

435-无重叠区间

题述

435. 无重叠区间

难度 中等 626 收藏 分享 切换为英文 接收动态 反馈

给定一个区间的集合 `intervals`，其中 `intervals[i] = [starti, endi]`。返回 需要移除区间的最小数量，使剩余区间互不重叠。

示例 1:

输入: `intervals = [[1,2],[2,3],[3,4],[1,3]]`

输出: 1

解释: 移除 `[1,3]` 后，剩下的区间没有重叠。

示例 2:

输入: `intervals = [[1,2], [1,2], [1,2]]`

输出: 2

解释: 你需要移除两个 `[1,2]` 来使剩下的区间没有重叠。

示例 3:

输入: `intervals = [[1,2], [2,3]]`

输出: 0

解释: 你不需要移除任何区间，因为它们已经是无重叠的了。

提示:

- `1 <= intervals.length <= 105`
- `intervals[i].length == 2`
- `-5 * 104 <= starti < endi <= 5 * 104`

通过次数 124,439 | 提交次数 245,582

思路

冥冥之中感觉要排序，但是究竟是按照右边界排序，还是按照左边界排？

按照右边界排序，就要从左向右遍历，因为右边界越小越好，只要右边界越小，留给下一个区间的空间就越大，所以从左向右遍历，优先选右边界小的。

按照左边界排序，就要从右向左遍历，因为左边界数值越大越好（越靠右），这样就给前一个区间的空间就越大，所以可以从右向左遍历。

题目只是要求移除区间的个数，没有必要去真实的模拟删除区间

按照右边界排序，从左向右记录非交叉区间的个数。最后用区间总数减去非交叉区间的个数就是需要移除的区间个数了。

此时问题就是要求非交叉区间的最大个数。

右边界排序之后，局部最优：优先选右边界小的区间，所以从左向右遍历，留给下一个区间的空间大一些，从而尽量避免交叉。全局最优：选取最多的非交叉区间。

局部最优推出全局最优

题解

C++

```
class Solution {
public:
    //排序规则：按照区间右边界进行排序
    static bool cmp(const vector<int>& a, const vector<int>& b)
    {
        return a[1] < b[1];
    }

    int eraseOverlapIntervals(vector<vector<int>>& intervals)
    {
        if(intervals.size() == 0)
        {
            return 0;
        }

        sort(intervals.begin(), intervals.end(), cmp);
        int count = 1;    //非交叉区间个数
        int end = intervals[0][1]; //记录区间的分隔点
        for(int i = 1; i < intervals.size(); i++)
        {
            if(end <= intervals[i][0])
            {
                end = intervals[i][1];
                count++;
            }
        }
        return intervals.size() - count;
    }
};
```

Python

```
class Solution:
    def eraseOverlapIntervals(self, intervals: List[List[int]]) -> int:
        if len(intervals) == 0:
            return 0
        intervals.sort(key = lambda x: x[1])
        count = 1    #非交叉区间个数
        end = intervals[0][1]    #区间分割点
        for i in range(1, len(intervals)):
            if end <= intervals[i][0]:
                count += 1
                end = intervals[i][1]
        return len(intervals) - count
```

题目描述

评论 (593)

题解 (1.1k)

提交记录

执行结果: **通过** [显示详情 >](#)

[添加备注](#)

执行用时: **220 ms**, 在所有 Python3 提交中击败了 **60.01%** 的用户

内存消耗: **39 MB**, 在所有 Python3 提交中击败了 **33.54%** 的用户

通过测试用例: **58 / 58**

炫耀一下:



[写题解, 分享我的解题思路](#)

提交结果	执行用时	内存消耗	语言	提交时间	备注
通过	220 ms	39 MB	Python3	2022/03/09 11:03	▶
通过	396 ms	87.7 MB	C++	2022/03/09 10:58	▶

思考

总结如下难点:

- 难点一: 一看题就有感觉需要排序, 但究竟怎么排序, 按左边界排还是右边界排。
- 难点二: 排完序之后如何遍历, 如果没有分析好遍历顺序, 那么排序就没有意义了。
- 难点三: 直接求重复的区间是复杂的, 转而求最大非重复区间个数。
- 难点四: 求最大非重复区间个数时, 需要一个分割点来做标记。