高级语言课程设计实验报告

**班级学号：2210010150**

**姓名: 杨屹**

**图示

中度可信度描述已自动生成完成时间2023 05 28**

* 课程设计总体要求：

按照题目进行程序设计，变量定义符合实际，程序结构清晰，模块设计合理，链接测试结果正确，有测试图。实验报告书写完整。

* 课程设计题目要求
* 设计的思路

要求：根据题目简述题目设计的思路和方法以及所采用的基本结构

* 实现的方法（附主要模块代码）

按照设计思路详细实现每个模块的实现过程

* 测试结果（需要进行正常和异常测试结果图）

需要对所完成的程序进行测试一般需要进行三次

两次正常参数测试 一次异常测试

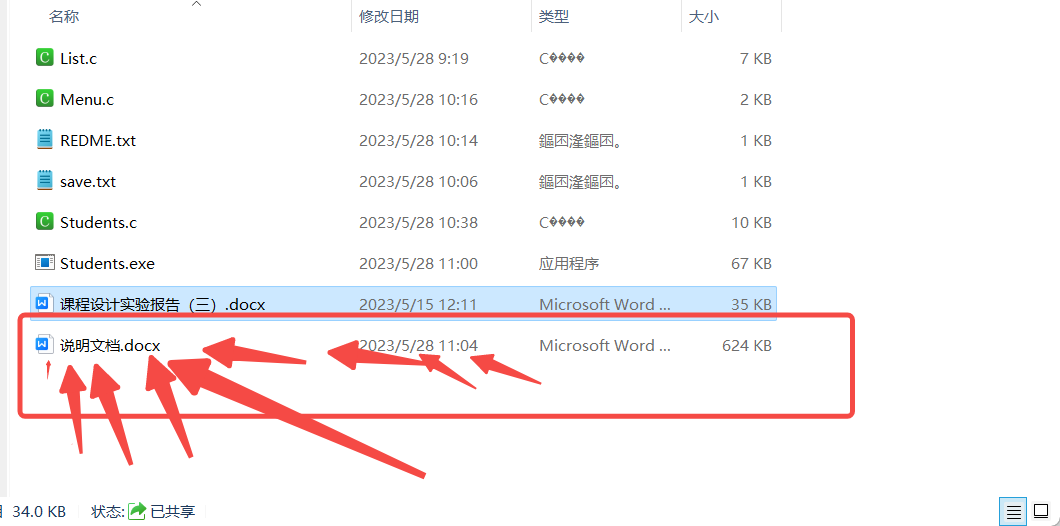
* 不足和需改进的地方

写清楚本次设计的完成情况不足和待改进的地方

* 教师评语
* 附程序代码（同时提交电子版程序）

课程设计三

本实验报告附赠~~精美~~说明文档, 请务必查看



1. **题目：**

设计一个学生管理系统



参照界面完成各个模块的设计

1. **简述设计思想**

变量的定义和含义

输入模块设计

处理模块设计

输出模块设计

文件分工:

list.c 链表的库, 生成链表, 添加节点,等等操作

主要方法:

1. *List*\* mkList()

使用方法举例:

List myList = mkList();

本方法可一步生成一个可用的链表

链表中默认附带一个值全为0的节点.

1. *Node*\* mkNode()

创建一个初始化好的空节点.

使用方法举例:

Node \*myNode = mkNode();

1. // 对已知List 的末尾增加一个 Node

*List*\* Node\_add( *List*\* *LIST* )

使用方法举例:

假设已有一个可用链表LIST, 则可:

Node\_add( LIST );

该链表末尾便会新增一个空节点

1. *List* \*Sort( *List* \**LIST* )

排序, 使用查找排序, 本函数根据 LIST -> score[ 6 ] 的大小进行降序排列

使用方法举例:

假设已有一个可用的, 存完数据的链表 LIST

Sort( LIST );

1. // 传入( 链表LIST, 地址p ), 此函数在 LIST 末尾 追加一个p.

int AppendNode( *List* \**LIST*, *Node* \**p* )

使用方法举例:

在排序中, 遇到了这样的情景:

已知原链表LIST, 新链表newList.

为提升代码效率, 希望采用改变指针等方式, 高效对每组数据( 即每个节点 )

进行排序

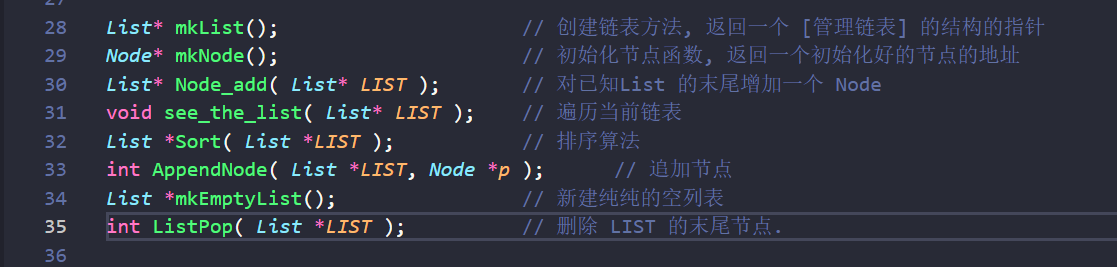
思路如下:

1. 在 LIST 中找到最大值 , 用指针 maxP 指向.

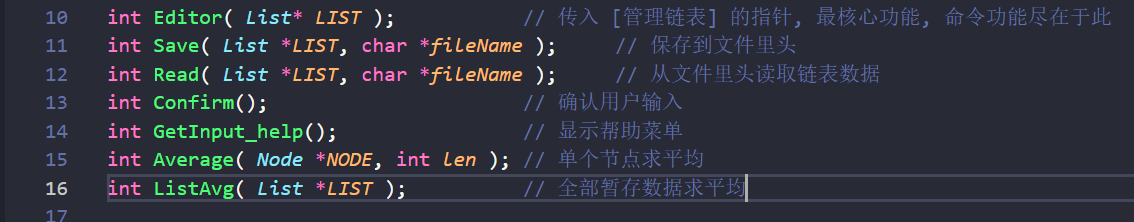
在 newList 的末尾追加 maxP.

1. 实现（主要模块实现）

List.c 中的函数原型



Student.c 中的函数原型



程序实在是有点多, 详见代码吧…

1. 测试结果展示（截图）

图文使用说明见<说明文档>





d

1. 不足和改进

暂时还没开发针对性编辑功能, 难以对数据进行针对性修改,

不过目前可以用记事本打开save.txt进行修改



1. 教师评语及打分
2. 程序代码（附）

