函数设计

问题参数化描述

详细设计

1. 存储定义 类型

struct User

{

Char stu\_name[10];

Char stu\_num[20];

Char tea\_name[10]

Char tea\_num[20]

Double achie

Int arship

Struct User\* next

}

struct User\* temp\_storage 链表

1. 函数设计 （功能合并、整合）

登录

Int logon(string u)

{

//学号 u

//与系统学号作比较

//若用户为教师返回1

为学生返回0

学工号不存在则提示错误

}

浏览

Void xsxx\_ll()

{

//打印输出

}

查找

Void xsxx\_cz(string u)

{

//学号 u

//遍历链表

//打印输出

}

删除

Void xsxx\_sc(string u)

{

//学号 u

//与链表中学生信息对比

//该学生信息不存在，报错

该学生信息存在

//用户选择是否保存

//用户选择保存

//删除该学生所有信息

//更新链表写入文件

//用户选择不保存

//退出程序

//无返回值

}

添加

Void xsxx\_tj(string u)

{

//学号 u

//与链表中学生信息对比

//若重复，提示错误

文件中不存在该学生信息

//往学生信息表中添加该学生信息

//更新链表写入文件

//无返回值

}

修改

Void xsxx\_ xg(string u)

{

//学号 u

//与链表中学生信息对比

//该学生信息不存在，报错

该学生信息存在

//找到与用户输入一致的学生

用户选择是否保存

//用户选择保存

//根据用户输入修改学生信息

//更新链表写入文件

用户选择不保存

//退出程序

//无返回值

}

int write\_data\_Student(struct User\* head)

{

//参数为链表的头

//写入文件的函数

//将链表里的内容写入文件

}

struct User\* temp\_storage(struct User\* head)

{

//创建链表

//将用户输入的内容存进链表里

}

函数调用关系

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| main | Logon | interface | Xsxx\_ll | |
| interface\_two | Xsxx\_cz | |
| Xsxx\_xg | write\_data\_Studen |
| Xsxx\_sc | write\_data\_Studen |
| Xsxx\_tj | write\_data\_Studen\_two |