



## Այլմոլորակայիններ

Մեր արբանյակը հենց նոր հայտնաբերել է այլմոլորակայինների քաղաքակրթություն մի հեռավոր մոլորակի վրա: Մենք արդեն ստացել ենք մոլորակի ինչ-որ քառակուսի տարածքի ցածր որակի լուսանկար: Լուսանկարում կան քաղաքակրթ կյանքի բազմաթիվ նշաններ: Մեր մասնագետները լուսանկարի վրա նշել են  $n$  հետաքրքիր կետեր: Կետերը համարակալված են  $0$ -ից  $n - 1$  թվերով: Այժմ մենք ուզում ենք կատարել բարձր որակի լուսանկարներ, որոնք կպարունակեն բոլոր հետաքրքիր  $n$  կետերը:

Արբանյակը ցածր որակով լուսանկարում պատկերված տարածքը բաժանել է  $m \times m$  քառակուսային ցանցի: Ցանցի տողերը և սյուները համարակալված են  $0$ -ից  $m - 1$  թվերով (տողերը համարակալված են վերևից ներքև, իսկ սյուները՝ ձախից աջ):  $(s, t)$ -ով կնշանակենք  $s$ -րդ տողի  $t$ -րդ վանդակը:  $i$  համարով հետաքրքիր կետը գտնվում է  $(r_i, c_i)$  վանդակում: Յուրաքանչյուր վանդակ կարող է պարունակել կամայական թվով հետաքրքիր կետեր:

Մեր արբանյակը այժմ շարժվում է անփոփոխ ուղեծրով, որը անցնում է վերոնշյալ քառակուսային ցանցի գլխավոր անկյունագծի ուղիղ վերևով: Ցանցի գլխավոր անկյունագծի ցանցի վերին ամենաձախ և ստորին ամենաաջ կետերը միացնող հատվածն է: Արբանյակը կարող է բարձր որակով լուսանկարել այնպիսի տարածքներ, որոնք բավարարում են հետևյալ պայմաններին՝

- տարածքը քառակուսու տեսքի ունի,
- այդ քառակուսու երկու հակադիր գագաթները ընկած են ցանցի գլխավոր անկյունագծի վրա,
- ցանցի ցանկացած վանդակ կա՛մ ամբողջովին գտնվում է լուսանկարված տարածքում, կա՛մ ամբողջովին դուրս է նրանից:

Արբանյակը կարող է կատարել առավելագույնը  $k$  հատ բարձր որակի լուսանկար:

Լուսանկարներ կատարելուց հետո արբանյակը ցանցի յուրաքանչյուր լուսանկարված վանդակի բարձր որակի լուսանկարը ուղարկում է երկիր մոլորակ (անկախ նրանից պարունակում է այդ վանդակը հետաքրքիր կետ, թե՛ ոչ): Յուրաքանչյուր լուսանկարված վանդակի լուսանկարը միայն մեկ անգամ է ուղարկվում երկիր, անկախ նրանից թե քանի անգամ է այդ վանդակը լուսանկարվել:

Այսպիսով մենք պետք է լուսանկարելու համար ընտրենք  $k$  հատ քառակուսի տարածք այնպես, որ՝

- առնվազն մեկ հետաքրքիր կետ պարունակող յուրաքանչյուր վանդակ

լուսանկարվի գոնե մեկ անգամ,

- առնվազն մեկ անգամ լուսանկարված վանդակների քանակը պետք է լինի հնարավորինս քիչ:

Ձեր խնդիրն է գտնել լուսանկարված վանդակների հնարավոր ամենափոքր քանակը:

## Իրականացման մանրամասներ

Դուք պետք է իրականացնեք հետևյալ ֆունկցիան`

- `int64 take_photos(int n, int m, int k, int[] r, int[] c)`
  - `n`: հետաքրքիր կետերի քանակը,
  - `m`: ցանցի տողերի (ինչպես նաև սյուների) քանակը,
  - `k`: առավելագույնը քանի բարձր որակի նկար է կարող կատարել արբանյակը,
  - `r` և `c`: `n` երկարությամբ զանգվածներ, որոնք նկարագրում են հետաքրքիր կետերը:  $i$ -րդ ( $0 \leq i \leq n - 1$ ) հետաքրքիր կետը գտնվում է  $(r[i], c[i])$  վանդակում,
  - ֆունկցիան պետք է վերադարձնի առնվազն մեկ անգամ լուսանկարված վանդակների հնարավոր ամենափոքր քանակը, պայմանով, որ բոլոր հետաքրքիր կետերը լուսանկարվեն գոնե մեկ անգամ:

## Օրինակներ

### Օրինակ 1

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

Այս օրինակում ունենք  $7 \times 7$  չափի ցանց` 5 հետաքրքիր կետերով: Հետաքրքիր կետերը գտնվում են հետևյալ 4 իրարից տարբեր վանդակներում`  $(0, 3)$ ,  $(4, 4)$ ,  $(4, 5)$  և  $(4, 6)$ : Կարելի է կատարել ամենաշատը 2 բարձր որակի լուսանկար:

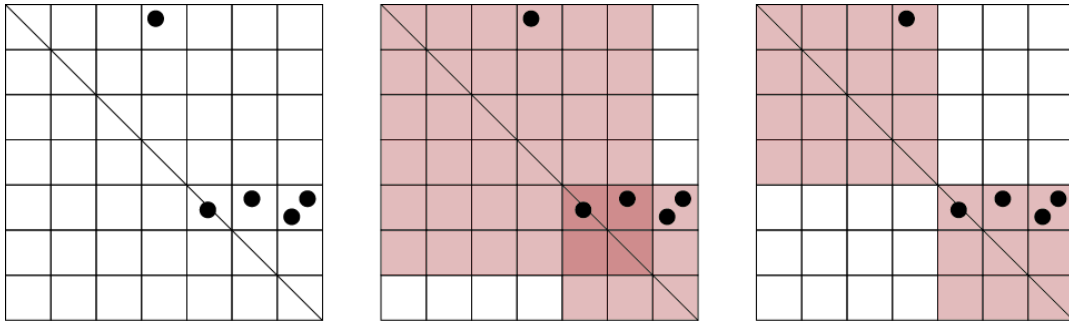
Բոլոր 5 հետաքրքիր կետերը լուսանկարելու տարբերակներից մեկը հետևյալն է` լուսանկարել  $6 \times 6$  չափի տարածք, որը պարունակում է  $(0, 0)$  և  $(5, 5)$  կետերը և լուսանկարել  $3 \times 3$  չափի տարածք, որը պարունակում է  $(4, 4)$  և  $(6, 6)$  կետերը: Եթե արբանյակը կատարի այս երկու լուսանկարները, ապա անհրաժեշտ կլինի երկիր ուղարկել 41 վանդակի լուսանկար: Այս քանակը օպտիմալ չէ:

Օպտիմալ լուծման դեպքում կատարում ենք 2 լուսանկար` առաջինը`  $(0, 0)$  և  $(3, 3)$  կետերը պարունակող  $4 \times 4$  չափի տարածք, երկրորդը`  $(4, 4)$  և  $(6, 6)$  կետերը պարունակող  $3 \times 3$  չափի տարածք: Արդյունքում լուսանկարվում է 25 վանդակ, ինչը օպտիմալ է, հետևաբար, `take_photos` ֆունկցիան պետք է վերադարձնի 25:

Նկատենք, որ  $(4, 6)$  վանդակը բավարար է լուսանկարել միայն մեկ անգամ, անկախ նրանից, որ այն պարունակում է երկու հետաքրքիր կետ:

Այս օրինակը պատկերված է ստորև բերված նկարում: Նկարի ձախ մասում

պատկերված է օրինակում նկարագրված ցանցը: Նկարի միջնամասում պատկերված է ոչ օպտիմալ լուծումը, որում **41** վանդակ է լուսանկարվում: Նկարի աջ մասում պատկերված է օպտիմալ լուծումը:



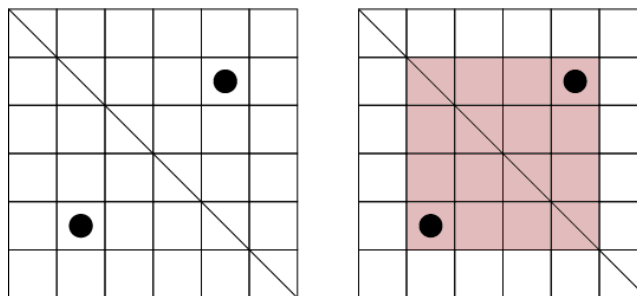
## Օրինակ 2

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

Այս օրինակում ունենք սիմետրիկ դասավորված **2** հետաքրքիր կետ:

Հետաքրքիր կետերը գտնվում են **(1,4)** և **(4,1)** վանդակներում: Ցանակացած թույլատրելի լուսանկար, որը պարունակում է կետերից մեկը, պարունակում է նաև մյուսը, ուստի բավարար է կատարել միայն մեկ լուսանկար:

Ստորև բերված նկարում պատկերված է այս օրինակը և համապատասխան օպտիմալ լուծումը: Այդ լուծման մեջ լուսանկարվում է **16** վանդակ:



## Ենթախնդիրներ

Բոլոր ենթախնդիրներում  $1 \leq k \leq n$ :

1. (4 միավոր)  $1 \leq n \leq 50$ ,  $1 \leq m \leq 100$ ,  $k = n$ ,
2. (12 միավոր)  $1 \leq n \leq 500$ ,  $1 \leq m \leq 1000$ , բոլոր  $0 \leq i \leq n - 1$  համար  $r_i = c_i$ ,
3. (9 միավոր)  $1 \leq n \leq 500$ ,  $1 \leq m \leq 1000$ ,
4. (16 միավոր)  $1 \leq n \leq 4000$ ,  $1 \leq m \leq 1\,000\,000$ ,
5. (19 միավոր)  $1 \leq n \leq 50\,000$ ,  $1 \leq k \leq 100$ ,  $1 \leq m \leq 1\,000\,000$ ,
6. (40 միավոր)  $1 \leq n \leq 100\,000$ ,  $1 \leq m \leq 1\,000\,000$ .

## Գրեյդերի օրինակ

Գրեյդերի օրինակը մուտքը ստանում է հետևյալ ֆորմատով՝

- տող 1: 3 ամբողջ թիվ՝  $n$ ,  $m$  և  $k$ ,
- տող  $2 + i$  ( $0 \leq i \leq n - 1$ ): 2 ամբողջ թիվ՝  $r_i$  և  $c_i$ :