实验报告

1.解题思路

本次程序是利用带头尾哨兵的双向链表实现稳定的插入排序。由此先初始化，获得头尾哨兵（互相链接），并返回头哨兵的值，保存尾哨兵的值。然后开始读入文件，读入时先读入一行字符串，再用scanf函数给结构中的各变量赋值，每一次的新结构均插入在前一个结构与尾哨兵之间，使得结构链接顺序与输入相同。接着进行插入排序，思路是从第二项（不算头哨兵）开始遍历前面的数据比较大小，若小则交换大于等于则跳出，等于时跳出保障了稳定性（后处理的数据遇到前面的相同数据不交换）。然后输出数据，最后清理内存空间。

2.测试过程

部分测试代码仍在源文件中未删除，供参考，而部分冗余测试代码已删除。双向链表完整性的测试我是在排序后输出了头哨兵的下一个与尾哨兵的前一个，证明头尾无差错，接着又逆向输出（源文件输出为正向输出）了一遍数据，证明该链表双向均完整。排序算法的稳定性我是通过修改输入的数据测试极端情况得出的，如令所有人的班号都相同，观察输出的结构。

3.头尾哨兵与双向链接的作用

第一个方面是双向链接使得输入时链表数据顺序与文件一致，具体的实现方式见解题思路，这通过上课教授的单向链表不易实现。第二个是头尾哨兵储存地址使得许多操作简化许多，思路也更加清晰，如给函数传参、循环初始条件等。第三个方面是可以实现稳定的插入排序，在我的程序中稳定性实现的基础之一就是顺序读入数据，这是原来的单向链表不易做到的；在进行排序交换的时候，双向的指针使得交换的过程更为复杂，但是这也另其更加灵活自由，给了我们更大的操作空间，可以实现稳定等要求。