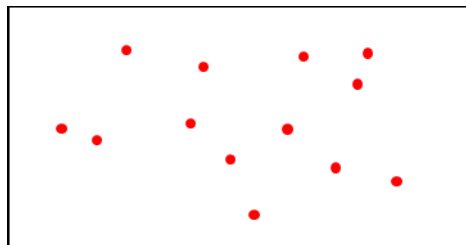




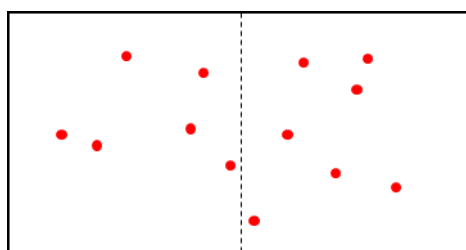
## بخش عملی تمرین ۲ (قسمت ۱)

### توضیح الگوریتم

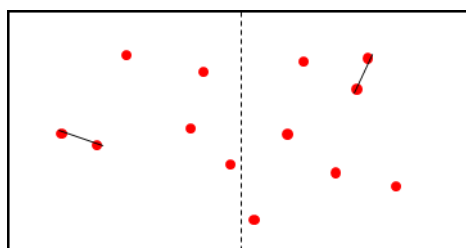
در این الگوریتم با آرایه‌ای از نقاط به طول  $n$  روبرو هستیم.



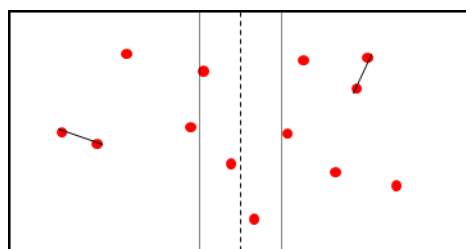
در هر مرحله باید قسمت مورد بررسی را به دو قسمت تقسیم میکنیم.



در هر قسمت به طور بازگشتی عمل میکنیم و minimum فاصله موجود را در هر سمت پیدا میکنیم و کوچک‌ترینشان را انتخاب میکنیم. حالت پایه این تقسیم بندی بازگشتی زمانی رخ میدهد که در قسمت تولید شده سه نقطه وجود داشته باشد که برای محاسبه آن از brute force استفاده میکنیم.



نکته‌ای که وجود دارد این است که ممکن است کوتاه‌ترین فاصله بین دو نقطه‌ای باشد که هر کدام در یک قسمت از دو قسمت چپ و راست باشد. برای این کار در فاصله مینیمم دو طرف که محاسبه کردیم نیز مینیمم را پیدا میکنیم. حال نقاط موجود را بر حسب  $y$  شان مرتب میکنیم.



حال کمترین حالت را برای دو عدد مینیممی که در اختیار داریم را باز میگردانیم.



## بخش عملی تمرین ۲ (قسمت ۱)

### تحلیل زمانی

تحلیل را با این فرض انجام می‌دهیم که از *sorting* ای استفاده می‌کنیم که  $O(n \lg n)$  دارد.

در آنجا که بررسی می‌کنیم ممکن است در دو طرف خط میانی نقاطی باشند که فاصله کمتر از مینیمم داشته باشند، می‌توانیم با  $O(n)$  مشخص کنیم کدام نقاط در این نواحی قرار خواهند گرفت. حال برای مرتب سازی این نقاط با توجه به فرضی که کردیم  $O(n \lg n)$  زمان می‌خواهیم. برای پیدا کردن مینیمم فاصله این نقاط به  $O(n)$  احتیاج داریم.

بنابراین می‌توان گفت:

$$T(n) = 2T\left(\frac{n}{2}\right) + \overbrace{O(n) + O(n \lg n) + O(n)}^{O(n \lg n)} = 2T\left(\frac{n}{2}\right) + O(n \lg n)$$

پس:

$$T(n) = O(n \lg n \lg n)$$