

# 056. Merge Intervals

## 056 Merge Intervals

- Sort+array

### Description

Given a collection of intervals, merge all overlapping intervals.

For example,

Given `[1,3], [2,6], [8,10], [15,18]`,

return `[1,6], [8,10], [15,18]`.

### 1. Thought line

### 2. Greedy+array

```
1 /**
2  * Definition for an interval.
3  * struct Interval {
4  *     int start;
5  *     int end;
6  *     Interval() : start(0), end(0) {}
7  *     Interval(int s, int e) : start(s), end(e) {}
8  * };
9  */
10 class Solution {
11 public:
12     vector<Interval> merge(vector<Interval>& intervals) {
13         vector<Interval> result;
14
15         sort(intervals.begin(), intervals.end(), [](const Interval & a, const Interval & b){
16             return a.start < b.start;
17         });
18
19         for (int i=0; !intervals.empty() && i<=intervals.size()-1; ++i){
20             Interval temp = intervals[i];
21             while (i+1<=intervals.size()-1 && intervals[i+1].start<=temp.end){
22                 temp.end = temp.end>intervals[i+1].end?temp.end:intervals[i+1].end;
23                 ++i;
24             }
25             result.push_back(temp);
26         }
27         return result;
28     }
29 };
```