

高级程序设计与实验随堂测试十二

1. (13分) 题目描述:

写一个 C++ 函数，其输入是两个从小到大排好序的整数向量，长度分别为 n_1 和 n_2 ，返回一个新的长度为 n_1+n_2 的整数向量，其内容是将输入的两个向量中的内容合并，元素也按从小到大的顺序排列。

你的函数运行时间应是 $O(n_1+n_2)$ 的，否则最多只得 8 分。

函数格式: `vector<int> merge(const vector<int> & A, const vector<int> & B);`

2. (17分) 定义一个结构体，它叫 `interval`，其成员是两个整型变量: `start`, `end`

这个结构体代表一个区间，`start` 和 `end` 分别表示区间的起始位置和结束位置(`start <= end`)。

再写一个 C++ 函数，它的输入是盛放 N 个 `interval` 类型变量的向量，输入中的区间可能有重叠，你的函数应将所有重叠的区间合并，并返回一个盛放 `interval` 类型的向量，使得输出的这些区间没有重叠，并且刚好覆盖输入中的所有区间。

示例:

- 输入: `intervals = [[1, 3], [8, 10], [2, 6], [15, 18]]`
- 输出: `[[1, 6], [8, 10], [15, 18]]`
- 解释: 区间 `[1, 3]` 和 `[2, 6]` 重叠，将它们合并为 `[1, 6]`。

你的函数运行时间应为 $O(N \log N)$ 的，否则最多得 12 分。

函数格式: `vector<interval> merge(const vector<interval> & A);`

边界情况解释: `[1, 6]`, `[6, 9]` 应当合并为 `[1, 9]`，但是 `[1, 5]`, `[6, 9]` 是两个不重叠的敬意。

3. (Bonus, 10 分) 给定一个字符串 s , 请将 s 分割成一些子串, 使每个子串都是回文串。返回或者直接打印 s 所有可能的分割方案。

示例:

- 输入: $s = \text{"aab"}$
- 输出: $[[\text{"a"}, \text{"a"}, \text{"b"}], [\text{"aa"}, \text{"b"}]]$

既可以提交一个或几个 C++ 函数, 也可以提交整个程序。