

Solution 1

Solution 2

Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     let UP = "up"
5     let DOWN = "down"
6     let LEFT = "left"
7     let RIGHT = "right"
8
9     func coordToString(_ coord: [Int]) -> String {
10         let x = coord[0]
11         let y = coord[1]
12
13         return "\(x)-\(y)"
14     }
15
16     // O(n^2) time | O(n) space
17     func rectangleMania(_ coords: [[Int]]) -> Int {
18         let coordsTable = getCoordsTable(coords)
19         return getRectangleCount(coords, coordsTable)
20     }
21
22     func getCoordsTable(_ coords: [[Int]]) -> [String: [Int: [[Int]]]] {
23         var coordsTable: [String: [Int: [[Int]]]] = ["x": [:], "y": [:]]
24
25         for coord in coords {
26             let x = coord[0]
27             let y = coord[1]
28
29             if var tableAtX = coordsTable["x"], var coordinatesForX = tableAtX[x] {
30                 coordinatesForX.append(coord)
31                 tableAtX[x] = coordinatesForX
32                 coordsTable["x"] = tableAtX
33             } else if var tableAtX = coordsTable["x"] {
34                 tableAtX[x] = [coord]
35                 coordsTable["x"] = tableAtX
36             }
37
38             if var tableAtY = coordsTable["y"], var coordinatesForY = tableAtY[y] {
39                 coordinatesForY.append(coord)
40                 tableAtY[y] = coordinatesForY
41                 coordsTable["y"] = tableAtY
42             } else if var tableAtY = coordsTable["y"] {
43                 tableAtY[y] = [coord]
44                 coordsTable["y"] = tableAtY
45             }
46         }
47
48         return coordsTable
49     }
50
51     func getRectangleCount(_ coords: [[Int]], _ coordsTable: [String: [Int: [[Int]]]]) -> Int {
52         var rectangleCount = 0
53
54         for coord in coords {
55             let lowerLeftY = coord[1]
56             rectangleCount += clockwiseCountRectangles(coord, coordsTable, UP, lowerLeftY)
57         }
58
59         return rectangleCount
60     }
61
62     func clockwiseCountRectangles(_ coord: [Int], _ coordsTable: [String: [Int: [[Int]]]], _ direction: String, _ lowerLeftY: Int) -> Int {
63         let x1 = coord[0]
64         let y1 = coord[1]
65
66         if direction == DOWN {
67             if let tableAtX = coordsTable["x"], let relevantCoordinates = tableAtX[x1] {
68                 for coord2 in relevantCoordinates {
69                     let lowerRightY = coord2[1]
70                     if lowerLeftY == lowerRightY {
71                         return 1
72                     }
73                 }
74             }
75
76             return 0
77         } else {
78             var rectangleCount = 0
79
80             if direction == UP {
81                 if let tableAtX = coordsTable["x"], let relevantCoordinates = tableAtX[x1] {
82                     for coord2 in relevantCoordinates {
83                         let y2 = coord2[1]
84
85                         let isAbove = y2 > y1
86
87                         if isAbove {
88                             rectangleCount += clockwiseCountRectangles(coord2, coordsTable, RIGHT, lowerLeftY)
89                         }
90                     }
91                 }
92             } else if direction == RIGHT {
93                 if let tableAtY = coordsTable["y"], let relevantCoordinates = tableAtY[y1] {
94                     for coord2 in relevantCoordinates {
95                         let x2 = coord2[0]
96
97                         let isRight = x2 > x1
98
99                         if isRight {
100                             rectangleCount += clockwiseCountRectangles(coord2, coordsTable, DOWN, lowerLeftY)
101                         }
102                     }
103                 }
104             }
105
106             return rectangleCount
107         }
108     }
109 }
110
```

