

游戏中弱角色的数量

你正在参加一个多角色游戏，每个角色都有两个主要属性：**攻击** 和 **防御**。给你一个二维整数数组 `properties`，其中 `properties[i] = [attacki, defensei]` 表示游戏中第 i 个角色的属性。

如果存在一个其他角色的攻击和防御等级 **都严格高于** 该角色的攻击和防御等级，则认为该角色为 **弱角色**。更正式地，如果认为角色 i **弱于** 存在的另一个角色 j ，那么 $attack_j > attack_i$ 且 $defense_j > defense_i$ 。

返回 **弱角色** 的数量。

示例 1：

输入： `properties = [[5,5],[6,3],[3,6]]`

输出： `0`

解释： 不存在攻击和防御都严格高于其他角色的角色。

示例 2：

输入： `properties = [[2,2],[3,3]]`

输出： `1`

解释： 第一个角色是弱角色，因为第二个角色的攻击和防御严格大于该角色。

示例 3：

输入： `properties = [[1,5],[10,4],[4,3]]`

输出： `1`

解释： 第三个角色是弱角色，因为第二个角色的攻击和防御严格大于该角色。

```
class Solution:
    def numberOfWeakCharacters(self, properties: List[List[int]]) -> int:
        #按照第一个属性 从大到小排序
        properties.sort(key=lambda x: (-x[0], x[1]))
        #遍历第二个属性 判断是否前面大于后面 如果是则存在弱角色
        maxdef=0
        count=0 #统计弱角色
        for _, defences in properties:
            if maxdef>defences:
                count+=1
            else:
```

```
        maxdef=defences  
    return count
```