

程序设计基础_带头尾哨兵的双向链表的稳定的插入排序算法

题目要求

在做本次作业前，请充分阅读、理解并调通示例代码 `struct -e`。仿照示例代码以“文件+重定向”方式，完成逐条、逐项读取，并分别存储为链表（含实现动态申请、释放）。

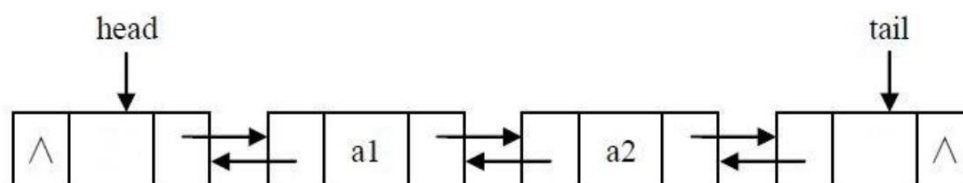
本次作业要求重新实现插入排序算法，要求：

- 1) 将链表改为双向链表；
- 2) 为链表增设头、尾哨兵；
- 3) 借助嵌套方式，实现数据项与链接项的分离；
- 4) 输入时，链表内数据的顺序和文件内数据的顺序一致。
- 5) 使之成为稳定的排序。

[sort.cpp](#)(点击即可下载)中给出了程序框架及使用的结构定义，请在此基础上进行补充，完成上述功能。

附加说明

带头尾哨兵的双向链表是指，链表中每个节点都能指向前一个节点和后一个节点，并且在头尾分别有一个不存储数据的节点，用作“哨兵”，例如下图所示的含有两个数据节点（a1,a2）的链表。哨兵的作用在于，消除插入或删除数据节点时的一些特殊情况，更便于操作。



课上讲过，在此再次强调一次：排序的稳定性是指，在排序前后，具有相同关键字的多条记录间的先后关系能够保持不变。换句话说，任意两条关键字相同的记录，排序前序号较小的，在排序后序号仍较小。

输出要求

每个学生的信息占一行，每个字段用“\t”分隔

实验报告要求

实验报告的内容：

- 1) 解题思路。
- 2) 测试过程，包括排序后双向链表的完整性的测试、排序算法的稳定性测试等。
- 3) 阐述在你的程序中，双向连接和头尾哨兵的作用。
- 4) 提交 word 文件，不要超过 200KB。