Given a list of **non-overlapping** axis-aligned rectangles rects, write a function pick which randomly and uniformily picks an **integer point** in the space covered by the rectangles.

Note:

1. An **integer point** is a point that has integer coordinates.
2. A point on the perimeter of a rectangle is **included** in the space covered by the rectangles.
3. ith rectangle = rects[i] = [x1,y1,x2,y2], where [x1, y1] are the integer coordinates of the bottom-left corner, and [x2, y2] are the integer coordinates of the top-right corner.
4. length and width of each rectangle does not exceed 2000.
5. 1 <= rects.length <= 100
6. pick return a point as an array of integer coordinates [p\_x, p\_y]
7. pick is called at most 10000 times.

**Example 1:**

**Input:**

["Solution","pick","pick","pick"]

[[[[1,1,5,5]]],[],[],[]]

**Output:**

[null,[4,1],[4,1],[3,3]]

**Example 2:**

**Input:**

["Solution","pick","pick","pick","pick","pick"]

[[[[-2,-2,-1,-1],[1,0,3,0]]],[],[],[],[],[]]

**Output:**

[null,[-1,-2],[2,0],[-2,-1],[3,0],[-2,-2]]

**Explanation of Input Syntax:**

The input is two lists: the subroutines called and their arguments. Solution's constructor has one argument, the array of rectangles rects. pick has no arguments. Arguments are always wrapped with a list, even if there aren't any.