1288. Remove Covered Intervals

https://leetcode.com/problems/remove-covered-intervals/

February 2, 2022

-Priyanshu Arya

1288. Remove Covered Intervals

Medium 1510 40 Add to List Share

Given an array `intervals` where `intervals[i] = [li, ri]` represent the interval `[li, ri)`, remove all intervals that are covered by another interval in the list.

The interval `[a, b)` is covered by the interval `[c, d)` if and only if `c <= a` and `b <= d`.

Return *the number of remaining intervals.*

Example 1:

```
Input: intervals = [[1,4],[3,6],[2,8]]
Output: 2
Explanation: Interval [3,6] is covered by [2,8], therefore it is removed.
```

Example 2:

```
Input: intervals = [[1,4],[2,3]]
Output: 1
```

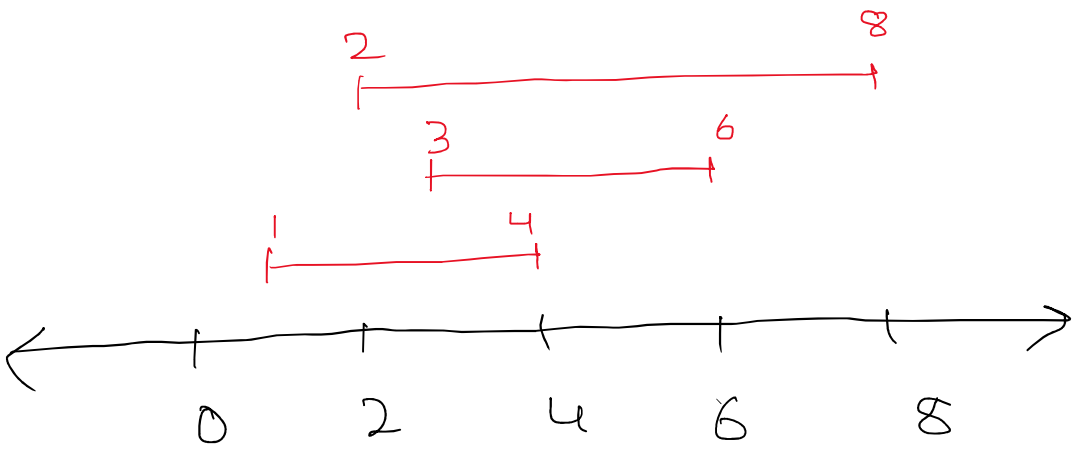
Constraints:

- `1 <= intervals.length <= 1000`
- `intervals[i].length == 2`
- `0 <= li <= ri <= 10^5`
- All the given intervals are **unique**.

Accepted 77,614 Submissions 134,524

Example 1:

```
Input: intervals = [[1,4],[3,6],[2,8]]
Output: 2
Explanation: Interval [3,6] is covered by [2,8], therefore it is removed.
```



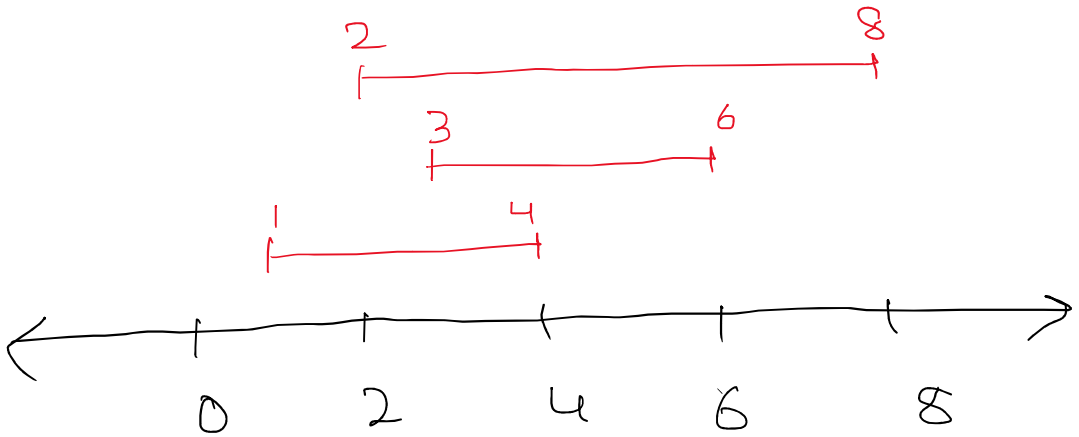
Return: No. of Intervals which are not covered by any other Intervals

Element is covered when two intervals let's say

[a, b] and [c, d]

are like

covered $c <= a$ and $b <= d$



In this Example

→ [1,4] is not covered by any other interval

→ [3,6] is covered by [2,8]

→ [2,8] is not covered by any other interval

[3,6] [2,8]

[a,b] [c,d]

$c <= a$ and $b <= d$

covered element

Bruteforce Approach-1

Time complexity : $O(n^2)$
Space complexity : $O(1)$

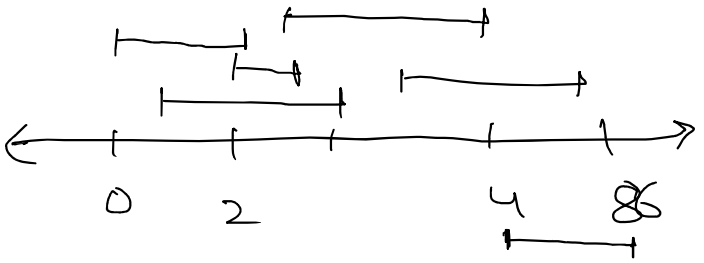
Algorithm:

1- Take an interval
2- match it from all others interval one by one
   i- if interval covered from any other interval then simply continue
   ii - else count in your ans
3- return ans

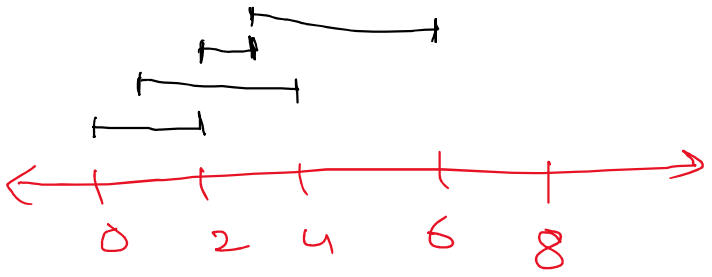
Approach - 2 (Sorting)

i) If we sort our intervals in increasing order so no before element can cover after element.

for Ex [[1,4] [2,3] [0,2] [5,7] [3,6]]



: ← Before Sorting



: ← After Sorting

If we get some intervals of same starting point but different ending point.

For Ex [[2,4] [2,3] [2,6] [2,9]]

then what?

So in that case which interval must be included?

If you see the largest ending point will cover all intervals in it

So include the interval which have largest ending point.

Algorithm for Approach(2)

1- Sort all intervals in increasing order by first element and Sort all intervals in decreasing order if more than one element have same first element.

For Ex: [[3, 4] [4, 6] [2, 5] [2, 3]]

Output: [[2, 5] [2, 3] [3, 4] [4, 6]]

2- Iterate through all intervals and check if interval get covered then continue else add in our answer

3- return ans

Psuedo Code

```
remove covered intervals(nums):
    nums.sort(first element in increasing order(High priority))
    nums.sort(second element if more than one element have
                                        same first element)
    res = [nums[0]]
    for s,e in nums[1:]:
        prevs, preve = res[-1]
        if prevs <= s and preve >= e:
                continue
        else
                res.append([s,e])
    return len(res)
```

Time Complexity:

$O(n \log n)$ ← Sorting

\+ $O(n)$ ← for iterating through all intervals

$O(n \log n)$ ← Time Complexity

Space Complexity: $O(1)$ ← Constant Space

Python3 · Autocomplete

```python
class Solution:
    def removeCoveredIntervals(self, intervals: List[List[int]]) -> int:
        intervals.sort(key=lambda i: (i[0], -i[1]))
        res = [intervals[0]]

        for s, e in intervals[1:]:
            prevs, preve = res[-1]

            if prevs <= s and preve >= e:
                continue
            res.append([s, e])

        return len(res)
```

sorting first element in high priority then second element in intervals

'-1' sign refers sorting in decreasing order and writing i[1] after i[0] is showing the less priority for sorting.

Testcase | Run Code Result | Debugger

Accepted Runtime: 41 ms

Your input
```
[[1,4],[3,6],[2,8]]
[[1,4],[2,3]]
```

Output
```
2
1
```

Expected
```
2
1
```

```
C:\Users\Priyanshu\AppData\Local\Programs\Python\Python310\python.exe "R:/DSA/Leetcode 1/Intervals/1288. Remove Covered Intervals/1288. Remove Covered Intervals.py"
Before sorting [[3, 4], [4, 6], [2, 5], [2, 3]]
After sorting [[2, 5], [2, 3], [3, 4], [4, 6]]
Previous Interval [2, 5] and Current Interval [2, 3]
Previous Interval [2, 5] and Current Interval [3, 4]
Previous Interval [2, 5] and Current Interval [4, 6]
Adding interval [4, 6] in result
Final Result List is [[2, 5], [4, 6]]
2

Process finished with exit code 0
```

```
C:\Users\Priyanshu\AppData\Local\Programs\Python\Python310\python.exe "R:/DSA/Leetcode 1/Intervals/1288. Remove Covered Intervals/1288. Remove Covered Intervals.py"
Before sorting [[3, 4], [4, 6], [2, 5], [2, 3], [1, 2], [3, 6], [5, 6]]
After sorting [[1, 2], [2, 5], [2, 3], [3, 6], [3, 4], [4, 6], [5, 6]]
Previous Interval [1, 2] and Current Interval [2, 5]
Adding interval [2, 5] in result
Previous Interval [2, 5] and Current Interval [2, 3]
Previous Interval [2, 5] and Current Interval [3, 6]
Adding interval [3, 6] in result
Previous Interval [3, 6] and Current Interval [3, 4]
Previous Interval [3, 6] and Current Interval [4, 6]
Previous Interval [3, 6] and Current Interval [5, 6]
Final Result List is [[1, 2], [2, 5], [3, 6]]
3
```

Thank you

If you like Please share this and feel free to connect for any queries.

Discord: https://discord.gg/qPer56TP

Mail: priyanshuarya2482000@gmail.com