



## International Olympiad in Informatics 2016

12-19th August 2016

Kazan, Russia

day2\_3

aliens  
Country: ISR

# Aliens

הלוויין שלנו (satellite) גילה חייזרים על כוכב לכת מרוחק. ברשותנו קיימת כבר תמונה ברזולוציה נמוכה של איזור ריבועי מתוך כוכב הלכת. בתמונה רואים סימנים רבים לחיים תבוניים. המומחים שלנו זיהו  $n$  נקודות מעניינות בתמונה. הנקודות ממוספרות מ-0 עד  $n - 1$ . כעת אנו רוצים לצלם תמונות ברזולוציה גבוהה שמכילות את כל  $n$  הנקודות הללו.

הלוויין חילק את האיזור של התמונה עם הרזולוציה הנמוכה לגריד של  $m$  על  $m$  תאים. גם השורות וגם העמודות של הגריד ממוספרות מ-0 עד  $m - 1$  (מלמעלה ומשמאל, בהתאמה). נשתמש ב- $(s, t)$  כדי לסמן את התא בשורה  $s$  ועמודה  $t$ . נקודה שמספרה  $i$  נמצאת בתא  $(r_i, c_i)$ . כל תא יכול להכיל מספר כלשהו של נקודות.

הלוויין שלנו חג בצורה קבועה ועובר בדיוק מעל האלכסון הראשי של הגריד. האלכסון הראשי הוא הישר שמחבר את הפינה השמאלית העליונה עם הפינה הימנית התחתונה של הגריד. הלוויין יכול לצלם תמונות ברזולוציה גבוהה של כל איזור שמקיים את התנאים הבאים:

- צורת האיזור היא ריבוע,
- שתי פינות נגדיות של הריבוע נמצאות על האלכסון הראשי של הגריד,
- כל תא של הגריד נמצא כולו בפנים או כולו מחוץ לאיזור המצולם.

הלוויין מסוגל לצלם לכל היותר  $k$  תמונות ברזולוציה גבוהה.

ברגע שהלוויין מסיים לצלם את התמונות, הוא שולח תמונת רזולוציה גבוהה של כל תא לבסיס האם (בין אם התא מכיל נקודות מעניינות ובין אם לא). המידע על כל תא מצולם יישלח פעם אחת בלבד, גם אם התא צולם מספר פעמים.

לכן, עלינו לבחור לכל היותר  $k$  איזורים ריבועיים שיצולמו, המבטיחים שיתקיים:

- כל תא שמכיל לפחות נקודה מעניינת אחת מצולם לפחות פעם אחת, וגם
- מספר התאים שמצולמים לפחות פעם אחת הוא מינימלי.

משימתכם היא למצוא את המספר הכולל המינימלי האפשרי של תאים מצולמים.

## פרטי מימוש

עליכם לממש את הפונקציה הבאה (שיטה):

• `int64 take_photos(int n, int m, int k, int[] r, int[] c)`

- $n$ : מספר הנקודות המעניינות,
- $m$ : מספר השורות (והעמודות) בגריד,
- $k$ : המספר המקסימלי של תמונות שהלוויין יכול לצלם,
- $r$  ו- $c$ : שני מערכים באורך  $n$  המתארים את הקואורדינטות של תאי הגריד שמכילים נקודות מעניינות. עבור  $0 \leq i \leq n-1$ , הנקודה המעניינת ה- $i$  נמצאת בתא  $(r[i], c[i])$ ,
- הפונקציה צריכה להחזיר את המספר הכולל המינימלי האפשרי של תאים שמצולמים לפחות פעם אחת (בהינתן שהתמונות חייבות להכיל את כל הנקודות המעניינות).

אנא השתמשו בקבצי ה-`template` לפרטי המימוש עבור שפת התכנות שלכם.

## דוגמאות

### דוגמה 1

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

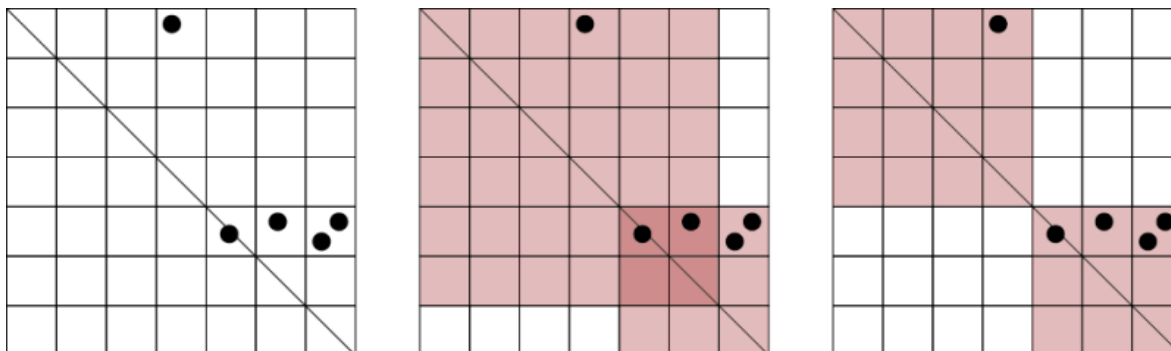
בדוגמה זו יש לנו גריד בגודל  $7 \times 7$  עם 5 נקודות מעניינות. הנקודות המעניינות ממוקמות בארבעה תאים שונים:  $(0, 3)$ ,  $(4, 4)$ ,  $(4, 5)$  ו- $(4, 6)$ . עליכם לצלם לכל היותר 2 תמונות ברזולוציה גבוהה.

דרך אחת לתפוס את כל חמש הנקודות המעניינות היא לצלם שתי תמונות: תמונה של ריבוע בגודל  $6 \times 6$  המכילה את התאים  $(0, 0)$  ו- $(5, 5)$ , ותמונה של ריבוע בגודל  $3 \times 3$  המכילה את התאים  $(4, 4)$  ו- $(6, 6)$ . אם הלוויין יצלם את שתי התמונות הללו, הלוויין ישלח את המידע על 41 תאים. מספר זה אינו אופטימלי.

הפתרון האופטימלי משתמש בתמונה ראשונה כדי לצלם ריבוע בגודל  $4 \times 4$  שמכיל את התאים  $(0, 0)$  ו- $(3, 3)$  ותמונה נוספת כדי לצלם ריבוע בגודל  $3 \times 3$  שמכיל את התאים  $(4, 4)$  ו- $(6, 6)$ . כך יהיו רק 25 תאים מצולמים, וזה אופטימלי, לכן `take_photos` צריכה להחזיר 25.

שימו לב שמספיק לצלם את התא  $(4, 6)$  פעם אחת, למרות שהוא מכיל שתי נקודות מעניינות.

ניתן לראות דוגמה זו באיור להלן. האיור השמאלי ביותר מראה את הגריד שמתאים לדוגמה. האיור האמצעי מראה את הפתרון הלא אופטימלי שבו 41 תאים מצולמים. האיור הימני מראה את הפתרון האופטימלי.

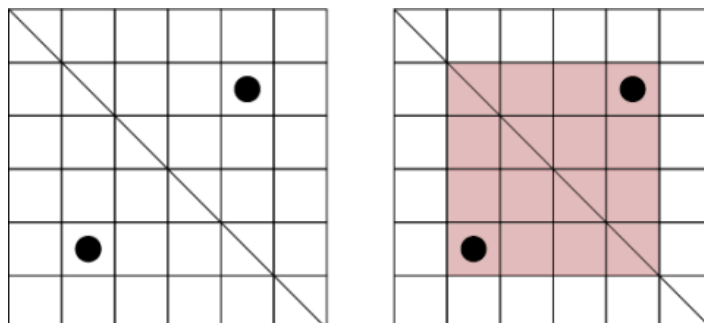


## דוגמה 2

take\_photos(2, 6, 2, [1, 4], [4, 1])

כאן יש לנו 2 נקודות מעניינות הממוקמות באופן סימטרי: בתאים (1, 4) ו-(4, 1). כל תמונה תקינה שמכילה אחד מהם מכילה גם את האחר. לכן, מספיק להשתמש בתמונה אחת.

האיור להלן מראה את הדוגמה ואת הפתרון האופטימלי שלה. בפתרון זה הלוויין מצלם תמונה אחת של 16 תאים.



## תת משימות

בכל תת המשימות מתקיים  $1 \leq k \leq n$ .

1. (4 נקודות):  $1 \leq n \leq 50$  וגם  $1 \leq m \leq 100$  וגם  $k = n$ .
2. (12 נקודות):  $1 \leq n \leq 500$  וגם  $1 \leq m \leq 1000$  וגם לכל  $i$  כך ש- $0 \leq i \leq n - 1$  מתקיים  $r_i = c_i$ .
3. (9 נקודות):  $1 \leq n \leq 500$  וגם  $1 \leq m \leq 1000$ .
4. (16 נקודות):  $1 \leq n \leq 4000$  וגם  $1 \leq m \leq 1\,000\,000$ .
5. (19 נקודות):  $1 \leq n \leq 50\,000$  וגם  $1 \leq k \leq 100$  וגם  $1 \leq m \leq 1\,000\,000$ .
6. (40 נקודות):  $1 \leq n \leq 100\,000$  וגם  $1 \leq m \leq 1\,000\,000$ .

## גריידר לדוגמה

הגריידר קורא את הקלט בפורמט הבא:

- שורה 1: המספר  $n$  ואחריו  $m$  ואחריו  $k$ .
- שורה  $i + 2$  ( $0 \leq i \leq n - 1$ ): המספר  $r_i$  ואחריו  $c_i$ .