5899. Two Best Non-Overlapping Events

My Submissions (/contest/biweekly-contest-64/problems/two-best-non-overlapping-events/submissions/)

Back to Contest (/contest/biweekly-contest-64/)

You are given a **0-indexed** 2D integer array of `events` where `events[i] = [startTime`$_i$`, endTime`$_i$`, value`$_i$`]`. The i^{th} event starts at `startTime`$_i$ and ends at `endTime`$_i$, and if you attend this event, you will receive a value of `value`$_i$. You can choose **at most two non-overlapping** events to attend such that the sum of their values is **maximized**.

Return *this* ***maximum*** *sum.*

Note that the start time and end time is **inclusive**: that is, you cannot attend two events where one of them starts and the other ends at the same time. More specifically, if you attend an event with end time `t`, the next event must start at or after `t + 1`.

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

Example 1:

Time	1	2	3	4	5
Event 0	2				
Event 1				2	
Event 2		3			

```
Input: events = [[1,3,2],[4,5,2],[2,4,3]]
Output: 4
Explanation: Choose the green events, 0 and 1 for a sum of 2 + 2 = 4.
```

Example 2:

Time	1	2	3	4	5
Event 0	2				
Event 1				2	
Event 2	5				

```
Input: events = [[1,3,2],[4,5,2],[1,5,5]]
Output: 5
Explanation: Choose event 2 for a sum of 5.
```

Example 3:

Time	1	2	3	4	5	6
Event 0	3					
Event 1	1					
Event 2					5	

```
Input: events = [[1,5,3],[1,5,1],[6,6,5]]
Output: 8
Explanation: Choose events 0 and 2 for a sum of 3 + 5 = 8.
```

Constraints:

- `2 <= events.length <= 10`^{5}
- `events[i].length == 3`
- `1 <= startTime`$_i$ `<= endTime`$_i$ `<= 10`^{9}
- `1 <= value`$_i$ `<= 10`^{6}

JavaScript

```
const maxTwoEvents = (a) => {
    let d = [], res = 0, maxPreSum = 0;
    for (const [start, end, sum] of a) {
        d.push([start, sum, 1]);
        d.push([end + 1, sum, -1])
```

```
    }
    d.sort((x, y) => {
        if (x[0] != y[0]) return x[0] - y[0];
        return x[2] - y[2];
    });
    for (const [, sum, flag] of d) {
        if (flag == -1) {
            maxPreSum = Math.max(maxPreSum, sum);
        } else {
            res = Math.max(res, maxPreSum + sum);
        }
    }
    return res;
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/579525544/) | More Details > (/submissions/detail/579525544/)

Share your acceptance!