

مسئله پیدا کردن نزدیکترین دو نقطه

ورودی مسئله: n نقطه در فضای (دو بعدی) $P = \{p_1, p_2, \dots, p_n\}$



خروجی مسئله: $x, y \in P$ جله نزدیکترین
 $d(x, y) = \min_{i, j} \{d(p_i, p_j)\}$
 فاصله میان x و y
 فاصله اقلیدسی
 کمترین باشد

الگوریتم اول:

برای هر زوج نقاط فاصله را اندازه بگیر و بین زوج نقطه ای که کمترین فاصله را داشته باشند گزارش کن.

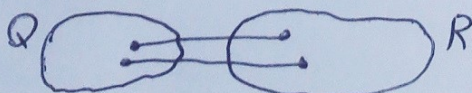
زمان اجرا $\Theta(n^2)$

الگوریتم براساس روش تقسیم و غلبه

$$P = Q \cup R$$

نقاط را به دو قسمت مساوی تقسیم می کنیم.

حقیقت را به طور مجزا حل می کنیم و پس نزدیکترین نقاط را میان Q و R پیدا می کنیم.



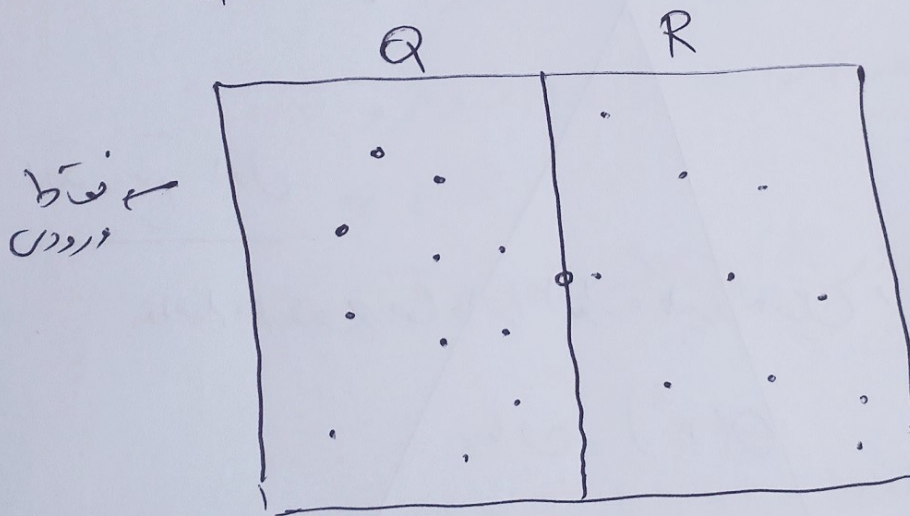
جزئیات الگوریتم

قدم اول: نقاط را بر اساس مولفه x مرتب کن
 لیست $Q_x \rightarrow$ نقاط مرتب شده بر اساس مولفه x

قدم دوم: نقاط را بر اساس مولفه y مرتب کن
 لیست $Q_y \rightarrow$ نقاط مرتب شده بر اساس مولفه y

قدم سوم: لیست Q_x را به دو قسمت مساوی تقسیم کن

$$Q_x = Q \cup R$$

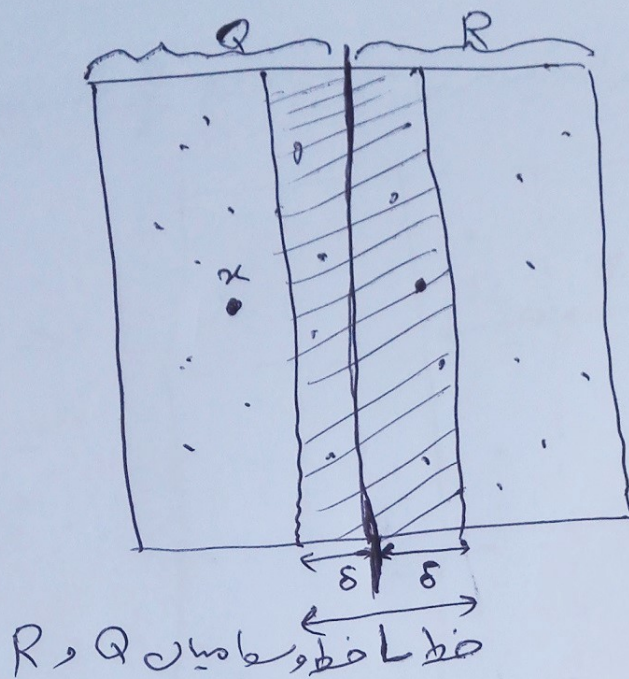


قدم چهارم: نزدیکترین زوج نقاط را در دو قسمت Q و R بصورت بازگشت پیدا کن

طول نزدیکترین فاصله در $Q \leftarrow \delta_1$

طول نزدیکترین فاصله در $R \leftarrow \delta_2$

$$\delta = \min \{ \delta_1, \delta_2 \}$$



قدم پنجم (قدم اصلی)
نزدیکترین نقاط را در
نوار هاشور زده پیدا می‌کنیم.
اگر فاصله‌شان از δ کمتر بود
آن را گزارش می‌کنیم در غیر این
صورت δ به عنوان کمترین
فاصله گزارش می‌شود.

اولین مشاهده: نقاطی که ~~در خارج~~ سمت هاشور زده هستند
ممکن است بتوانند جزو نزدیکترین نقاط باشند.

فرض کنید x در خارج نوار هاشور زده باشد و جزو Q باشد.

$$x \in Q$$

فاصله x از هر نقطه در R از δ بیشتر خواهد بود پس x فاصله‌اش از خط L
از δ بیشتر است.

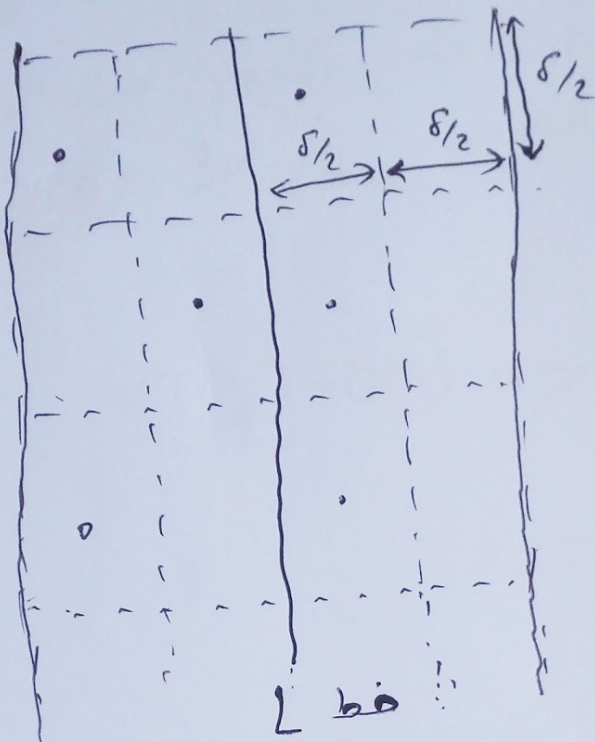
پس کافی است تنها نقاط داخل نوار هاشور زده را در نظر بگیریم.
زوج نقاطی که یکی در Q و دیگری در R باشد.

فرض کنید لیست Y نقاط داخل نوار هاشور زده باشد که برای آن مولفه y
مرتبه شده اند. لیست Y را می‌توان با استفاده از لیست مرتب Y در زمان
 $O(n)$ ساخت.

ممکن است کل نقاط در داخل Y باشند!

محدوده داخل نوار را به یک سری مربع با ضلع $\frac{\delta}{2}$ تقسیم می‌کنیم

در هر مربع حداکثر
یک نقطه وجود دارد
۱/۲

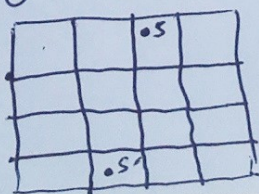


مشاهده: اگر دو نقطه s, s' داخل نوار باشند که $d(s, s') < \delta$
آنگاه در لیست مرتب y_k نقاط s و s' حداکثر ۱۵ اندیس با هم فاصله دارند.

.....
لیست s
شامل حداکثر ۱۵ افقطه است.

برهان خلف

فرض کنید دو نقطه s و s' وجود دارند که در لیست مرتب y_k ۱۶ اندیس با هم فاصله دارند.
فرض کنید نقطه s مولفه i کمتری دارد. (در لیست مرتب y_k زودتر آمده است)
چون در هر مربع حداکثر یک نقطه وجود دارد، نقطه s' باید حداقل ۳ طبقه پایین تر از s
در مربع بعدی یا در مربع



پس فاصله s و s' از y_k بیشتر است!
تناقض!

در نتیجه کافی است در لیست مرتب یک نقطه را با مدکثر 15
نقطه بتدا آن مقایسه کنیم. مقدار کل مقایسه ها در اکثر
15n خواهد بود. پس زمان اجرا $O(n)$ خواهد بود.

زمان اجرای کل الگوریتم

$$T(n) = \begin{cases} 2T(n/2) + O(n) & n > 2 \\ 1 & n = 2 \end{cases}$$

$$T(n) = \Theta(n \log n)$$

$$T(n) = \begin{array}{l} \text{زمان اجرای} \\ \text{به کردن نزدیکترین} \\ \text{نقاط در صفت } Q \\ T(n/2) \end{array} + \begin{array}{l} \text{زمان اجرای} \\ \text{به کردن نزدیکترین} \\ \text{نقاط در صفت } R \\ T(n/2) \end{array} + \begin{array}{l} \text{زمان اجرای به کردن} \\ \text{نزدیکترین نقاط در} \\ \text{نوارها شور زده} \\ O(n) \end{array}$$