**数据结构设计思想文档报告**

1. **任务题目**

排序综合。

1. **数据类型**

结构体：SqList

包括：int \*型的指针pBase（用来存放序列的首地址）

Int型的cnt（有效元素的数目）

SqList \*类型的L

Bool类型的flag

FILE \*类型的fp（保持文件时用）

Char类型的a数组

还有其他int类型变量

1. **算法思想**

选择排序：

一共比较i（序列长度减一）次，每次都比较第i个和第i+1,i+2…个

起泡排序：

一共比较i（序列长度减一）次，从第一个开始，每次都比较第j个和第j+1个，第一次比较完，确定最后一个是最大的，如此类推。

快速排序：

通过一趟排序将序列分为两部分，一部分整体比另一部分整体小，再通过递归反复对各部分进行相同操作。

1. **N-S流程图**

定义变量L,m,n,I,flag,fp,a[200]。

输出("\n\n");

输出(" 欢迎进入排序综合系统！\n");

输出("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

输出(" 1.创建一个序列\n");

输出(" 2.查看序列\n");

输出(" 3.选择排序\n");

输出(" 4.起泡排序\n");

输出(" 5.快速排序\n");

输出(" 6.退出程序\n");

输出("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

输出("请根据您的需要输入要执行操作的序号：");

输入m的值

M

1 2 3 4 5 6

输出"请输入您 输出"您的序 调用sort\_1 调用sort\_2 调用sort\_3 输出

的序列长度：" 列为：\n 函数 函数 函数 谢谢

输入 n 调用 输出 输出 输出 使用，

调用 traverse函数 "对您的序列 "对您的序列 "对您的序列 return

CreateList函数 进行选择排 进行起泡排 进行快速排 false

输出"此操作结输出"此操作 后结果如下:” 后结果如下:” 后结果如下:”

束，请进行其 结束，请进行 调用 调用 调用

他操作！" 其他操作！” traverse函数 traverse函数 traverse函数

输出"\n\n" 输出"\n\n" 程序1 程序2 程序3

输出"此操作 输出"此操作 输出"此操作

结束，请进行结束，请进行结束，请进行

其他操作！” 其他操作！” 其他操作！”

输出"\n\n" 输出"\n\n" 输出"\n\n"

flag = true;

**图4.1 主函数N-S图**

|  |
| --- |
| (fp=fopen("G:\\选择排序.txt","w+"))==NULL  T F  输出 for (i=0; i<n; i++)  "打开文件失败!" a[i] = Find(&L, i);  fprintf(fp, "%d ", a[i]);  fclose(fp); |

**图4.2 程序1 N-S图**

|  |
| --- |
| (fp=fopen("G:\\起泡排序.txt","w+"))==NULL  T F  输出 for (i=0; i<n; i++)  "打开文件失败!" a[i] = Find(&L, i);  fprintf(fp, "%d ", a[i]);  fclose(fp); |

**图4.3程序2 N-S图**

|  |
| --- |
| (fp=fopen("G:\\快速排序.txt","w+"))==NULL  T F  输出 for (i=0; i<n; i++)  "打开文件失败!" a[i] = Find(&L, i);  fprintf(fp, "%d ", a[i]);  fclose(fp); |

**图4.4 程序3 N-S图**