

1 链表的头结点，即head变量，里面不存储任何数据，只是为了随时能够找到这个链表并对其进行操作，每个链表节点初始化时候都要讲nxt值设置为NULL，以表示链表的终止。

2 链表节点便是一个struct变量，之前在linklist.h中有声明。

3 malloc函数回去自己查书，不多做阐述

4 这个程序中带有汉字的字符串均使用了wchar_t型字符数组，表示形式为L"***"，输入输出格式化控制符为%ls，-7表示左对齐，留7个空位。

5 输入后点击回车时产生的换行符\n会维持在输入缓冲区内，要及时清除掉，以免不小心读取到造成错误

6 fopen("", "rb"), rb参数为readbinaryfile，用于打开二进制文件（课程要求），打开book.data文件

7 读取时删除旧表，以防把新读取的链接到旧的上。

8 fread函数读取文件内容到临时变量中，并使用insertlist函数将其加到链表尾部。同样新节点的nxt指针要设置为NULL

9 typedef函数直接将结构体变量类型设置为linknode，结构体指针设置为linklist。

10 #ifndef#define#endif等预编译指令防止将一个头文件引用两次，虽然这里没什么用，但是这是写项目的一个习惯。

11 fwrite函数将临时变量的内容写入到文件中，为了写入每一个节点内容，使用

12 while(t->nxt!=NULL)t=t->nxt;来遍历链表所有元素

13 因为链表头结点head中不放任何数据，所以第一个就是t->nxt。

14 _getws(wchar_t *)函数类似于gets，用于读入一行（注意是一行）字符串

15 DispList依旧使用上述循环操作遍历元素。

16 记得查看->运算符的使用方式，自己查书或百度

17 关于选择排序内容，书上会有详细介绍，基本操作就是找到链表中最小的，将其与第一个交换，以此类推

18 使用指针来交换变量内容，书上也有讲，自己查看

19 关于wcscmp()，类似于strcmp()，当两字符串相同时返回0，所以要非运算一下！

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34