

شهر ایده‌آل

لئوناردو، مانند دیگر دانشمندان و هنرمندان ایتالیایی هم‌عصر خود، شدیداً علاقمند به طراحی شهری بود. هدف او مدل کردن یک شهر ایده‌آل بود: شهری راحت با فضای زیاد و استفاده منطقی از منابع که مثل شهرهای قرون وسطی تنگ و دلگیر نباشد.

شهر ایده‌آل

این شهر با قرار دادن N بلوک بر روی یک گرید (صفحه) نامتناهی از خانه‌ها ساخته شده است. هر خانه از صفحه به وسیله یک زوج مختصات $(row, column) = (ستون, سطر)$ مشخص می‌شود. برای هر خانه (i, j) ، خانه‌های مجاور آن عبارتند از:

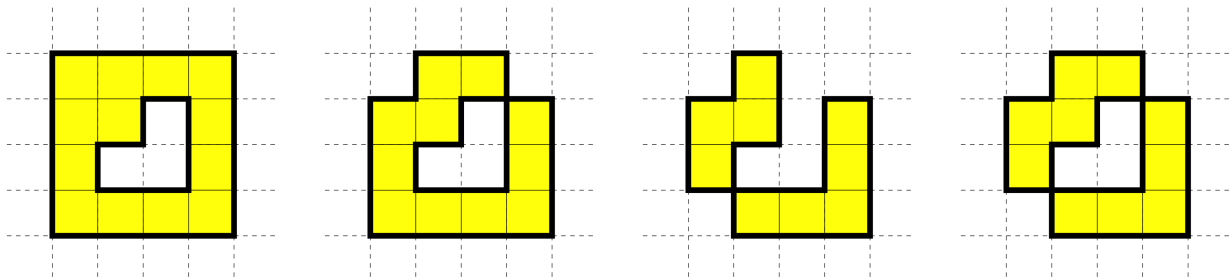
$$(i, j - 1), (i + 1, j), (i - 1, j), \text{ و } (i, j + 1).$$

یک بلوک وقتی روی صفحه قرار داده می‌شود، دقیقاً یک خانه از صفحه را اشغال می‌کند. یک بلوک می‌تواند در خانه (i, j) قرار بگیرد اگر و تنها اگر $1 \leq i, j \leq 2^{31} - 2$. ما همچنین از مختصات یک خانه برای ارجاع دادن به بلوکی که روی آن خانه قرار گرفته استفاده می‌کنیم. دو بلوک مجاور هستند، اگر در دو خانه مجاور قرار گرفته باشند. در یک شهر ایده‌آل، همه بلوک‌ها به یکدیگر وصل هستند و در شهر حفراهی وجود ندارد؛ به عبارت دقیق‌تر، خانه‌ها دو شرط زیر را برآورده می‌کنند:

- برای هر دو خانه "خالی" (خانه‌ای که روی آن بلوک نیست)، یک دنباله (یک مسیر) از خانه‌های خالی مجاور وجود داشته باشد که آن دو را به هم متصل کند.
- برای هر دو خانه پر، دنباله‌ای از خانه‌های پر مجاور وجود داشته باشد که آن دو را به هم متصل کند.

مثال ۱

هیچکدام از شکل‌های زیر یک شهر ایده‌آل را نمایش نمی‌دهد. دو شکل اول (از سمت چپ) شرط اول، شکل سوم شرط دوم و شکل چهارم هیچکدام از دو شرط را برآورده نمی‌کنند.



فاصله

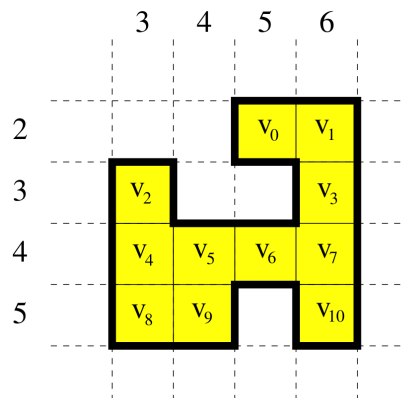
هنگام گشت و گذار در شهر، یک قدم را خارج شدن از یک بلوک و وارد شدن به بلوک مجاورش تعریف می‌کنیم. طبیعی است که نمی‌توان وارد خانه‌های خالی شد و در آن‌ها گشت و گذار کرد. فرض کنید v_0, v_1, \dots, v_{N-1} مختصات N بلوک قرار گرفته بر روی صفحه باشند. به ازای هر دو بلوک متفاوت v_i و v_j ، فاصله‌ی آنها $d(v_i, v_j)$ کمترین تعداد قدمی است که ما را از یکی به دیگری می‌رساند.

مثال ۲

شکل زیر یک شهر ایده‌آل را که از قرار گرفتن $N = 11$ بلوک در مختصات‌های زیر بدست آمده است را نشان می‌دهد:

$$v_0 = (2, 5), v_1 = (2, 6), v_2 = (3, 3), v_3 = (3, 6), v_4 = (4, 3), v_5 = (4, 4), v_6 = (4, 5), v_7 = (4, 6), v_8 = (5, 3), v_9 = (5, 4) \text{ و } v_{10} = (5, 6).$$

بعنوان مثال $d(v_6, v_{10}) = 2$ ، $d(v_1, v_8) = 6$ ، $d(v_1, v_3) = 1$ و $d(v_9, v_{10}) = 4$.



شرح مساله

وظیفه شما نوشتن برنامه‌ای است که مجموع فاصله هر دو بلوک v_i و v_j برای هر $i < j < N$ را محاسبه کند. در واقع، برنامه شما باید جمع زیر را محاسبه کند:

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

بطور خاص، شما باید تابع $DistanceSum(N, X, Y)$ را پیاده‌سازی کنید که بعنوان ورودی N تعداد بلوک‌ها، X و Y دو آرایه‌ای حاوی مختصات بلوک‌ها را دریافت و عبارت بالا را محاسبه کند. اندازه دو آرایه X و Y برابر N می‌باشد. دقت کنید بلوک i ام در مختصات $(X[i], Y[i])$ قرار دارد که $0 \leq i \leq N - 1$ و $1 \leq X[i], Y[i] \leq 2^{31} - 2$. از آنجا که ممکن است نتیجه آنقدر بزرگ شود که در ۳۲ بیت جا نشود، شما باید آن را به پیمانه (مد) 1000000000 (یک میلیارد) محاسبه کنید.

در مثال ۲ تعداد $55 = 10 \times 11 / 2$ زوج بلوک موجود است. مجموع همه زوج فاصله‌ها برابر 174 می‌شود.

زیرمساله ۱ [۱۱ نمره]

می‌توانید فرض کنید $N \leq 200$.

زیرمساله ۲ [۲۱ نمره]

می‌توانید فرض کنید $N \leq 2000$.

زیرمساله ۳ [۲۳ نمره]

می‌توانید فرض کنید $N \leq 100000$.

همچنین دو شرط زیر هم برقرار است: هر دو خانه پر i و j که $X[i] = X[j]$ را در نظر بگیرید؛ همه خانه‌های بین آن‌ها نیز پر است. و نیز هر دو خانه پر i و j که $Y[i] = Y[j]$ را در نظر بگیرید؛ همه خانه‌های بین آن‌ها نیز پر است.

زیرمساله ۴ [۴۵ نمره]

می‌توانید فرض کنید $N \leq 100000$.

جزئیات پیاده‌سازی

شما باید دقیقاً یک برنامه به نام `city.c`، `city.cpp` یا `city.pas` ارسال کنید. این فایل باید زیربرنامه توصیف شده در بالا را با مشخصه زیر پیاده‌سازی کند.

C/C++ programs

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

Pascal programs

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

این زیربرنامه باید مانند توصیف بالا رفتار کند. البته شما مختار هستید برای استفاده داخلی، زیربرنامه‌های دیگری در برنامه خودتان پیاده‌سازی کنید. برنامه ارسالی شما نباید با ورودی/خروجی استاندارد یا هر فایل دیگری تعامل داشته باشد.

مصحح نمونه

مصحح نمونه، مهیا شده برای محیط مساله، انتظار ورودی به فرمت زیر را دارد:

- سطر اول: N
- سطر ۲ تا $N+1$: $X[i]$ و $Y[i]$ (اول $X[i]$)

محدودیت‌های زمان و حافظه

- محدودیت زمان: یک ثانیه
- محدودیت حافظه: 256MiB