# 安徽工程大学

课程设计说明书

课程设计名称：高级语言程序设计

课程设计题目：计算机设备管理

指 导 老 师：伊 芸 芸

专 业 班 级：计算机科学与技术

学 生 姓 名：张 子 杨

学 号：3210701218

起 止 日 期：2022.5.21-2021.6.8

总 评 成 绩：

**前 言**

随着单位的快速发展，计算机的普及和计算机在单位各部设备的调整，计算机设备的维护与维修工作也大量增加，如何更及时、有效地掌握单位计算机设备的有关维护、维修、新增、报废、移动、调整等情况，对学校管理计算机相关设备的工作人员提出了挑战。以前手工记账的手段，已远远不能适应这样的要求，如何利用现代技术的管理手段进行计算及设备管理，也是许多单位面临的问题。

计算机实验室设备管理系统的设计和应用,有利于提高实验室设备管理和维护的效率,促进实验室设备管理的科学化和信息化发展。对系统在设计中采用的关键性技术进行了分析,对系统的设计原则以及系统设计的功能需求进行了研究,结合计算机实验室设备管理的信息化需要,在SOA面向服务架构的基础上,提出了系统的设计方案并进行了研究。

本文以学校的计算机设备管理为例，讲述了如何开发软件来管理学校计算机设备以及开发计算机设备管理系统中遇到的问题和解决方案。

**目 录**

[**第1章 概述**](#_Toc24580)

[1.1 设计目的](#_Toc8687)

[1.2 课程涉及的内容](#_Toc11270)

[1.3 课程设计的要求与数据](#_Toc9578)

[**第2章 课程设计需求分析**](#_Toc7351)

[2.1 用牛顿迭代法解方程](#_Toc19734)

[2.2 埃及分数](#_Toc30483)

[2.3计算机设备管理系统](#_Toc3577) -7-

[**第3章 系统设计**](#_Toc30754)

[3.1 总体设计](#_Toc9417)

[3.2 详细设计](#_Toc13921)

[**第4章 编码**](#_Toc28710)

[4.1 主页面编码](#_Toc6047)

[**第5章 运行与调试分析**](#_Toc395)

[**第6章 总结**](#_Toc18094)

[**参考文献**](#_Toc17087)

[**附录**](#_Toc12174)

**任 务 书**

一、**课程设计题目：** 计算机设备管理系统 （填写课题组中的第3题名称）

**二、设计任务和内容**

内容共有3道大题。第1、2题多为应用类题目，第3题为综合类题目，涉及循环、数组、函数、链表等主要知识点。针对选题，实现三道大题，并进行综合设计与实现。具体任务包括：

1、熟悉实训教材中的课程设计的目的、要求与过程。

2、掌握实训教材中的课程设计的内容。

3、针对选题进行需求分析，特别是功能分析，划出功能图。

4、算法分析与设计，写出每个模块的设计思想，画出每个算法的流程图或N-S图。

5、编写源程序。

6、静态走查程序和上机调试程序，写出调试报告。

7、综合程序。如果三道题目实现时，是以三个程序独立完成的，则将三道题目，组合成一个程序（以命令菜单的方式，供用户选择每题实现的功能），并进行调试。也可以在开始设计时就以一个程序实现，通过菜单的方式实现功能的选择。

8、书写上述文档和撰写课程设计说明书。

**三、设计步骤和要求**

**1、设计步骤**

首先进行相关资料查阅和学习、调查研究，熟悉基本的业务流程和管理对象的属性，确定系统功能和数据需求。然后结合软件工程的理论完成设计任务，主要包括需求分析、总体设计以及详细设计、编码与调试部分。

**2、课程设计说明书要求与格式**

**(1) 基本要求：**

①能反映完成了上述设计内容要求。

② 要求撰写不少于5000个文字（20页）的文档。

③ 文档中至少要包括：主要功能分析、系统功能结构图、算法流程图、系统运行与调试截图、小结、参考文献。

④ 课程设计说明书可以打印或手写，其中用户界面设计、程序运行与调试需要截图打印。

**(2) 文档格式要求**

① 封面

② 前言

③ 目录

④ 任务书（须有指导教师签名及日期）

⑤ 正文（分章、层次等，每一章从新一页开始）**（每章内容参考如下：）**

**第1章 概述**

　 包括课题的课程设计的目的与要求、选题的内容、设计的过程、开发环境等内容。

**第2章 需求分析**

描述完成哪些功能，以及每一个功能所完成的任务。

**第3章 系统设计**

3.1总体设计

画出系统功能结构图。

3.2详细设计

画出主要算法的流程图。

确定程序中所用的主要数据的数据类型，选择所用的数据结构，如：结构体、数组、链表等。

列出程序中用到的所有函数的名称、功能，调用关系。

　 界面设计

**第4章 编码**

　　列出程序中结构体类型的定义。

列出程序中全局变量、程序中通用的符号常量。

列出自己设计的所有函数的原型，并说明功能、参数的意义。

给出主要功能的代码并有适当的注释。

**第5章 运行与调试**

运行、调试程序并截图，进行说明调试的目的与结果。初步学会软件测试。

**第6章 小结**

对本次课程设计过程中的收获、体会以及不足或存在的问题进行总结。

**参考文献**

按照要求列出至少10篇参考文献，注意撰写格式。

**附录：附上第4章没有列出的代码。**

**(3) 说明书排版与装订要求**

说明书排版参照“安徽工程大学本科毕业设计（论文）正文模板”的要求。课程设计说明书必须按要求和顺序装订，封面的指导老师是课表中课程设计指导老师，不一定是理论课老师。

说明：该门课程的任务在放寒假前布置给学生。

教师签名：

日 期： 2022 年 5 月 26 日

# 第1章 概述

## 1.1 设计目的

（1） 用牛顿迭代法解方程：牛顿迭代法（Newton's method）又称为牛顿-拉夫逊方法（Newton-Raphson method），它是牛顿在17世纪提出的一种在实数域和复数域上近似求解方程的方法。多数方程不存在求根公式，因此求精确根非常困难，甚至不可能，从而寻找方程的近似根就显得特别重要。  
牛顿迭代法使用函数f(x）的泰勒级数的前面几项来寻找方程f(x) = 0的根。牛顿迭代法是求方程根的重要方法之一，其最大优点是在方程f(x) = 0的单根附近具有平方收敛，而且该法还可以用来求方程的重根、复根，此时线性收敛，但是可通过一些方法变成超线性收敛。另外该方法广泛用于计算机编程中。

（2） 埃及分数：埃及分数是指分子是1的分数(三分之二除外，它有自己的符号)，也叫单位分数。古代埃及人在进行分数运算时。只使用分子是1的分数。因此这种分数也叫做埃及分数，或者叫单分子分数。

埃及人之所以未能把算术和代数发展到较高水平，其分数运算之繁杂也是原因之一。

（3）建立计算机设备管理系统：对计算机设备相关信息进行管理。

计算机设备的相关信息包括：计算机设备编号、名称、单价、数量和金额等。

## 1.2 课程涉及的内容

（1） 牛顿迭代法：令f(x)=a\*x\*x\*x+b\*x\*x+c\*x+d=[(a\*x+b)\*x+c]\*x+d。求出方程的表达式f(x)的导数f1(x)=(3\*a\*x+2\*b)\*x+c。计算曲线上的点（x0,f(x0）)的切线与x轴的交点x:

x=x0-f(x0)/f1(x0)

任意取初始点x，在循环中进行迭代：

x0=x

x=x0-f(x0)/f1(x0)

依次继续，直至x与x0的差满足指定精度的要求为止。

（2） 埃及分数：若真分数的分子a能整除分母b，则真分数经过化简就可以得到埃及分数；

若真分数的分子不能整除分母，则可以在原来的分数中分解出一个分母b/a+1的埃及分数。用这种方法将剩余部分反复分解，最后可得到结果。

（3） 提供文件的输入输出操作。

要能够完成设备的录入与修改。

需要提供设备添加和修改操作。

实现对设备进行分类统计。

## 1.3 课程设计的要求与数据

1. 建立该系统的储存结构。
2. 录入计算机设备的相关信息。
3. 查找某个计算机设备的相关信息。
4. 修改某个计算机设备的相关信息。
5. 删除某个计算机设备的相关信息。
6. 计算每个计算机设备的金额。
7. 统计所有计算机设备的总数量和总金额。

将每个计算机设备当做一个结点，类型为结构型，结构体中包含计算机设备的相关信息和结点指针。定义计算机设备结点的结构为：

struct node

{ char no[12]; /\*设备编号\*/

char name[40]; /\*名称\*/

float dj; /\*单价\*/

int num; /\*数量\*/

float jine; /\*金额\*/

struct node \*link; /\*指向下一结点的指针\*/

};

按照编号查找结点，可从头结点开始顺序查找，比较结点可用字符串比较函数strcmp来实现。利用公式求出金额后，将结果循环给结点中的数据项。统计总数量和总金额时，从链表头开始将相应字段循环叠加，直至链表尾。

**第2章 课程设计需求分析**

## 2.1 用牛顿迭代法解方程

设r是f(x)=0的根，选取x0作为r的初始近似值，过点(x0,f(x0))做曲线y=f(x)的切线L，L的方程为y=f(x0）+f’(x)(x-x0)，求出L

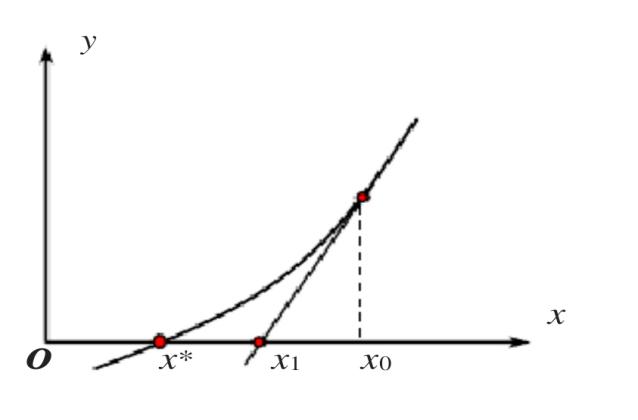


图2-1牛顿迭代法示意图

与x轴交点的横坐标x1=x0-f(x0)/f’(x0)，称x1为r的一次近似值。过点(x1,f(x1))做曲线y=f(x)的切线，并求该切线与x轴交点的横坐标x2=x1-f(x1)/f’(x1)，称

x2为r的二次近似值。重复以上过程，得r的近似值序列，其中，xn+1=xn-f(xn)/f’(xn)称为r的n+1次近似值，上式称为牛顿迭代公式。  
 用牛顿迭代法解非线性方程，是把非线性方程f(x）=0线性化的一种近似

方法。把凡f(x)在点x0的某邻域内展开成泰勒级数，取其线性部分（即泰

勒展开的前两项），并令其等于0，即f(x0)+f’(x0)(x-x0)=0，以此作为非线性方程f(x)=0的近似方程，若f’(x0)=0，则其解为x1=x0-f(x0)/f’(x0)， 这样，得到牛顿迭代法的一个迭代关系式：xn+1=xn-f(xn)/f’(xn)。

## 2.2 埃及分数

【贪心算法】

设a、b为互质正整数，a<b 分数a/b 可用以下的步骤分解成若干个单位分数之和：  
步骤一：用b除以a，得商数q1 及余数r1。（r1=b - a\*q1）  
步骤二：把a/b 记作：a/b=1/(q1+1）+(a-r)/b(q1+1）  
步骤三：重复步骤2，直到分解完毕  
 3/7=1/3+2/21=1/3+1/11+1/231  
 13/23=1/2+3/46=1/2+1/16+1/368  
 以上其实是数学家斐波那契提出的一种求解埃及分数的贪心算法，准确的算法表述应该是这样的：  
 设某个真分数的分子为a，分母为b;  
 把b除以a的商部分加1后的值作为埃及分数的某一个分母c；  
 将a乘以c再减去b，作为新的a；  
 将b乘以c，得到新的b；  
 如果a大于1且能整除b，则最后一个分母为b/a；算法结束；  
 或者，如果a等于1，则，最后一个分母为b；算法结束；  
 否则重复上面的步骤。  
 备注：事实上，后面判断a是否大于1和a是否等于1的两个判断可以合在一起，及判断b%a是否等于0，最后一个分母为b/a，显然是正确的。

## 2.3 计算机设备管理系统

（1） 画出系统功能结构图。

（2） 画出主要算法的流程图。

（3） 确定程序中所用的只要数据的数据类型，选择所用的数据结构。

（4） 列出程序中用到的所有函数的名称、功能，调用关系。

struct node

{ char no[12]; /\*设备编号\*/

char name[40]; /\*名称\*/

float dj; /\*单价\*/

int num; /\*数量\*/

float jine; /\*金额\*/

struct node \*link; /\*指向下一结点的指针\*/

};

（5） 界面设计。

# 第3章 系统设计

## 3.1 总体设计

1. 该系统使用时首先在屏幕上打印出菜单

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 计算机设备管理 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [1]建立管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [2]查找指定设备编号的记录 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [3]修改管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [4]删除管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [5]输出管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [6]计算每个设备金额 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [7]统计设备总数量和总金额 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [0]退出系统 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

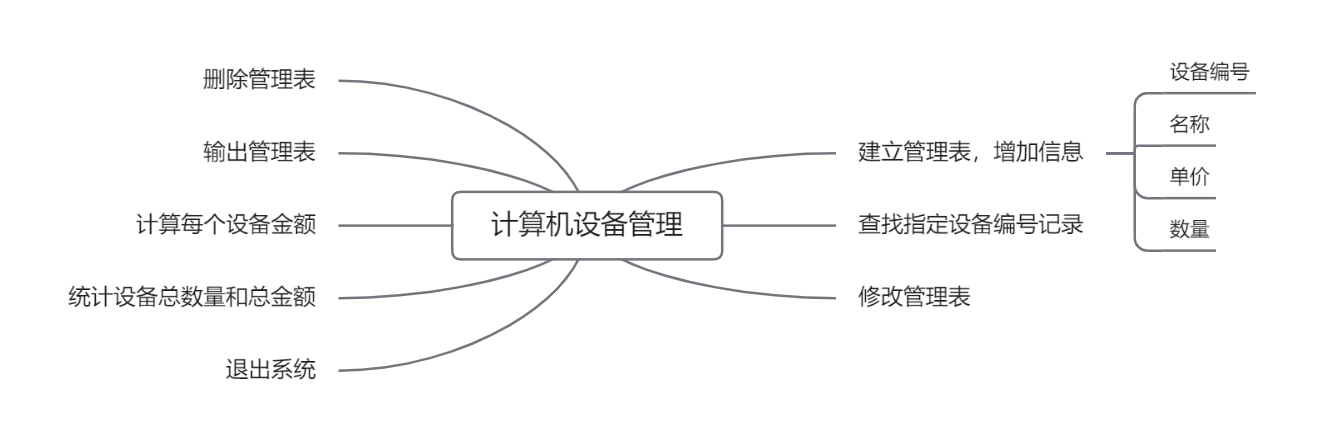
（2）下面是系统功能图

图3-1计算机设备管理系统思维导图

## 3.2 详细设计

1. 牛顿迭代法解方程基本原理

给定初始值x0，&为根的容许误差，@为f(x)的容许误差，N为迭代次数的容许值。如果f’(x)=0或迭代次数大于N，则算法失败，结束：否则执行。

计算 x1=x0-f(x0)/f’(x0)。

若x1-x2<&或f(x1)<@,则输出x1，程序结束：否则执行。

令x0=x1,转向1。

流程图：

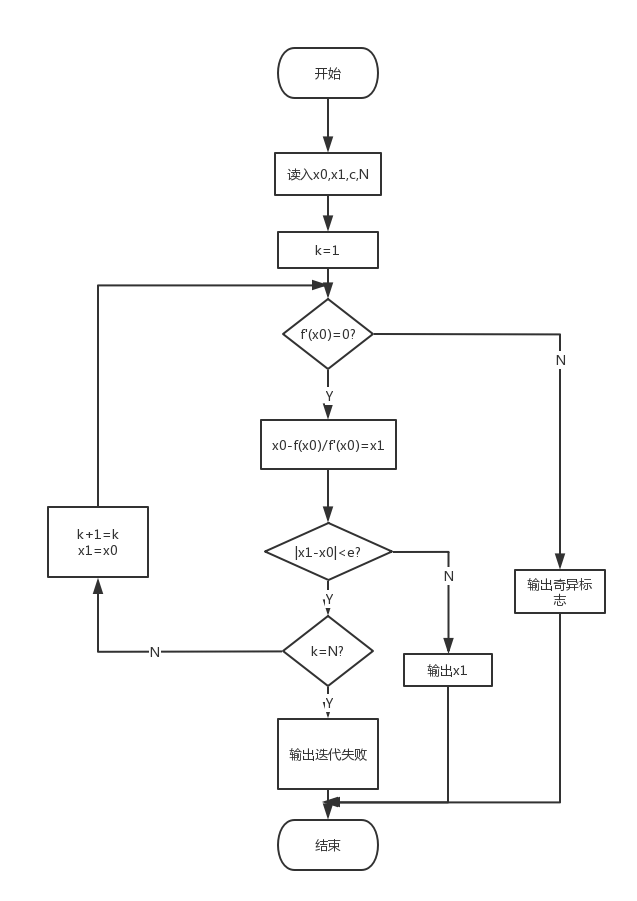


图3-3 牛顿迭代法流程图

1. 埃及分数基本原理

在古埃及，人们使用单位分数的和(形如1/a的, a是自然数)表示一切有理数。 如：2/3=1/2+1/6,但不允许2/3=1/3+1/3,因为加数中有相同的。 对于一个分数a/b,表示方法有很多种，但是哪种最好呢？ 首先，加数少的比加数多的好，其次，加数个数相同的，最小的分数越大越好。 如： 19/45=1/3 + 1/12 + 1/180 19/45=1/3 + 1/15 + 1/45 19/45=1/3 + 1/18 + 1/30, 19/45=1/4 + 1/6 + 1/180 19/45=1/5 + 1/6 + 1/18. 最好的是最后一种，因为1/18比1/180,1/45,1/30,1/180都大。 给出a,b(0〈a〈b〈1000),编程计算最好的表达方式。 Input 第一行:N 表示有N组测试数据，每组测试数据为一行包含a,b(0〈a〈b〈1000)。 Output 每组测试数据若干个数，自小到大排列，依次是单位分数的分母。

流程图为：

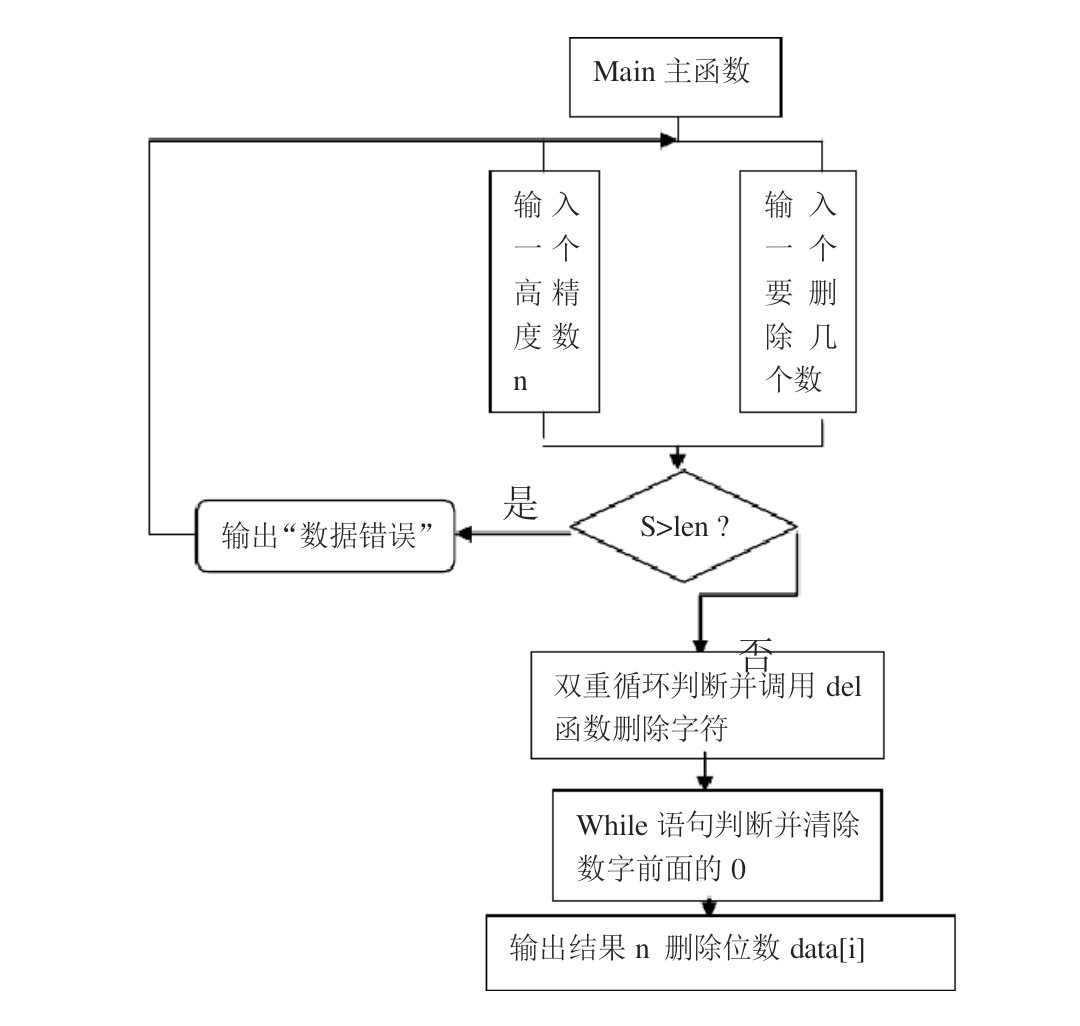


图3-4 埃及分数流程图

1. 计算机设备管理系统基本原理

计算机设备管理系统基于先进的计算机信息化技术，解决设备管理中的难点和一些关键问题。计算机设备管理系统侧重于对设备的备件管理，同时兼顾设备管理相关的基本信息设置、录入、数据查询、数据修改及删除等功能。

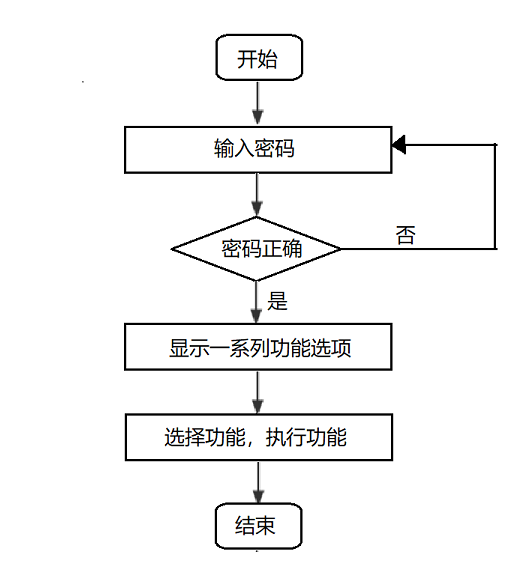


图3-5 总系统流程图

# 第4章 编码

## 4.1 主页面编码

（1）用牛顿迭代法解方程

int main()  
{  
    double solut(double,double,double,double );//函数原型声明  
    double a,b,c,d;  
    scanf("%lf%lf%lf%lf",&a,&b,&c,&d);  
    printf("%.2f",solut(a,b,c,d));  
    return 0;  
}  
   
double solut(double a,double b,double c,double d)  
{  
    double x=1,x0;  
    do  
    {  
        x0=x;  
        x=x0-(((a\*x+b)\*x+c)\*x+d)/((3\*a\*x+2\*b)\*x+c);  
    }  
    while(fabs(x-x0)>=1e-5);

为点xn+1与xn之间的距离，当两点的距离无限接近于0时，就时我们所要求的根x\*。如果取x\*这个点为例，我们发现方程在该点的切线与x轴的交点为x\*，此时x与x0之间的距离为零。  
    return x;

}

（2）埃及分数

struct FS{  
    int a;  
    int b;  
};  
int is\_aijifs(FS\* f);  
FS simple(FS\* f);  
int gcd(FS\* f);  
int main(){  
    FS f;  
    int m=2;  
    printf("请输入分子分母，以空格隔开！\n");  
    scanf("%d%d",&f.a,&f.b);  
    printf("%d/%d=",f.a,f.b);  
    while(1){  
        if(is\_aijifs(&f)){  
            printf("%d/%d",f.a,f.b);  
            break;  
            }  
        while(1.0\*f.a/f.b<1.0\*1/m)   m++;  
            printf("1/%d+",m);  
            f.a=f.a\*m-1\*f.b;  
            f.b=f.b\*m;  
            //printf("%d %d",f.a,f.b);  
            f=simple(&f);  
        //    printf("%d %d",f.a,f.b);  
        }  
    return 0;  
}  
int is\_aijifs(FS\* f){  
    if(f->a==1) return 1;  
    return 0;  
}  
FS simple(FS\* f){  
    int g=gcd(f);  
    f->a=f->a/g;  
    f->b=f->b/g;  
    //printf("   %d\n",f->a);  
    return \*f;//有问题  
}  
int gcd(FS\* f){  
    FS f1;  
    f1=\*f;  
    int r;  
    while(1){  
        r=f1.a%f1.b;  
        f1.a=f1.b;  
        f1.b=r;  
        //printf("    %d\n",r);  
        if(r==0) return f1.a;  
    }  
}

（3）计算机设备管理系统

int main()

{

struct node\* head = NULL;//定义指向struct node 结构体变量的指针变量

int pw = 123;

int paw = 0;

printf("请输入登录密码：\n");

scanf("%d", &paw);

if (paw == pw)

{

printf("密码正确\n");

while (1)

{

int n;

menu();

printf("请输入:>\n");

scanf("%d", &n);

switch (n)

{

case 0:

printf("退出:>\n");

return 0;

break;

case 1:

create(&head);//创建链表函数,增加信息

break;

case 2:

search(head);//查找信息

break;

case 3:

revise(&head);//修改管理表

break;

case 4:

delete(&head);//删除管理表

break;

case 5:

print(head);//打印信息

break;

case 6:

count(head);//计算每个设备金额

break;

case 7:

stats(head);//统计设备总数量和总金额

break;

default:

printf("输入错误，请重新输入\n");

break;

}

}

}

else

{

printf("密码错误，请选择下列操作>\n");

printf("请输入0退出，或输入其他任意数字重新输入\n");

int input = 0;

scanf("%d", &input);

if (input == 0)

return 0;

}

}

return 0;

}

# 第5章 运行与调试分析

（1）登录系统并输入密码。

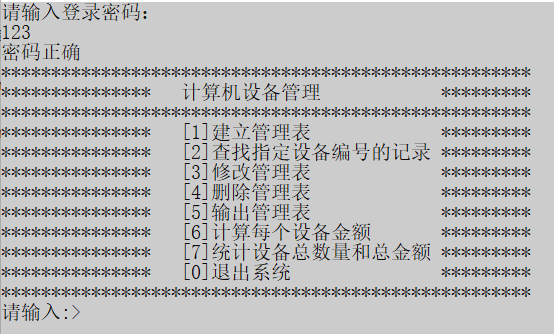


图5-1 系统登录页面

（2）计算机设备管理系统首先要建立管理表

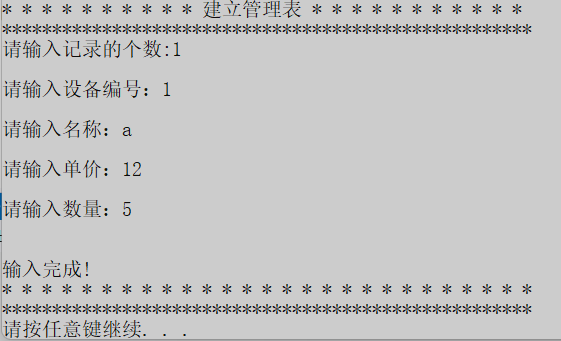
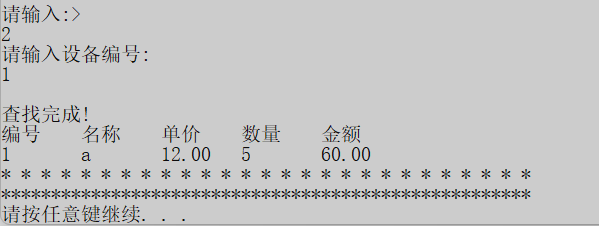
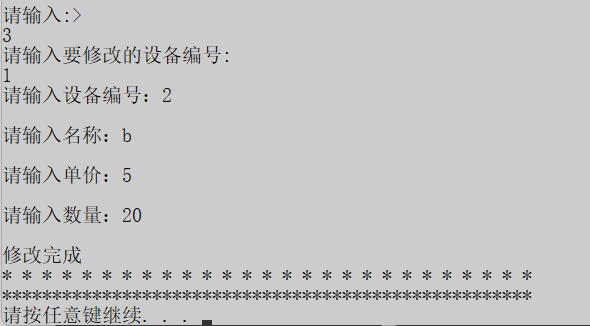


图5-2 建立管理表页面



。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。

图 图5-3查找指定设备页面。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。

。。。。。。。。。。。。。。。。

。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。

图5-4 修改管理表页面

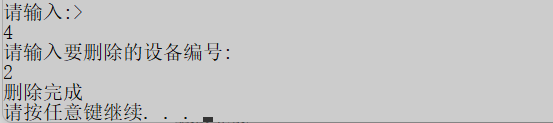


图5-5 删除管理表页面

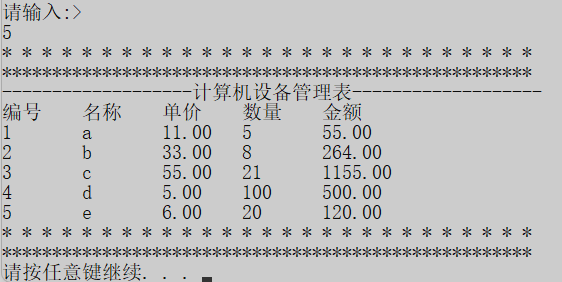


图5-6 输出管理表页面

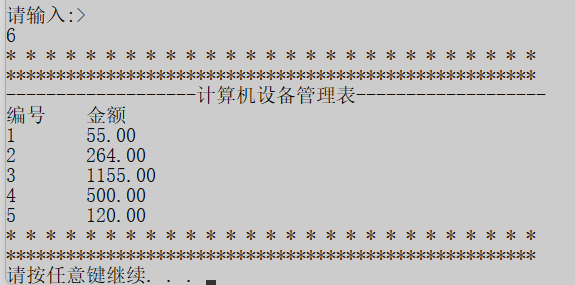


图5-7计算每个设备金额页面

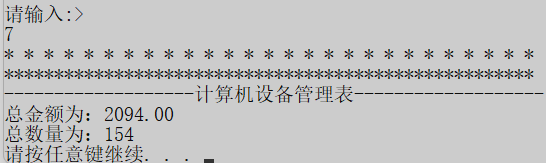


图5-8统计设备总数量和总金额页面



图5-9退出页面

# 第6章 总结

经过一学期C语言程序的学习，让我深刻明白了这门课的重要性。C语言作为一门计算机学习的基础课，是我们掌握计算机必学的课程，对于现如今互联网飞速发展的时代，掌握C语言对我们尤其重要。

通过这次课程设计，本人在多方面都有所提升。通过这次课程设计，综合运用本专业所学的课程理论，巩固与计算机设备管理系统等课程的学习，掌握了一个项目设计的方法和步骤，提高了计算能力，绘图能力，熟悉了规范和标准，同时各科相关课程都有全面的应用，独立思考能力得到了提高。在设计过程中体现出自己单独设计计算机设备管理系统的能力以及综合运用知识的能力体会了学以致用、突出自己劳动成果的喜悦心情，并从中发现自己的不足从而加以弥补。

# 参考文献

【1】朱晓刚 《计算机设备管理系统分析 》[J]2008（6）

【2】闫剑锋 《计算机网络技术在煤矿设备管理中的应用》[J].2021(1)

【3】张国俊 《计算机与电子类设备机房的管理与维护探讨》[J].2021(1)

【4】丁雪非、 罗尉平《计算机网络设备维护与安全管理的措施探讨》[J]2018(11)

【5】孙程程 《计算机局域网的管理与维护技术》[J]2020(12)

【6】封磊 《大数据时代计算机专业设备管理探究》[J]2020(11)

【7】常雪玲 《探析医疗设备计算机维护与管理》[J]2020(11)

【8】潘玮 《计算机技术与网络技术在机房管理中的应用》[J]2018（6）

【9】朱福良 《企业计算机设备运维管理探析》[J]2020(10)

【10】景巧丽、赵全洲、曹海英、元元 《计算机实验室设备管理系统设计与研究》[J]2020(10)

# 附录

题目1

int main()  
{  
    double solut(double,double,double,double );//函数原型声明  
    double a,b,c,d;  
    scanf("%lf%lf%lf%lf",&a,&b,&c,&d);  
    printf("%.2f",solut(a,b,c,d));  
    return 0;  
}  
   
double solut(double a,double b,double c,double d)  
{  
    double x=1,x0;  
    do  
    {  
        x0=x;  
        x=x0-(((a\*x+b)\*x+c)\*x+d)/((3\*a\*x+2\*b)\*x+c);  
    }  
    while(fabs(x-x0)>=1e-5);//为点xn+1与xn之间的距离，当两点的距离无限接近于0时，就时我们所要求的根x\*。如果取x\*这个点为例，我们发现方程在该点的切线与x轴的交点为x\*，此时x与x0之间的距离为零。  
    return x;  
}

题目2  
struct FS{  
    int a;  
    int b;  
};  
int is\_aijifs(FS\* f);  
FS simple(FS\* f);  
int gcd(FS\* f);  
int main(){  
    FS f;  
    int m=2;  
    printf("请输入分子分母，以空格隔开！\n");  
    scanf("%d%d",&f.a,&f.b);  
    printf("%d/%d=",f.a,f.b);  
    while(1){  
        if(is\_aijifs(&f)){  
            printf("%d/%d",f.a,f.b);  
            break;  
            }  
        while(1.0\*f.a/f.b<1.0\*1/m)   m++;  
            printf("1/%d+",m);  
            f.a=f.a\*m-1\*f.b;  
            f.b=f.b\*m;  
            //printf("%d %d",f.a,f.b);  
            f=simple(&f);  
        //    printf("%d %d",f.a,f.b);  
        }  
    return 0;  
}  
int is\_aijifs(FS\* f){  
    if(f->a==1) return 1;  
    return 0;  
}  
FS simple(FS\* f){  
    int g=gcd(f);  
    f->a=f->a/g;  
    f->b=f->b/g;  
    //printf("   %d\n",f->a);  
    return \*f;//有问题  
}  
int gcd(FS\* f){  
    FS f1;  
    f1=\*f;  
    int r;  
    while(1){  
        r=f1.a%f1.b;  
        f1.a=f1.b;  
        f1.b=r;  
        //printf("    %d\n",r);  
        if(r==0) return f1.a;  
    }

题目3：计算机设备管理系统

1. 头文件：

#pragma once

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

#include<string.h>

#include<Windows.h>

struct node //声明结构体类型struct node

{

char no[12];//设备编号

char name[40];//名称

double dj;//单价

int num;//数量

double jine;//金额

struct node\* next;//指向下一个节点的指针

};

struct node\* head;//定义指向struct node 结构体变量的指针变量

void menu();

void create(struct node\*\* phead);

void search(struct node\* head);

void revise(struct node\*\* phead);

void delete(struct node\*\* phead);

void print(struct node\* head);

void count(struct node\* head);

void stats(struct node\* head);

1. 系统文件

#define \_CRT\_SECURE\_NO\_WARNINGS 1

#include"system.h"

void menu()

{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 计算机设备管理 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [1]建立管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [2]查找指定设备编号的记录 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [3]修改管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [4]删除管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [5]输出管理表 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [6]计算每个设备金额 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [7]统计设备总数量和总金额 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [0]退出系统 \*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

void create(struct node\*\* phead)//创建链表函数,增加信息

{

int n;

int i;

printf("\* \* \* \* \* \* \* \* \* \* 建立管理表 \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请输入记录的个数:");

scanf("%d", &n);

printf("\n");

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

{

struct node\* newnode = (struct node\*)malloc(sizeof(struct node));//开辟动态存储区

printf("请输入设备编号：");

scanf("%s", &newnode->no);//输入结构体变量struct node 中的成员no的值

printf("\n");

printf("请输入名称：");

scanf("%s", &newnode->name);

printf("\n");

printf("请输入单价：");

scanf("%lf", &newnode->dj);

printf("\n");

printf("请输入数量：");

scanf("%d", &newnode->num);

printf("\n");

newnode->jine = newnode->dj \* newnode->num;

newnode->next = NULL;//赋给next空指针

if (\*phead == NULL)

\*phead = newnode;

else

{

struct node\* tail = \*phead;//定义尾结点指针

while (tail->next != NULL)

{

tail = tail->next;

}

tail->next = newnode;

}

}

printf("\n输入完成!\n");

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

system("pause");

}

void search(struct node\* head)//查找信息

{

char n[12];

struct node\* temp = head;

printf("请输入设备编号:");

printf("\n");

scanf("%s", n);

while(temp != NULL)

{

if (strcmp(temp->no, n) == 0)

{

printf("\n查找完成!\n");

printf("编号\t名称\t单价\t数量\t金额\t\n");

printf("%s\t%s\t%.2f\t%d\t%.2f\n", temp->no, temp->name, temp->dj, temp->num, temp->jine);

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

break;

}

temp = temp->next;

}

if (temp == NULL)

{

printf("\n您输入的设备编号有误，无法查询！\n");

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

system("pause");

}

void revise(struct node\*\* phead)//修改管理表

{

char n[12];

struct node\* temp = \*phead;

printf("请输入要修改的设备编号:");

printf("\n");

scanf("%s", n);

while (temp != NULL)

{

if (strcmp(temp->no, n) == 0)

{

printf("请输入设备编号：");

scanf("%s", &temp->no);//输入结构体变量struct node 中的成员no的值

printf("\n");

printf("请输入名称：");

scanf("%s", &temp->name);

printf("\n");

printf("请输入单价：");

scanf("%lf", &temp->dj);

printf("\n");

printf("请输入数量：");

scanf("%d", &temp->num);

printf("\n");

temp->jine = temp->dj \* temp->num;

printf("修改完成\n");

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

break;

}

temp = temp->next;

}

if (temp == NULL)

{

printf("\n您输入的设备编号有误，无法修改！\n");

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

system("pause");

}

void delete(struct node\*\* phead)//删除管理表

{

char n[12];

struct node\* temp = \*phead;

printf("请输入要删除的设备编号:");

printf("\n");

scanf("%s", n);

if (temp == NULL)

{

printf("\n您输入的设备编号有误，无法删除！\n");

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

}

if (strcmp(temp->no, n) == 0)

{

\*phead = temp->next;

free(temp->next);

temp->next = NULL;

printf("删除完成\n");

}

while (temp->next != NULL)

{

if (strcmp(temp->next->no, n) == 0)

{

temp->next = temp->next->next;

free(temp->next);

temp->next = NULL;

printf("删除完成\n");

break;

}

temp = temp->next;

}

system("pause");

}

void print(struct node\* head)//打印信息

{

struct node\* temp;

temp = head;

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("-------------------计算机设备管理表-------------------\n");

printf("编号\t名称\t单价\t数量\t金额\t\n");

while(temp != NULL)

{

printf("%s\t%s\t%.2f\t%d\t%.2f", temp->no, temp->name, temp->dj, temp->num, temp->jine);

printf("\n");

temp = temp->next;

}

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

system("pause");

}

void count(struct node\* head)//计算每个设备金额

{

struct node\* temp;

temp = head;

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("-------------------计算机设备管理表-------------------\n");

printf("编号\t金额\t\n");

while (temp != NULL)

{

printf("%s\t%.2f", temp->no, temp->jine);

printf("\n");

temp = temp->next;

}

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

system("pause");

}

void stats(struct node\* head)//统计总数量和总金额

{

struct node\* temp = head;

double sum = 0;

int count = 0;

printf("\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("-------------------计算机设备管理表-------------------\n");

while (temp != NULL)

{

sum += temp->jine;

count += temp->num;

temp = temp->next;

}

printf("总金额为：%.2lf\n", sum);

printf("总数量为：%d\n", count);

system("pause");

}

int main()

{

struct node\* head = NULL;//定义指向struct node 结构体变量的指针变量

int pw = 123;

int paw = 0;

while (1)

{

printf("请输入登录密码：\n");

scanf("%d", &paw);

if (paw == pw)

{

printf("密码正确\n");

while (1)

{

int n;

menu();

printf("请输入:>\n");

scanf("%d", &n);

switch (n)

{

case 0:

printf("退出:>\n");

return 0;

//break;

case 1:

create(&head);//创建链表函数,增加信息

break;

case 2:

search(head);//查找信息

break;

case 3:

revise(&head);//修改管理表

break;

case 4:

delete(&head);//删除管理表

break;

case 5:

print(head);//打印信息

break;

case 6:

count(head);//计算每个设备金额

break;

case 7:

stats(head);//统计设备总数量和总金额

break;

default:

printf("输入错误，请重新输入\n");

break;

}

}

}

else

{

printf("密码错误，请选择下列操作>\n");

printf("请输入0退出，或输入其他任意数字重新输入\n");

int input = 0;

scanf("%d", &input);

if (input == 0)

return 0;

}

}

return 0;

}