一、题目说明

题目是56. Merge Intervals，给定一列区间的集合，归并重叠区域。

二、我的做法

这个题目不难，先对intervals排序，然后取下一个集合，如果cur[0]>resLast[1]在直接放到集合中，否者合并。代码如下：

```
#include<iostream>
#include<vector>
#include<algorithm>
using namespace std;
class Solution{
    public:
        vector<vector<int>> merge(vector<vector<int>>& intervals){
            int len = intervals.size();
            if(len<2) return intervals;

            sort(intervals.begin(),intervals.end());
            vector<vector<int>> res;

            res.push_back(intervals[0]);
            vector<int> cur,resLast;

            for(int i=1;i<len;i++){
                cur = intervals[i];
                resLast = res[res.size()-1];

                if(cur[0]>resLast[1]){
                    res.push_back(cur);
                }else if(cur[0]<=resLast[1] && cur[1]>resLast[1]){
                    res.back() = {resLast[0],cur[1]};
                }
            }

            return res;
        }
};
int main(){
    Solution s;
    vector<vector<int>> m;
    vector<vector<int>> r;


    m = {{1,3},{2,6},{8,10},{15,18}};
    r = s.merge(m);
    for(int i=0;i<r.size();i++){
        for(int j=0;j<r[i].size();j++){
            cout<<r[i][j]<<"->";
        }
        cout<<"\n";
    }

    cout<<"--------"<<"\n";
```

```
    m = {{1,4},{4,5}};
    r = s.merge(m);
    for(int i=0;i<r.size();i++){
        for(int j=0;j<r[i].size();j++){
            cout<<r[i][j]<<"->";
        }
        cout<<"\n";
    }

    m = {{1,4},{0,4}};
    r = s.merge(m);
    for(int i=0;i<r.size();i++){
        for(int j=0;j<r[i].size();j++){
            cout<<r[i][j]<<"->";
        }
        cout<<"\n";
    }

    m = {{1,4},{2,3}};
    r = s.merge(m);
    for(int i=0;i<r.size();i++){
        for(int j=0;j<r[i].size();j++){
            cout<<r[i][j]<<"->";
        }
        cout<<"\n";
    }
    return 0;
}
```

性能如下：

```
Runtime: 24 ms, faster than 48.18% of C++ online submissions for Merge
Intervals.
Memory Usage: 12.5 MB, less than 82.56% of C++ online submissions for Merge
Intervals.
```

三、优化措施

暂时这样，不优化了。