알고리즘

과제번호	06주차
날 짜	2018.10.18
학 번	201302395
이 름	류경빈

과제 2	Divide and Conquer을사용하여 closest pair를찾는 프로그램 구현

```java
public double divideAndConquer(ArrayList<Point> arrayList){
    arrayList.sort(Comparator.comparing(Point::getX));
    System.out.println(arrayList.size());

    if (arrayList.size() <= 3){
        double min = distanceXY(arrayList.get(0), arrayList.get(1));
        for (int i = 0; i < arrayList.size(); i++){
            for (int j = i+1; j < arrayList.size(); j++){
                double temp = distanceXY(arrayList.get(i), arrayList.get(j));
                if (min > temp){
                    min  = temp;
                }
            }
        }
        return min;
    }
    double first = divideAndConquer(new ArrayList<>(arrayList.subList(0, arrayList.size()/2)));
    double second = divideAndConquer(new ArrayList<>(arrayList.subList(arrayList.size()/2, arrayList.size())));
    double min = (first < second) ? first : second;

    ArrayList<Point> reduce = new ArrayList<>();
    Point mid = arrayList.get(arrayList.size()/2);
    for (int i = 0; i < arrayList.size(); i++) {
        if (Math.abs(arrayList.get(i).getX() - mid.getX()) < min) {
            reduce.add(arrayList.get(i));
        }
    }

    reduce.sort(Comparator.comparing(Point::getY));

    for (int i = 0; i < reduce.size(); i++) {
        for (int j = i + 1; j < reduce.size(); j++) {
            if (Math.abs(reduce.get(j).getY() - reduce.get(i).getY()) < min) {
                double temp = distanceXY(reduce.get(i), reduce.get(j));
                if (min > temp) {
                    min = temp;
                }
            }
        }
    }

    return min;
}
```

구현 결과 화면