#### 天津理工大学高级程序设计语言I课程

#### 综合实验报告

姓名： 王帆 学号： 20152180 成绩：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| 评阅意见： |

一、项目名称：银行自动提款机ATM模拟系统

二、任务要求：

（1）有一个文件记录了所有帐户的信息，每个账户包括账户编号、姓名、密码和当前余额。设计适当的结构体读取和存储账户信息。

（2）用户登录系统：用户输入自己的id和密码，从文件中比对是否是合法的用户，当连续输入三次密码错误时，提示用户账户被冻结；

（3）用户合法后，可以查询余额、取款和存款，可以更改密码，并且所有的操作都更新到文件中；

（4）选做：ATM系统记录该机器最近的所有操作日志，以追加的方式写入到系统的另外一个文本文件中。

三、评分依据

（1）程序功能正确，用户界面交互设计合理；

（2）函数的定义清晰，程序结构组织条理；

（3）6月20日之前提交程序电子版到xugp@tjut.edu.cn和实验报告纸质版。

（4）若发现程序和实验报告雷同，按0分计。

报告中应该包括以下内容（3-5页左右为宜）：

1. 利用文字描述程序设计的思路，相应的结构，界面的组织，实现的方式；关键代码处理说明。
2. 程序设计过程中，运行的效果和测试情况；
3. 在设计期间，出现的常见的错误、原因及其解决方法；
4. 总结收获。

# 设计思路

ATM机是一种常见的自助柜员机，其本质为一个交互式自助系统。

在之前的课程中，我们也曾经利用所学知识制作模拟ATM机软件，本次实验，我们利用文件操作和结构体相关知识构建更加完备的ATM机软件，以满足更加良好的用户体验。

本程序设计思路：

1. 利用结构体相关知识，构建前导程序：用户信息生成。从而实现利用结构体从文件中读取相关用户信息，也能满足更加广泛的需求。
2. 信息匹配：利用for循环语句和if条件语句进行账户信息判断。如果错误，跳出判断语句重新获取信息，直到正确为止。
3. 函数思想：本程序各个功能均抽象为函数进行调用操作。各个函数在main函数中利用switch语句进行选择调用，从而实现程序模块化。
4. 宏的自定义使用：利用#define语句进行自定义宏，提高程序的可移植性。
5. 数据控制：由于存取款操作中可能会存在数据超过范围以及存取操作单位为100的情况，通过运用判断语句进行控制。
6. 其他函数调用：
7. 使用延时函数sleep进行模拟界面切换时读条操作，使界面更为良好。
8. 使用暂停函数system(“pause”)和清屏函数system(“cls”)进行界面优化

# 程序结构

# 界面组织及其实现方式

结构体：struct user\_list

结构体内部包含用户ID（int类型）， 用户姓名（char类型数组）， 密码（int类型），金钱（float类型）与账户状态（int类型，仅存在0/1）。

另外，结构体结尾直接声明了一个大小为100的结构体数组，作为程序调用。

主程序：模拟ATM机

首先调用welcome函数实现交互界面，获取ID密码，并进行正误判断。

welcome函数中包括五部分，分别是get\_id, check\_id, get\_password, check\_password, 和get\_point，分别进行ID获取，检测，密码获取，检测和内容获取。

其次调用menu函数，进入主菜单。

menu函数中运用switch进行选项操作，包括四个部分，分别是dm（存款），wm（取款），lm（查询余额），change（更改密码）与exit（退卡）

本程序中利用结构体进行信息存储，利用文件操作相关函数进行信息的读写

数据库：SQL.dat

数据库的格式为：ID 姓名 密码 余额 账户状态。

可以直接使用记事本编辑，后缀为dat格式。

其中账户状态为0时，说明账户由于3次错误输入密码而被冻结，可以手动改为1使其可用。

日志文件：log.txt

日志文件的格式为：时间戳 ID 日志内容

可以直接使用记事本打开查看，后缀为txt格式。

日志中可能出现的内容为：

1. draw money %d error.—取款%d错误
2. draw money %d succeed.—取款%d成功
3. deposit money %d error.—存款%d非法
4. deposit money %d succeed—存款%d成功
5. change password error.—密码修改失败
6. change password succeed.—密码修改成功
7. try to sign in the system.—尝试登陆系统
8. %d was stopped to sign in the system.—ID为%d的账户被禁止登陆系统

# 关键代码说明

1.

void delay()

{

int z;

printf("请稍后，正在处理中");

for(z=0;z<3;z++)

{

sleep(1);

printf(".");

fflush(stdout);

}

}

Delay函数调用sleep函数进行演示操作，并强制刷新。

（由于机器问题，本应为1000毫秒但是被调用为秒，故改为1）

2.

if(det[temp].condition == 0)

{

printf("\n账户已被冻结，请联系管理员！\n");

fprintf(log,"%d was stopped to sign in the system.\r\n",det[temp].id);

exit(-2);

}

冻结操作与记录

3.

void write\_SQL(int len,struct user\_list det[])

{

int i;

FILE \* fp = fopen("SQL.txt","w");

for(i =0;i<len;i++)

{

fprintf(fp,"%d %s %d %.2f %d\r\n",det[i].id, det[i].name, det[i].password, det[i].money, det[i].condition);

}

fclose(fp);

}

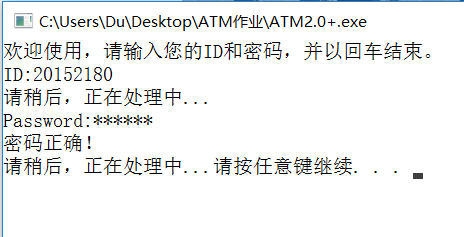
数据库更新函数：利用fprintf进行更新操作，从而实现数据的实时性

# 调试情况及出现的问题

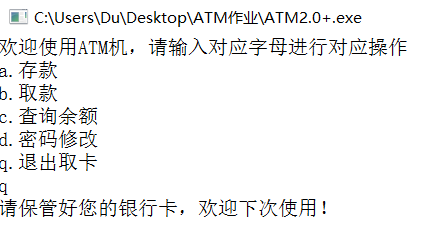
1. 分号，逗号等基本符号使用错误：如出现中英文符号混用，漏写或者混写。
2. 数组长度定义出错
3. 起初准备使用字符串控制ID和密码长度，但是出现了警告，提示char类型过长而可能导致的潜在风险。最后还是换作int类型。
4. 编译初期由于编程过程中的一些偏差导致某些变量重复声明多次，已改正。
5. ID验证程序提示找不到信息
6. sleep函数参数单位问题：本应为毫秒但是被调用为秒
7. 运行后数据库丢失（暂时无法解决）
8. 文件操作相关用法错误

# 运行结果

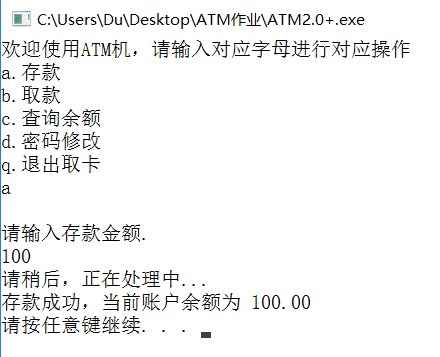
Welcome界面



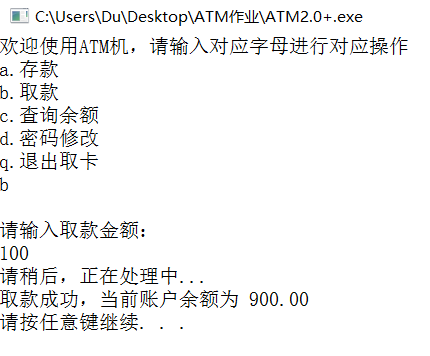
退卡选项



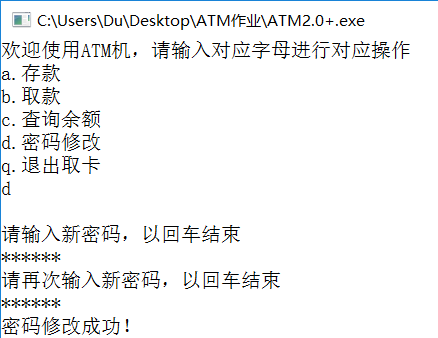
存款选项



取款选项



密码修改选项



# 总结与收获

通过本次ATM模拟的实验，我加深了对于文件操作与结构体操作相关知识的理解和掌握。最终的程序虽然还有BUG，但是相信在接下来的时间里我将通过和其他同学交流，查阅资料，请教老师将其修正。

程序设计的过程是思维转变为实际的过程，需要努力也需要练习。这次实验让我清醒地认识到我对于C语言的学习还只是浅尝，在接下来的暑假中我要继续钻研相关内容，打好基础，为之后其他语言和知识的学习做好铺垫。