**北京交通大学**

**计算机实验报告**

**实验报告（lab5）**

年 级：大学一年级

学 号：20281274

姓 名：杜海玮

专 业：计算机科学

任课老师：邹琪

二零二一年五月

目 录

[1）程序实现 1](#_Toc4002)

[A) 外部调用实验4程序（system函数） 1](#_Toc28155)

[B）设计文件目录关系 1](#_Toc5567)

[C）实验五配置文件 3](#_Toc8953)

[D) 自检函数设计与相对位置阐述 3](#_Toc18156)

[2） 综述与学习：程序设计文档 4](#_Toc11420)

[A）程序设计文档的要求 4](#_Toc10658)

[B）程序设计文档的目的 5](#_Toc64)

[3） 实验总结 5](#_Toc19334)

[A) 实验总结 5](#_Toc5062)

[B) 对小组其他成员的文档审查报告 6](#_Toc13950)

1）程序实现

A) 外部调用实验4程序（system函数）

1.1函数声明与作用

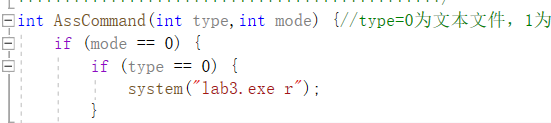


图 1-1 system函数（读取字符串）

System函数默认在当前目录中充当命令行角色，如图，我使用该调用当前目录中的lab4文件，使用random条数据记录。它也接受字符串变量命令，如下图：

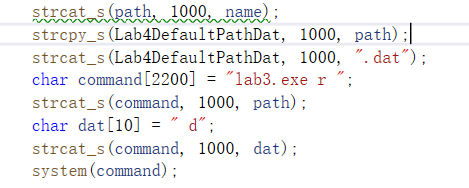


图 1-2 system函数（读取字符串变量）

1. 设计文件目录关系：

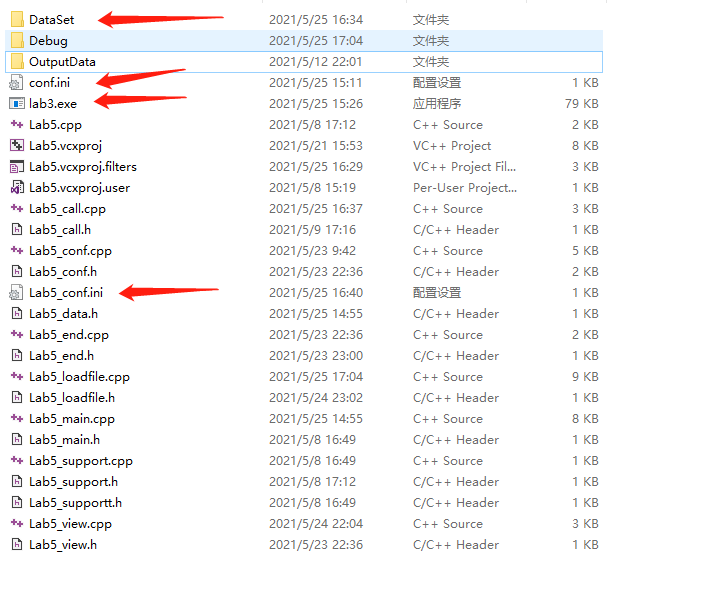


图 1-2 文件目录关系

我将四个重要子文件/目录置于lab5工作目录下，这样，各个程序可以方面的直接调用彼此而不必使用chdir命令。

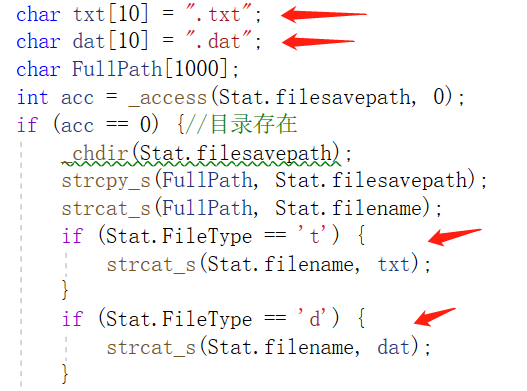


图 1-3 后缀名按用户指定加入

1. 实验五配置文件：

如图1-2，实验五配置文件是Lab5\_conf.ini，实验4配置文件是conf.ini，他们共处lab5程序根目录下，即相对根目录为空，可以直接用system命令调用。方便调用。Lab5\_conf.ini中或1或0，0代表自动模式，1代表手动模式。

1. 自检函数设计与相对位置阐述

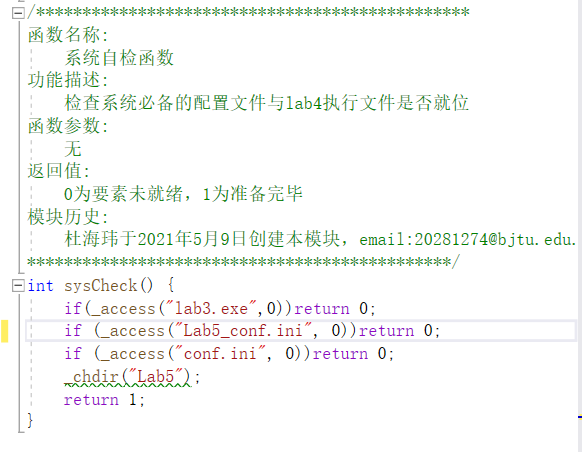


图 1-4 系统自检函数

如图所示，系统自检函数将检查lab4程序是否存在于指定位置，lab4和lab5配置文件是否存在于指定位置。

众所周知文件路径有相对位置与绝对位置两种，而采用相对位置能从当前工作目录出发导向目标目录，在实验5中，我使用了相对目录定位实验4生成的数据记录文件，即DataSet\DataFile.txt（或DataFile.txt），这样，在自动模式下我的读取函数就能准确抓取目标数据文件，配置文件同理也可准确读取。

1. 综述与学习：程序设计文档

A）程序设计文档的要求

程序设计文档包括了程序的设计要求、程序实现目标、程序各子文件细节、程序各函数细节等关于程序各个功能实现的抽象要求与标准。该文档起到了程序开发的指导作用。

B）程序设计文档的目的

我认为其中程序的设计要求、程序实现目标、程序各子文件要求应该在程序主体尚未动工时明确，这样才能更好的从全局角度编写代码。而其中的一些细节性记录则应随着程序的编写逐步完善，通过总结性的表格与对比统计，可以从程序本身浩瀚的代码中抽象出逻辑本质，进一步精炼设计思路。这样可以加快程序编写速度，加强程序编写质量。

综合来说，程序设计文档在程序开发中起到了提纲挚领的作用，它的编写与程序编写同步，它们互为表里，相辅相成。

1. 实验总结
2. 实验总结

1.实验收获：

通过本次实验我实践了链表的创建、初始化、加载数据与释放内存操作，熟悉了这一数据结构的运用。

通过本实验我深刻认识了多源文件在工程中的应用，通过良好的分类与代码管理，我的程序保持了一个清晰的结构，增强了可读性。

通过本实验我熟悉了system函数的使用，掌握了其两种输入方式的用法，并成功多次调用lab4程序以完成lab5要求。

通过本实验我深刻认识了二进制文件的读写机制，完善了对fopen函数的w、r、r+、wb和rb模式的认识。

通过本实验我认识了文件的相对路径在多文件管理、写入、读取中的作用，它方便地简化了程序操作。

通过本实验我认识了二维数组、结构体数组、指针数组与链表这四种数据结构的具体存在形式，加深了对它们的理解。

2.实验困难：

对二维数组的指针形式认识模糊：通过查阅资料、咨询同学等方式完善了对二维数组指针形式的理解，认识了该形式的灵活性。

在读取数据时报告内存不足：通过仔细阅读代码，发现是使用了函数局部变量形式的分配内存方式，这样导致了一旦函数关闭则分配的内存消失。解决方法是再增加一层指针，例如二维数组的输入参数形式为\*\*\*int，这样就可以像以往那样直接在数据的真实地址上修改数据，分配内存。其中的嵌套关系较难理解，需要多实践掌握。

在读取二进制数据文件时出现乱码：通过查阅资料、询问同学、请教老师等方式发现是写入与读写不规范，fopen的二进制写入模式应该是wb，读取模式应该是rb，使用其他模式会造成乱码，指针停滞不前，干脆利落的报错等情况。

在打开文件时报文件指针为空：通过仔细审查代码，发现是前面调用同样文件时没有进行关闭操作，这种低级错误仍然犯，需要警醒。

1. 对小组其他成员的文档审查报告

陈明强同学：他的函数整合了相对于我的子函数，在读写上没有我的中转站结构直接读取，更加快速，但缺少鲁棒性。

高原同学：free函数的循环操作更合理，函数注释上更详细。