

```

1  #include "stdafx.h"
2  #include "linklist.h"
3
4  void init(LinkList);
5  void insert(LinkList);
6  void view(LinkList);
7  void sort(LinkList);
8  void query(LinkList);
9  void stat(LinkList);
10 void del(LinkList);
11 void modify(LinkList);
12 void save(LinkList);
13 void load(LinkList);
14
15 int main()
16 {
17     setlocale(LC_ALL, ""); //设置程序语言为中文（能显示中文字符
18     LinkList head = (LinkList)malloc(sizeof(LinkNode));
19     //建立链表头结点，以后链表都以head来寻找
20     head -> nxt = NULL; //记得设置链表区域为空指针
21     init(head); //初始化程序
22     system("pause");
23     return 0;
24 }
25
26 void init(LinkList t)
27 {
28     int menu; //菜单操作变量
29     printf("\n\t%s\n\n", L"欢迎使用图书信息管理系统！");
30     do{
31         printf(" ----- \n");
32         printf(" |%-7ls|\n", L"1.录入数据");
33         printf(" |%-7ls|\n", L"2.浏览数据");
34         printf(" |%-9ls|\n", L"3.排序");
35         printf(" |%-7ls|\n", L"4.查询数据");
36         printf(" |%-7ls|\n", L"5.统计数据");
37         printf(" |%-7ls|\n", L"6.删除数据");
38         printf(" |%-7ls|\n", L"7.修改数据");
39         printf(" |%-7ls|\n", L"8.保存数据");
40         printf(" |%-7ls|\n", L"9.读入数据");
41         printf(" |%-7ls|\n", L"0.退出程序");
42         printf(" ----- \n");
43         printf("%ls", L"请输入选项(0-9): ");
44         scanf("%d", &menu);
45         fflush(stdin); //清空输入缓冲区
46         switch(menu) //switch选择结构
47         {
48             case 1: insert(t); break;
49             case 2: view(t); break;
50             case 3: sort(t); break;
51             case 4: query(t); break;
52             case 5: stat(t); break;
53             case 6: del(t); break;
54             case 7: modify(t); break;
55             case 8: save(t); break;
56             case 9: load(t); break;
57             case 0: break;
58             default: printf("%ls\n", L"输入错误");
59         }
60     }while(menu); //输入0的话可以退出，否则循环
61 }
62
63 void insert(LinkList t) //添加图书信息
64 {
65     LinkNode r; //直接定义一个结构体变量
66     r.nxt = NULL;
67     printf("%ls", L"图书编号: "); scanf("%d", &r.aid); fflush(stdin);
68     printf("%ls", L"书名: "); _getws(r.name);
69     printf("%ls", L"作者: "); _getws(r.auth);
70     printf("%ls", L"出版社: "); _getws(r.pub);
71     printf("%ls", L"出版时间: "); scanf("%d", &r.time);
72     printf("%ls", L"价格: "); scanf("%lf", &r.price);
73     fflush(stdin);

```

```

74     InsertList(t, &r); //使用插入节点函数添加到链表尾部
75 }
76
77 void view(LinkList t) //显示当前所有图书信息
78 {
79     DispList(t);
80 }
81
82 void sort(LinkList t) //以图书编号为顺序排序
83 {
84     SortList(t);
85     printf("%ls\n", L"排序完成!");
86 }
87
88 void query(LinkList t) //查询图书
89 {
90     int w;
91     printf("%ls\n", L"1.按图书编号");
92     printf("%ls\n", L"2.按作者");
93     scanf("%d", &w);
94     if(w == 1){
95         int id;
96         printf("%ls", L"图书编号: ");
97         scanf("%d", &id);
98         FindListId(t, id); //根据图书编号查找图书
99     }else if(w == 2){
100         wchar_t au[MAXN];
101         printf("%ls", L"作者: ");
102         scanf("%ls", au);
103         FindListAuth(t, au); //根据作者查找图书
104     }else{
105         printf("%ls\n", L"输入错误");
106     }
107     fflush(stdin);
108 }
109
110 void stat(LinkList t) //按照出版社统计图书
111 {
112     wchar_t pu[MAXN];
113     printf("%ls", L"出版社: ");
114     scanf("%ls", pu);
115     fflush(stdin);
116     StatList(t, pu);
117 }
118
119 void del(LinkList t) //删除图书
120 {
121     int id;
122     printf("%ls", L"图书编号: "); scanf("%d", &id);
123     fflush(stdin);
124     DeleteList(t, id);
125     printf("%ls\n", L"删除成功!");
126 }
127
128 void modify(LinkList t) //修改图书
129 {
130     int id;
131     LinkNode r; //定义一个结构体变量
132     printf("%ls", L"图书编号: "); scanf("%d", &id);
133     fflush(stdin);
134     r.nxt = NULL;
135     r.aid = id;
136     printf("%ls", L"书名: "); scanf("%ls", r.name);
137     printf("%ls", L"作者: "); scanf("%ls", r.auth);
138     printf("%ls", L"出版社: "); scanf("%ls", r.pub);
139     printf("%ls", L"出版时间: "); scanf("%d", &r.time);
140     printf("%ls", L"价格: "); scanf("%lf", &r.price);
141     fflush(stdin);
142     ModifyList(t, &r);
143 }
144
145 void save(LinkList t) //保存链表内容到二进制文件
146 {

```

```

147     FILE * fp;
148     if ((fp = fopen("book.data", "wb")) == NULL) {
149         fprintf(stderr, "Cannot open data file.\n");
150         return;
151     }
152     while(t -> nxt != NULL){
153         fwrite(t -> nxt, sizeof(LinkNode), 1, fp);
154         t = t -> nxt;
155     }
156     printf("%ls\n", L"保存成功! ");
157     fclose(fp);
158 }
159
160 void load(LinkList t) //读取二进制文件中链表内容
161 {
162     FILE * fp; //文件指针
163     LinkNode buf; //临时变量
164     if ((fp = fopen("book.data", "rb")) == NULL) {
165         fprintf(stderr, "Cannot open data file.\n");
166         return;
167     }
168     DestroyList(t); //删除旧表
169     while(fread(&buf, sizeof(LinkNode), 1, fp)){
170         buf.nxt = NULL; //防止建表错误
171         InsertList(t, &buf);
172     }
173     printf("%ls\n", L"读取成功! ");
174     fclose(fp);
175 }
176

```