Our Solution(s)   Run Code

Solution 1   Solution 2   Solution 3

```java
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

import java.util.*;

class Program {
  static String UP = "up";
  static String RIGHT = "right";
  static String DOWN = "down";
  static String LEFT = "left";

  // O(n^2) time | O(n^2) space - where n is the number of coordinates
  public static int rectangleMania(Point[] coords) {
    Map<String, Map<String, List<Point>>> coordsTable = getCoordsTable
    return getRectangleCount(coords, coordsTable);
  }

  public static Map<String, Map<String, List<Point>>> getCoordsTable(F
    Map<String, Map<String, List<Point>>> coordsTable =
        new HashMap<String, Map<String, List<Point>>>();
    for (Point coord1 : coords) {
      Map<String, List<Point>> coord1Directions = new HashMap<String,
      coord1Directions.put(UP, new ArrayList<Point>());
      coord1Directions.put(RIGHT, new ArrayList<Point>());
      coord1Directions.put(DOWN, new ArrayList<Point>());
      coord1Directions.put(LEFT, new ArrayList<Point>());
      for (Point coord2 : coords) {
        String coord2Direction = getCoordDirection(coord1, coord2);
        if (coord1Directions.containsKey(coord2Direction))
          coord1Directions.get(coord2Direction).add(coord2);
      }
      String coord1String = coordToString(coord1);
      coordsTable.put(coord1String, coord1Directions);
    }
```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```java
class Program {
  public static int rectangleMania(Point[] coords) {
    // Write your code here.
    return -1;
  }

  static class Point {
    public int x;
    public int y;

    public Point(int x, int y) {
      this.x = x;
      this.y = y;
    }
  }
}
```

Our Tests

Custom Output   Submit Code

Run or submit code when you're ready.

Run or submit code when you're ready.