We read pairs from the file until the first blank line is found. The format of a line is 2 numbers separated by a space.

Algorithm:

We create a matrix, where each line represents the x coordinate and then the y coordinates are added on the line corresponding to the x coordinate in the read pair.

After that, we go through all the possible 2 line combinations in the matrix. We skip over lines that only have one y-coordinate in them, they can't form a rectangle.

The intersection of the 2 lines represents the y-coordinates they have in common, this means we can pick any 2 y-coordinates from the intersection and we can form a rectangle.

Mathematically: we calculate 2-combinations of n (n is the number of elements in the intersection), this represents the number of rectangles that can be drawn between 2 lines.

In the end we just need to add the results we got for each 2 line combination.

EXAMPLE:

1. Input: (1,1), (1,3), (2,1), (2,3), (3,1), (3,3)

   Coordinates[x][y] is matrix
   Coordinates[1] = [1, 3]
   Coordinates[2] = [1, 3]
   Coordinates[3] = [1, 3]

   noRegtangles = 0
   X = 1 and X = 2 -> intesction for Y: [1,3] = 2 (>= 2 TRUE) -> noRegtangles += 1 = 1
   X = 1 and X = 3 -> intesction for Y: [1,3] = 2 (>= 2 TRUE) -> noRegtangles += 1 = 2
   X = 2 and X = 3 -> intesction for Y: [1,3] = 2 (>= 2 TRUE) -> noRegtangles += 1 = 3

   Output: 3

2. Input: (1,1), (1,3), (2,1), (3,1), (3,3)

   Coordinates[x][y] is matrix
   Coordinates[1] = [1, 3]
   Coordinates[2] = [1]
   Coordinates[3] = [1, 3]

   noRegtangles = 0
   X = 1 and X = 2 -> intesction for Y: [1] = 1 (>= 2 FALSE)
   X = 1 and X = 3 -> intesction for Y: [1,3] = 2 (>= 2 TRUE) -> noRegtangles += 1 = 1
   X = 2 and X = 3 -> intesction for Y: [1] = 1 (>= 2 FALSE)

   Output: 1

3. Input: (3,10), (3,8), (3,6), (3,4), (3,0), (6,0), (6,4), (6,8), (6,10)

   Coordinates[x][y] is matrix

   Coordinates[3] = [10,8,6,4,0]
   Coordinates[6] = [0,4,8,10]

   noRegtangles = 0

X = 0 and X = 1 -> intesction for Y: []

....

X = 3 and X = 6 -> intesction for Y: [10, 8, 4, 0] = 4 (>= 2 TRUE) -> noRegtangles += 4! / 2! (4-2)! = 6

The pairs that can be created are:

- 10-8: 3 10 and 6 10 with 3 8 and 6 8
- 10-4: 3 10 and 6 10 with 3 4 and 6 4
- 10-0: 3 10 and 6 10 with 3 0 and 6 0
- 8-4: 3 8 and 6 8 with 3 4 and 6 4
- 8-0: 3 8 and 6 8 with 3 0 and 6 0
- 4-0: 3 4 and 6 4 with 3 0 and 6 0

....

Output: 6