

עיר אידאלית

מסתבר שלאונרדו היה גם מתכנן ערים. הוא התעניין בערים אידאליות (Ideal City - מושג שיוגדר בהמשך). כשהוא ראה עיר אידאלית, עניין אותו לנתח תכונות מסוימות שלה.

הגדרת עיר אידאלית

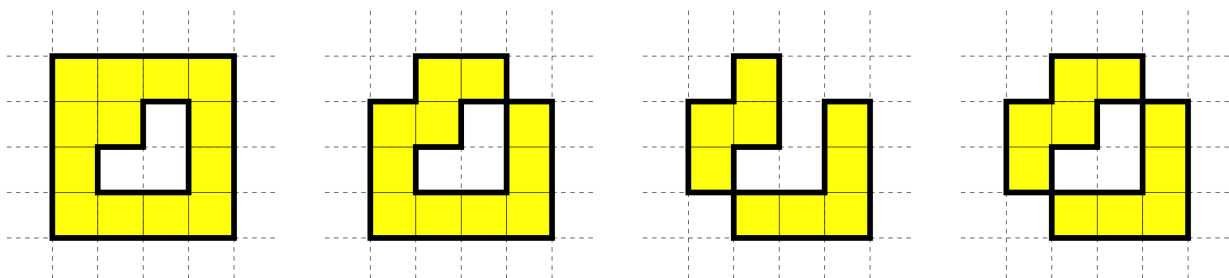
העיר בנוייה על-גבי לוח משבצות. בלוח, המשבצת $(0,0)$ היא השמאלית העליונה. המשבצת $(2^{31}-1, 2^{31}-1)$ היא המשבצת הימנית תחתונה. משבצות יקראו "סמוכות" אם הן גובלות אופקית או אנכית (אך לא באלכסון).

העיר עצמה מורכבת מ- N בלוקים בנויים (Blocks) שמסומנים באיורים בצהוב. בלוקים בנויים יכולים להיות בכל מקום בלוח, פרט למסגרת החיצונית. כלומר, בלוק מוגדר על-ידי קואורדינטות (i,j) כך שמתקיים $1 \leq i, j \leq 2^{31}-2$. עיר היא אידאלית אם מתקיים שהשטח הבנוי של העיר הוא קשיר וגם השטח הלא-בנוי של העיר הוא קשיר. כלומר, עיר היא אידאלית אם ורק אם מתקיימים שני התנאים הבאים:

- כל שתי משבצות ריקות בלוח מחוברות ביניהן באמצעות מסלול של משבצות ריקות. כלומר, אפשר ללכת מכל משבצת ריקה לכל משבצת ריקה אחרת דרך משבצות ריקות (כשעוברים בכל צעד ממשבצת למשבצת סמוכה).
- כל שתי משבצות בנויות בלוח מחוברות ביניהן באמצעות מסלול של משבצות בנויות. כלומר, אפשר ללכת מכל משבצת בנויה לכל משבצת בנויה אחרת דרך משבצות בנויות (כשעוברים בכל צעד ממשבצת למשבצת סמוכה).

דוגמא 1

כל האיורים שלהלן מתארים ערים שאינן אידאליות. שתי הערים הראשונות משמאל לא מקיימות את התנאי הראשון. העיר השלישית לא מקיימת את התנאי השני. העיר הרביעית לא מקיימת אף אחד מהתנאים.



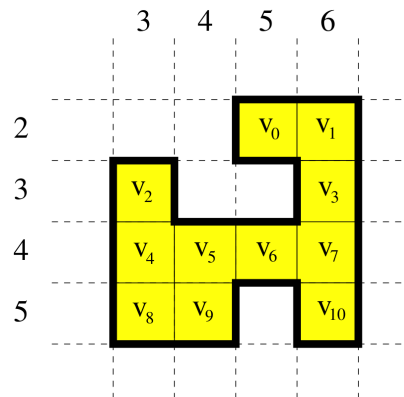
הגדרה של מרחק

נתעניין במרחקים בין זוגות של בלוקים בנויים בעיר. נגדיר את המרחק בין בלוקים בנויים באופן הבא: יהיו v_i ו- v_j שני בלוקים בנויים בעיר. המרחק ביניהם הוא מספר הצעדים המינימלי הדרוש כדי ללכת מ- v_i ל- v_j , דרך בלוקים בנויים בלבד (כאמור, בצעד אחד אפשר ללכת מבלוק בנוי, לבלוק בנוי אחר שסמוך לו אופקית או אנכית).

דוגמא 2

האיור שלהלן מתאר עיר אידאלית שבנוייה מ- $N = 11$ בלוקים בנויים. הקואורדינטות שלהם הן: $v_0 = (2, 5)$, $v_1 = (2, 6)$, $v_2 = (3, 3)$, $v_3 = (3, 6)$, $v_4 = (4, 3)$, $v_5 = (4, 4)$, $v_6 = (4, 5)$, $v_7 = (4, 6)$, $v_8 = (5, 3)$, $v_9 = (5, 4)$, $v_{10} = (5, 6)$

הנה כמה מרחקים לדוגמא: $d(v_6, v_{10}) = 2$, and $d(v_9, v_{10}) = 4$



תיאור הבעייה

נתון תיאור של עיר אידאלית. המשימה היא לחשב את סכום המרחקים בין כל זוגות הבלוקים הבנויים בעיר. מכיוון שסכום זה יכול להיות גדול, מאד, אתם תחשבו אותו מודולו 1,000,000,000. כלומר, על התוכנית שלכם לחשב את הסכום הבא מודולו 1,000,000,000:

$$\sum d(v_i, v_j), \text{ where } 0 \leq i < j \leq N - 1$$

מבחינת מימוש, עליכם לממש את הפונקציה $\text{DistanceSum}(N, X, Y)$. הפרמטר N הוא מספר טבעי שהוא מספר הבלוקים הבנויים בעיר. X ו- Y הם מערכים של מספרים שלמים שבכל אחד מהם N איברים. הבלוק הבנוי v_i נמצא בקואורדינטות $(X[i], Y[i])$, עבור $0 \leq i \leq N - 1$. מובטח שמתקיים $1 \leq X[i], Y[i] \leq 2^{31} - 2$. להזכירכם, עליכם להחזיר את סכום המרחקים מודולו 1,000,000,000 (מיליארד).

בדוגמא מספר 2, יש $11 \times 10 / 2 = 55$ זוגות בלוקים. סכום המרחקים בין הזוגות הוא 174.

[Subtask 1 [11 points

.You may assume that $N \leq 200$

[Subtask 2 [21 points

.You may assume that $N \leq 2\,000$

[Subtask 3 [23 points

.You may assume that $N \leq 100\,000$

בנוסף, מתקיימים שני התנאים הבאים:

- עבור כל שני בלוקים בנויים שנמצאים באותה שורה, גם כל המשבצות שביניהן הן בלוקים בנויים.
- עבור כל שני בלוקים בנויים שנמצאים באותה עמודה, גם כל המשבצות שביניהן הן בלוקים בנויים.

[Subtask 4 [45 points

.You may assume that $N \leq 100\,000$

Implementation details

You have to submit exactly one file, called `city.c`, `city.cpp` or `city.pas`. This file must implement the subprogram described above using the following signatures

C/C++ programs

```
int DistanceSum(int N, int *X, int *Y);
```

Pascal programs

```
function DistanceSum(N : LongInt; var X, Y : array of LongInt) : LongInt;
```

This subprogram must behave as described above. Of course you are free to implement other subprograms for its internal use. Your submissions must not interact in any way with standard input/output, nor with any other file

Sample grader

:The sample grader provided with the task environment will expect input in the following format

- ;line 1: N
- .[lines 2, ..., N + 1: X[i], Y[i]

Time and Memory limits

- .Time limit: 1 second
- .Memory limit: 256 MiB