'implementació

Se us demana que implementeu la següent funció (mètode):

- `int64 take_photos(int n, int m, int k, int[] r, int[] c)`
 - `n`: el nombre de punts d'interès,
 - `m`: el nombre de files (i de columnes) de la quadrícula,
 - `k`: el nombre màxim de fotografies que pot fer el satèl·lit,
 - `r` i `c`: dos arrays de mida `n` que descriuen les coordenades de les caselles de la quadrícula que contenen punts d'interès. Per a $0 \leq i \leq n - 1$, el punt d'interès i -èssim es troba a la casella $(r[i], c[i])$,
 - la funció ha de retornar el nombre més petit possible de caselles que es fotografiïn com a mínim una vegada (amb el benentès que les fotografies han de cobrir tots els punts d'interès).

Si us plau, feu servir les plantilles donades per veure els detalls d'implementació en el vostre llenguatge de programació.

Exemples

Exemple 1

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

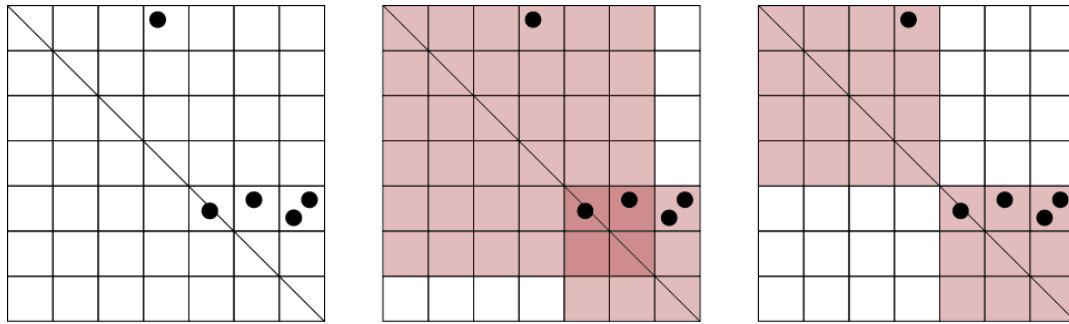
En aquest exemple tenim una quadrícula de 7×7 amb 5 punts d'interès. Els punts d'interès se situen a quatre caselles diferents: $(0, 3)$, $(4, 4)$, $(4, 5)$ i $(4, 6)$. Podeu fer un màxim de 2 fotografies en alta resolució.

Una manera possible de capturar tots els punts d'interès consisteix a fer dues fotografies: una del quadrat de 6×6 que conté les caselles $(0, 0)$ i $(5, 5)$, i l'altra del quadrat de 3×3 que conté caselles $(4, 4)$ i $(6, 6)$. Si el satèl·lit fa aquestes dues fotografies, transmetrà les dades de 41 caselles. Aquesta quantitat no és òptima.

La solució òptima fa una fotografia que captura el quadrat de 4×4 que conté les caselles $(0, 0)$ i $(3, 3)$ i una altra fotografia que captura el quadrat de 3×3 que conté les caselles $(4, 4)$ i $(6, 6)$. Això dóna un total de 25 caselles fotografiades, que és òptim, de manera que `take_photos` ha de retornar 25.

Fixeu-vos que n'hi ha prou amb fotografiar la casella $(4, 6)$ una única vegada, fins i tot si conté dos punts d'interès.

Aquest exemple es mostra en la figura següent. La figura de l'esquerra mostra la quadrícula que correspon a aquest exemple. La del mig mostra la solució subòptima on es fotografiën 41 caselles. La de la dreta mostra la solució òptima.

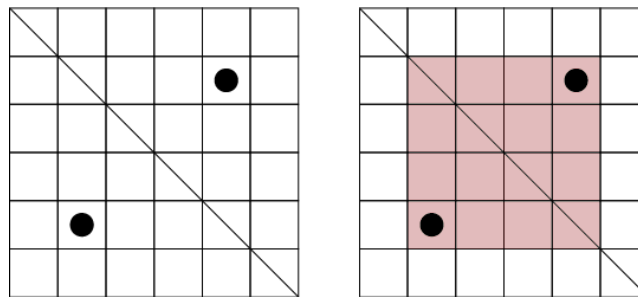


Exemple 2

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

Aquí tenim 2 punts d'interès siuats en posicions simètriques: a les caselles (1,4) i (4,1). Qualsevol fotografia vàlida que contingui un dels dos contindrà també l'altre. Per tant, n'hi ha prou amb fer una única fotografia.

Les figures a continuació mostren aquest exemple i la seva solució òptima. En aquesta solució el satèl·lit captura una única fotografia de 16 caselles.



Subtasques

Per a totes les subtasques, $1 \leq k \leq n$.

1. (4 punts) $1 \leq n \leq 50$, $1 \leq m \leq 100$, $k = n$,
2. (12 punts) $1 \leq n \leq 500$, $1 \leq m \leq 1000$, per a tot i tal que $0 \leq i \leq n - 1$,
 $r_i = c_i$,
3. (9 punts) $1 \leq n \leq 500$, $1 \leq m \leq 1000$,
4. (16 punts) $1 \leq n \leq 4000$, $1 \leq m \leq 1\,000\,000$,
5. (19 punts) $1 \leq n \leq 50\,000$, $1 \leq k \leq 100$, $1 \leq m \leq 1\,000\,000$,
6. (40 punts) $1 \leq n \leq 100\,000$, $1 \leq m \leq 1\,000\,000$.

Grader de mostra

El grader de mostra llegeix l'entrada en el format següent:

- línia 1: enters n , m i k ,
- línia $2 + i$ ($0 \leq i \leq n - 1$): enters r_i i c_i .