Maths : Video - 43

Leetcode - 3047

Medium

→ codestorywithmik

X → CSwithMIK

→ codestorywithMIK

WeekendWithMIK

Weekend WITH MIK

WeekendWithMIK

@WeekendWithMIK [redacted]

Welcome to WeekendWithMIK 🎥✨ >

[redacted]

MiK.

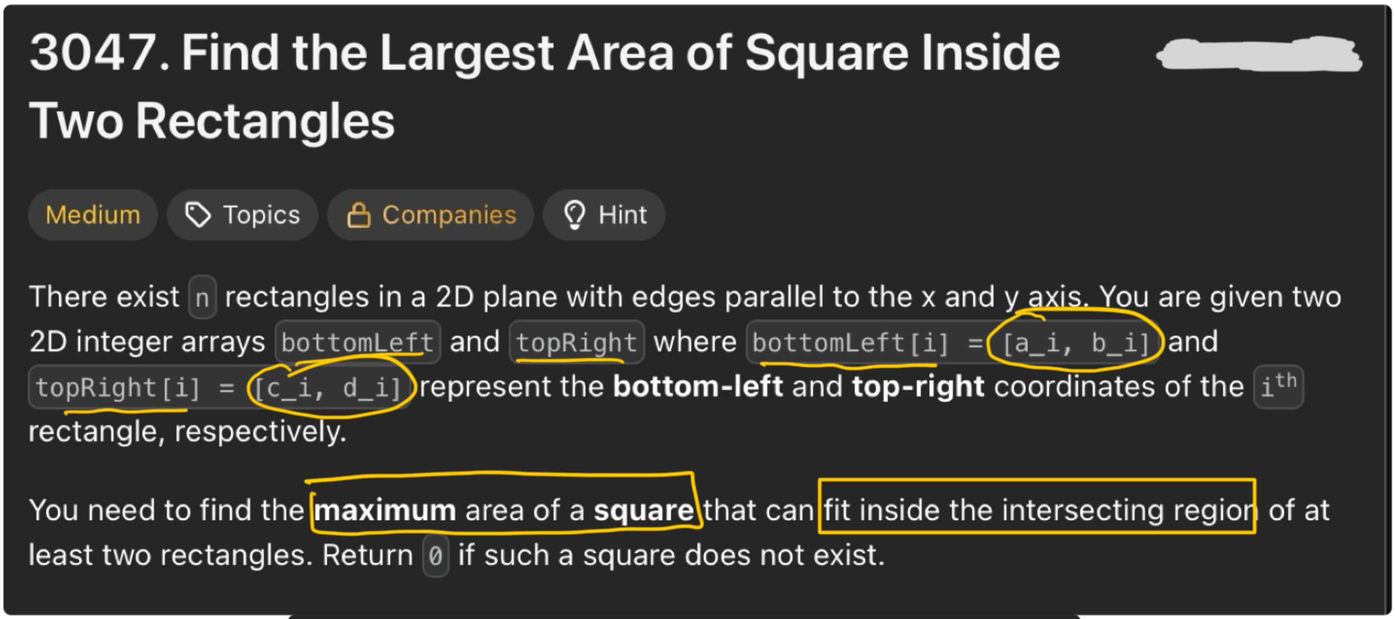
Try this channel to

see "Life behind the Scenes + Tech News

Motivation -

Don't waste another year.
Make this an impactful one.
You've to do it to get your
dream life.
Please don't waste time...

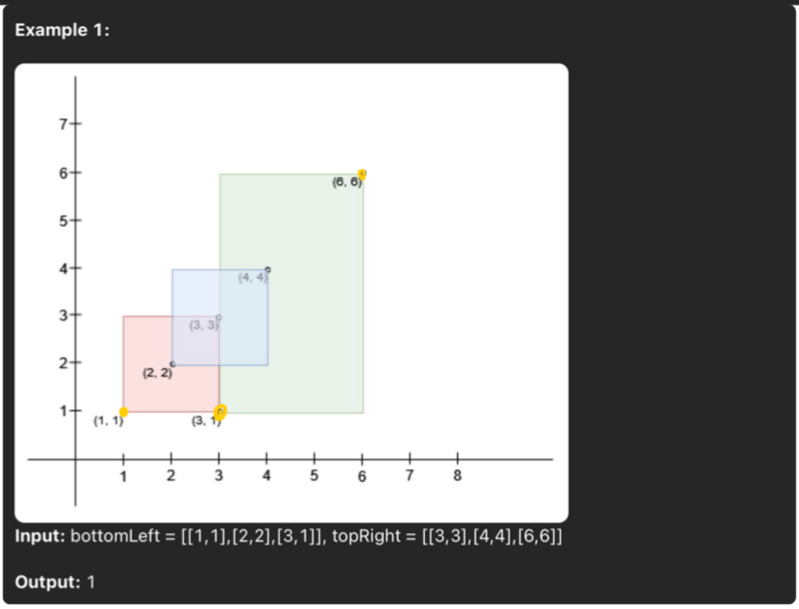
MIK...



3047. Find the Largest Area of Square Inside Two Rectangles [redacted]

There exist `n` rectangles in a 2D plane with edges parallel to the x and y axis. You are given two 2D integer arrays `bottomLeft` and `topRight` where `bottomLeft[i] = [a_i, b_i]` and `topRight[i] = [c_i, d_i]` represent the **bottom-left** and **top-right** coordinates of the i^{th} rectangle, respectively.

You need to find the **maximum** area of a **square** that can fit inside the intersecting region of at least two rectangles. Return `0` if such a square does not exist.



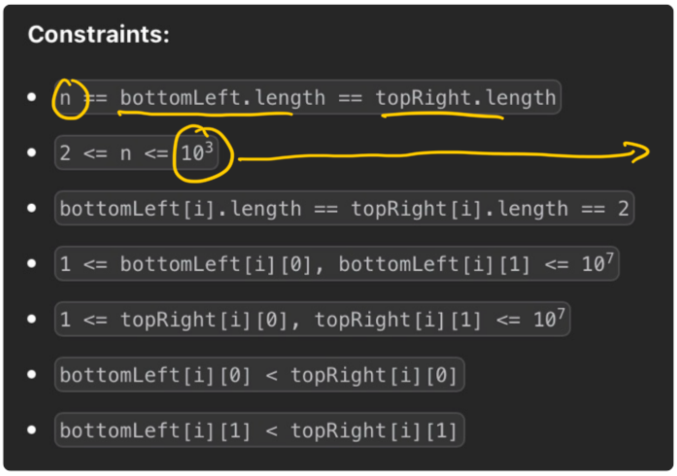
Example 1:

Input: bottomLeft = [[1,1],[2,2],[3,1]], topRight = [[3,3],[4,4],[6,6]]

Output: 1

side = 1

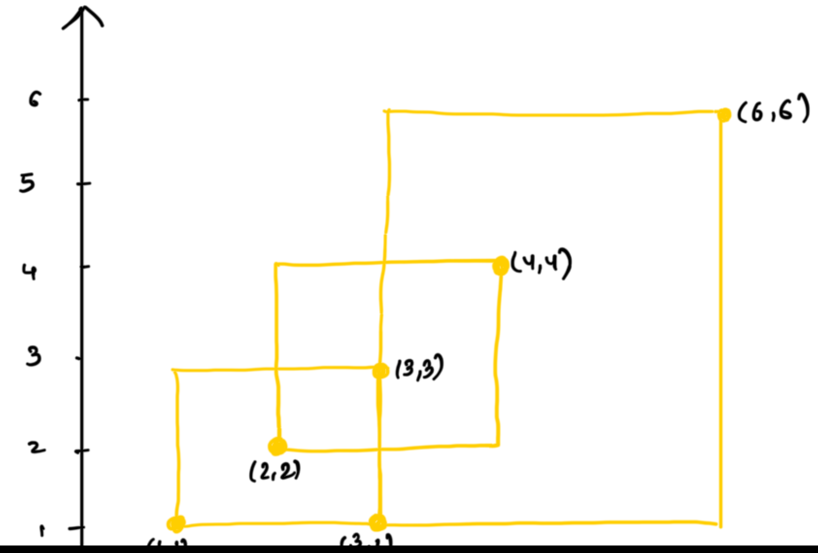
Constraint Analysis :

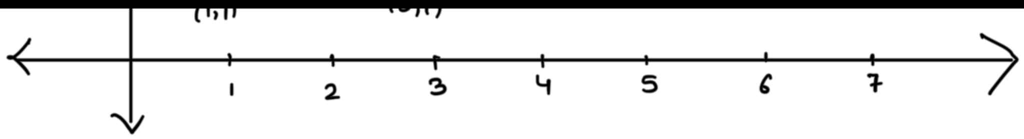


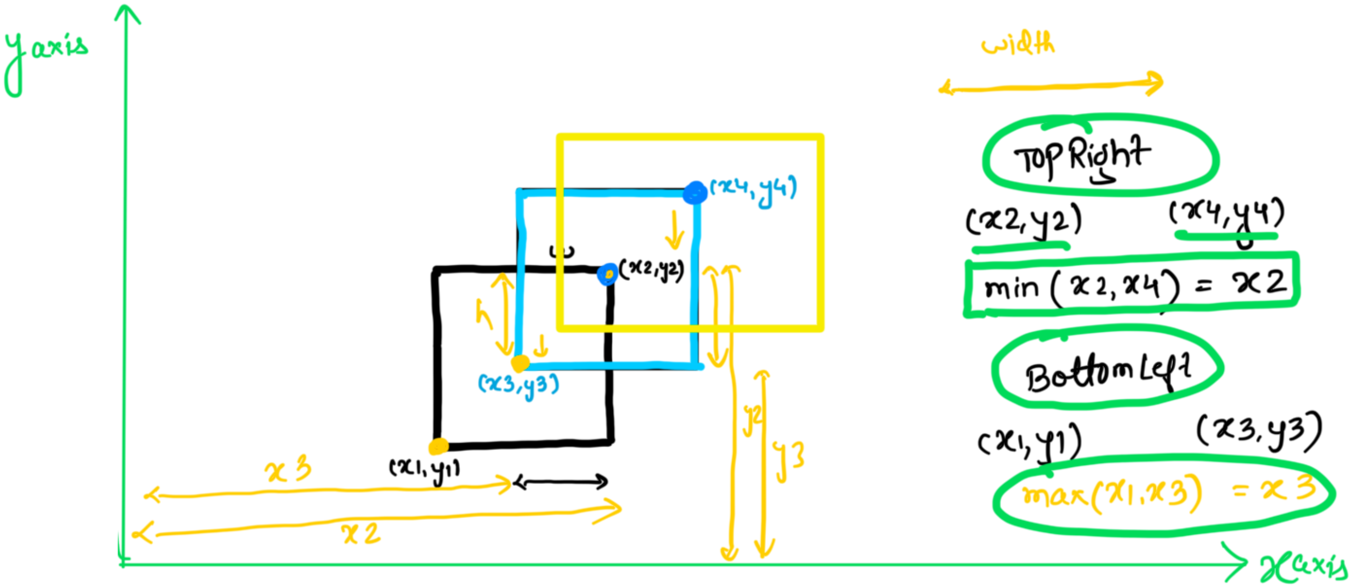
Constraints:

- `n == bottomLeft.length == topRight.length`
- `2 <= n <= 10`^{3}
- `bottomLeft[i].length == topRight[i].length == 2`
- `1 <= bottomLeft[i][0], bottomLeft[i][1] <= 10`^{7}
- `1 <= topRight[i][0], topRight[i][1] <= 10`^{7}
- `bottomLeft[i][0] < topRight[i][0]`
- `bottomLeft[i][1] < topRight[i][1]`

Thought Process







Top Right

(x2,y2) (x4,y4)

$\boxed{\min(x2, x4) = x2}$

Bottom Left

(x1,y1) (x3,y3)

$\max(x1, x3) = x3$

$$\text{width} = x2 - x3$$

yaxis

height

Top_right

(x2, y2) (x4, y4)

$\boxed{\min(y2, y4) = y2}$

Bottom left

(x_1, y_1) . (x_3, y_3)

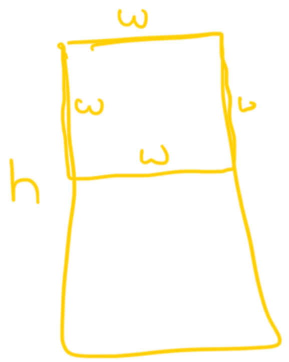
$$\boxed{\max(y_1, y_3) = y_3}$$

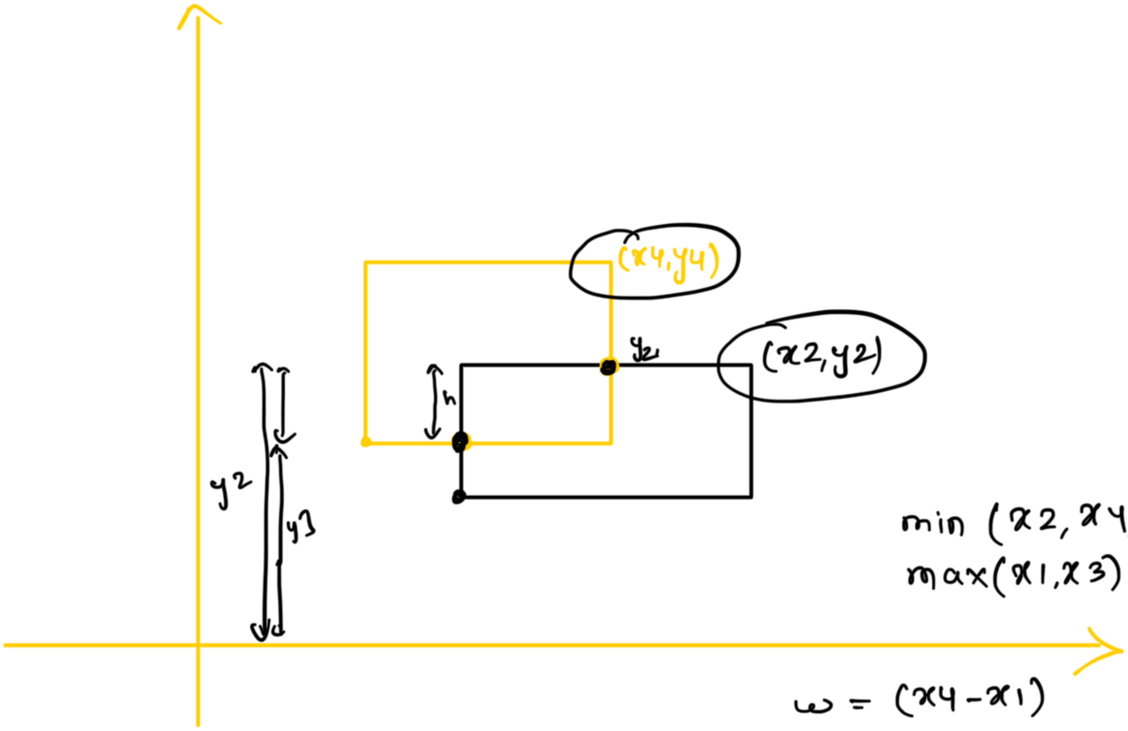
$$h = y_2 - y_3$$

$w, h \longrightarrow \min(w, h) = side$

aree = (side * sid);

2





$\min(x_2, x_4) = x_4$

$\max(x_1, x_3) = x_1$

$w = (x_4 - x_1)$

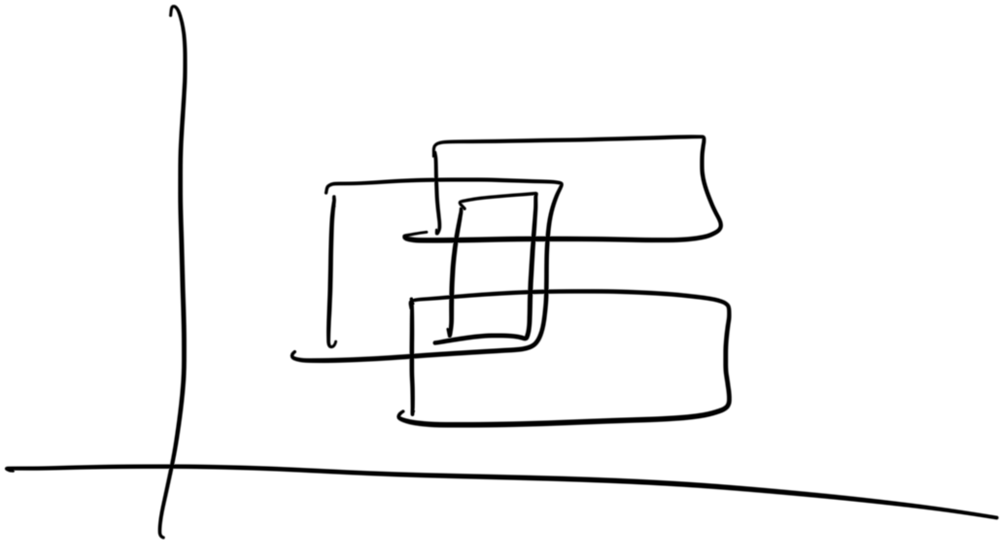
$\min(y_2, y_4) = y_2$

$\max(y_1, y_3) = y_3$

$h = y_2 - y_3;$

x = j - y

All intersections among all rectangles:-



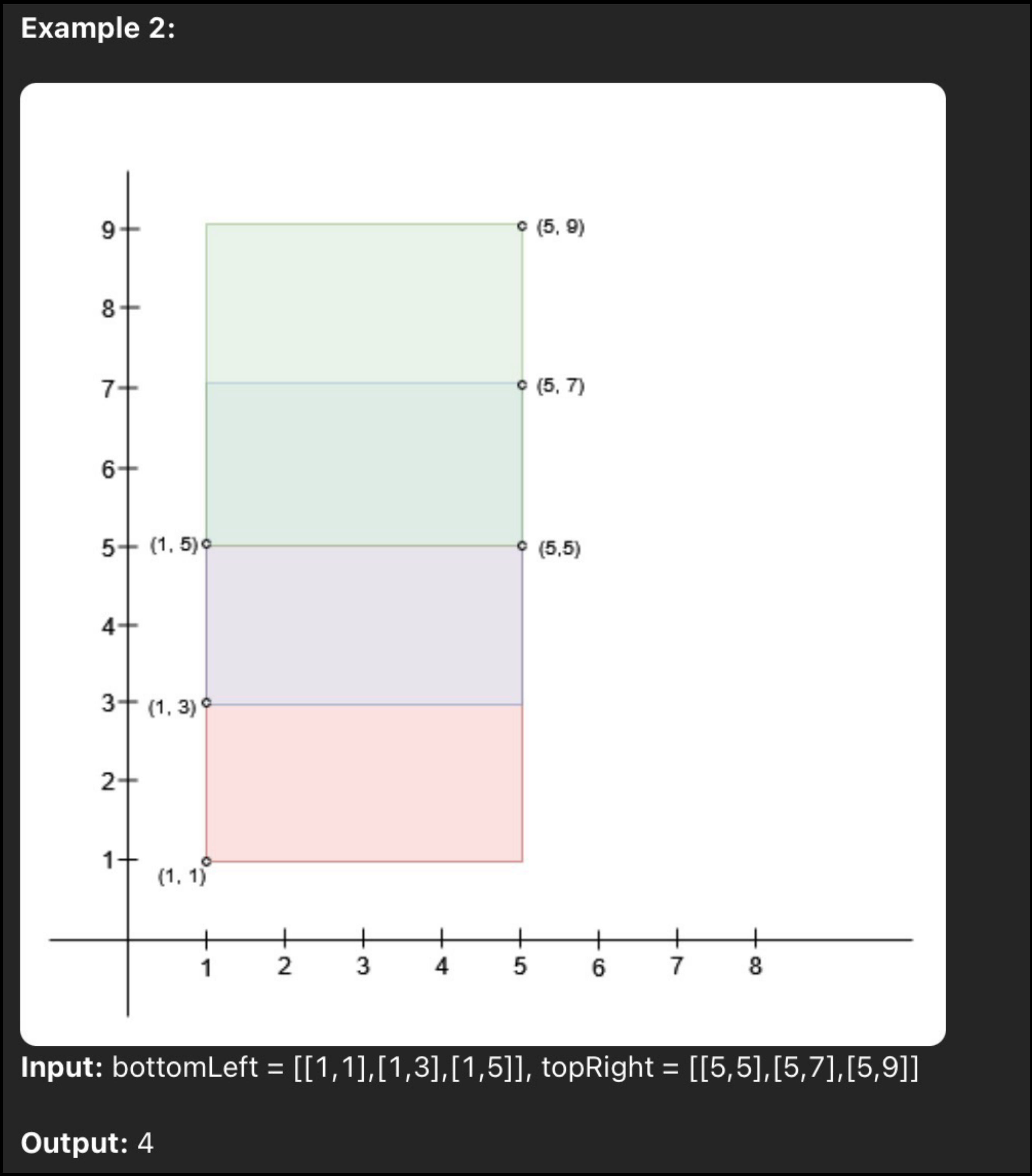
{(x1, y1), (x3, y3) }   {(x2, y2) (x4, y4)}

```
maxSide = 0;
for (i = 0; i < n; i++) {
    for( j = i+1; j<n ; j++) {
            // topright  } x   -> width . (w)
            // Bottomleft } x

            // topRi    } y   -> heigh  (h)
            // Bottlef' } y
            side = min(h,w);
            maxside = max(side, maxSide);
    }
}
```

return (maxSi or max S).

ε



Example 2:

Input: bottomLeft = [[1,1],[1,3],[1,5]], topRight = [[5,5],[5,7],[5,9]]

Output: 4