

提交方式:

Canvas 上提交

提交内容:

一个压缩包，命名方式：姓拼音_名拼音_学号.zip/rar，如：zhou_zhiming_2020000112.zip（不要用中文）
内含若干个文件夹，每个文件夹对应一道题的项目（仅保留.pro 和其他源文件；文件夹名也不要有中文）
(.cpp / .h 等)

评分标准:

代码正确性和完整性 90%

代码风格 10% （代码看着很乱的酌情扣 0-10 分。唯一的要求：代码需按层次缩进、对齐）

逾期惩罚:

逾期提交的，成绩 $\times= 0.8$ （无论逾期多久）

以下题目均要求：基于递归完成（此外，使用 Collections 时，需采用 Stanford Library 中提供的）

1. 完成以下函数，并在 main 函数中进行测试

- (1) `string reverse(string str)`; 返回字符串 `str` 的逆
- (2) `bool palindrome(string str)`; 返回字符串 `str` 是否回文

2. 逆序对

- (1) Console 输入
 - ① 一个 `Vector<int> vec`
- (2) Console 输出
 - ① 基于归并排序，计算 `vec` 中的逆序对的数量
- (3) 逆序对
 - ① 满足 `vec[j] < vec[i]` 的 `<i, j>` pair

要求基于归并排序思想完成

提示：归并过程中如果右边比左边小，则左边后续所有元素与右边当前元素均构成逆序对

3. 第 k 小数

- (1) Console 输入
 - ① 一个 `Vector<int> vec`
 - ② 一个整数 `k`
- (2) Console 输出
 - ① `vec` 中的第 `k` 小数

要求基于快速排序思想完成

提示：快速排序时，如果左半边的元素个数大于等于 `k`，则右边无需排序；如果小于 `k`，则左边无需排序

4. 二分查找+

- (1) Console 输入
 - ① 一个有序的 `Vector<int> vec`
 - ② 一个整数 `x`
- (2) Console 输出
 - ① `vec` 中小于 `x` 的元素中，下标最大的元素的下标

提示：`vec[mid]` 大于等于 `x`，则在左半边找；如果小于 `x`，则在右半边找，但需同时考虑 `mid` 这个位置有一个