**安徽农业大学经济技术学院**

**《C++程序设计》课程设计报告书**

|  |  |
| --- | --- |
| **课题名称** | **超市信息管理系统** |
| **姓 名** | **孟定国** |
| **学 号** | **1904070203** |
| **院 系** | **信息与计算机系** |
| **专 业** | **软件工程（2）班** |
| **指导教师** | **王文艺** |
| **起止日期** | **2020.12.30-2021.1.15** |

**2020年 1月 10日**

**《C++程序设计》课程设计任务书**

下达时间：2020年12月30日 设计时间：两周

指导教师：王文艺 完成班级：2019级软件工程专业

**【设计目的】**

本课程设计是在学生学完《C++》课程后进行的一次全面的综合练习，是计算机学科重要的实践性环节之一，目的旨在利用编程知识与技巧，达到理论与实践的统一。做到：

1. 巩固和加深对C++程序设计课程的基本知识的理解和掌握；
2. 掌握和提高C++编程和程序调试基本技能；
3. 进一步理解和运用结构化程序设计的思想和方法；
4. 基本掌握面向对象程序设计的基本思路和方法；
5. 提高运用C++解决简单的面向对象的程序设计问题的能力；
6. 掌握科学地撰写程序设计说明文档和总结报告的能力。

**【设计内容和任务】**

可以在指定的题目中任意选择一题，也可以自选题目，若选题相同，程序基本相同或撰写报告基本相同的情况下，经查证属于相互抄袭的，均以零分计入成绩。参考题目如下：

1、GPA计算器

2、学生信息管理系统

3、学生成绩管理系统

4、商品库存管理系统

5、图书馆管理系统

6、职工工资信息管理系统

7、聊天程序

8、通讯录

9、小型超市管理系统

……

**【设计要求】**

采用教师指导、学生自学和独立编程的操作方式，先由学生查阅与本课程设计有关的书籍和资料，利用visual C++6.0、MFC等设计环境与开发工具完成。

要求如下：

1、分析课程设计题目

2、写出详细设计说明；

3、编写程序代码、调试程序使其能正确运行，能正确地接受输入，产生正确的结果，界面友好，便于操作和使用；

4、课题完成后必须按要求提交课程设计报告，且设计报告必须符合规范。

**【设计报告】**

装订顺序依次为封面、任务书、目录、正文、参考文献、附录。

正文内容包括：系统需求分析，总体设计，详细设计，系统调试，结果分析和总结等。

正文格式如下：

一级标题（左对齐，宋体粗四号），二级标题（左对齐，宋体小四号，加粗）；正文： 小四，宋体，两边分散对齐；行间距1.15倍；左边距：3cm，右边距2cm；

**【提交要求】**

每位同学提交纸质设计报告书一份（用A4纸双面打印），要求编排格式统一、规范，内容充实，并附带电子版(以自己的学号和姓名命名文件名)。

**【设计时间与地点】**

时间：共两周，包括实习动员与任务布置；查阅资料、分析与设计；编写程序，进行调试；成果验收，完成设计报告。

地点：实验室、图书馆等

**【设计成绩评定】**

设计成绩是依据在设计中的表现综合评定，包括：

设计纪律（10%）：设计中遵守纪律，服从管理，以及设计态度等因素，如有严重违纪，可按学校有关规定直接评为不及格。

设计正确性（40%）：根据设计运行正常和异常评定相应等级

设计界面（10%）：根据图形界面评定

设计报告（30%）：设计报告的完成情况

考核形式（10%）：通过提问及验收时间等评定

备注：在实验结果的基础上，发表了学术论文的同学可以直接评为优秀成绩。

**目 录**

1. **系统需求分析**

1.1总体介绍

1.2功能分析

1.2.1 超市系统的创建

1.2.2 添加商品信息

1.2.3 修改商品信息

1.2.4 删除商品信息

1.2.5 查找商品信息

1.2.6 显示商品信息

1.2.7 清空所有信息

1.2.8 退出系统管理的创建

1.3数据分析

1.3.1 框架图展示

1.3.2 设计步骤展示

1. **总体设计**

2.1初始界面展示

2.1.1当商品数量为零时页面展示

2.1.2 当商品数量不为零是页面展示

2.2 Sup主函数信息展示

1. **函数功能介绍**

3.1程序主界面

3.2各功能模块的流程

1. **函数功能介绍**

4.1.1增加商品信息

4.1.2显示商品信息

4.1.3删除商品信息

4.1.4修改商品信息

4.1.5查找商品信息

4.5.1按照姓名查找

4.5.2按照编号进行查找

4.1.6按照商品信息序号进行排序

4.1.7退出管理系统

4.1.8清空所有文档

4.2**各种选项功能介绍**

4.3**结果分析**

1. **总结**

5.1个人收获

5.2个人小结

**超市管理系统**

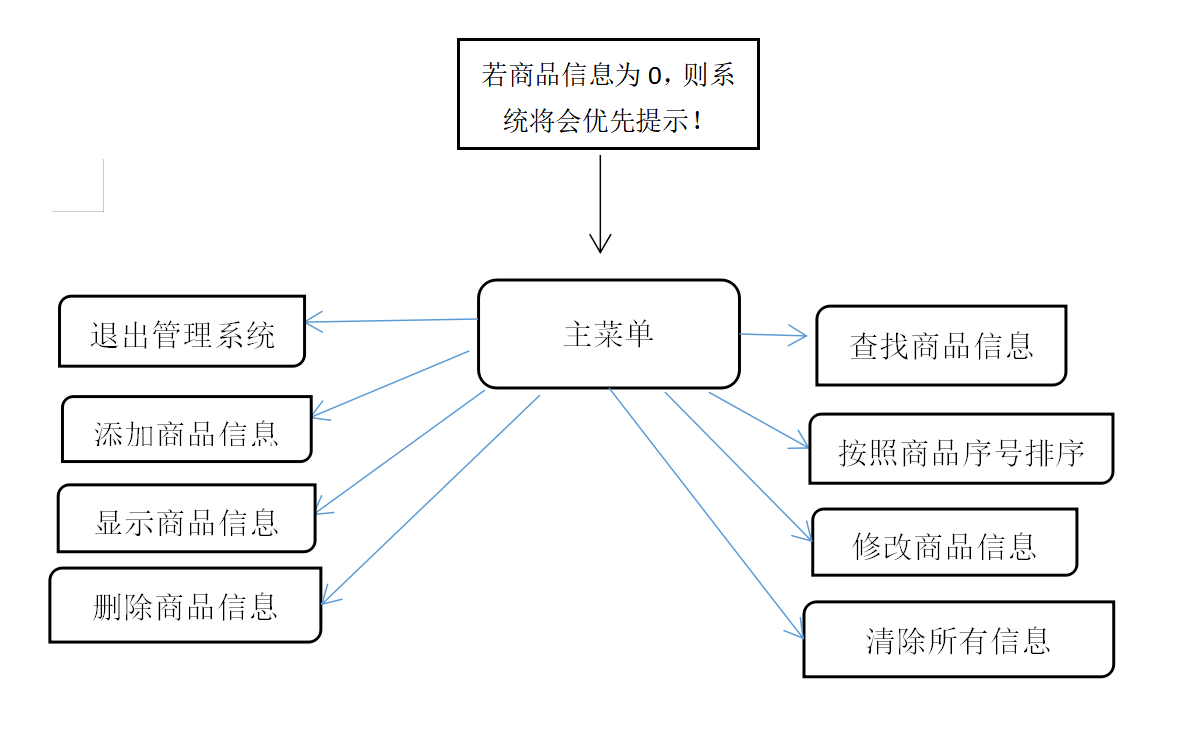
1. **系统需求分析**
2. **总体构思**

网络信息化时代的快速发展，各个大小型超市都需要一个网络数据统计，所以就想利用编程软件做一个超市管理系统。使用系统实现达到增加商品，输出商品，修改商品，删除商品，对商品进行排序，清除所有信息的功能。

1. **功能分析**

 2.1超市系统的创建：系统启动时，从超市功能文件加载超市各商品信息，如果商品数量为零，应添加商品信息；  
 **2.2**添加商品信息：增加新的商品信息到超市信息系统中；  
 **2.3**修改商品信息：修改现有的商品的基本信息;  
 2.4删除商品信息：从商品信息系统中删除一个已经存在的商品信息;  
 2.5查找商品信息：查找商品是否存在。  
 2.6显示商品信息：显示商品信息。  
 2.7清空所有信息：把商品信息全部清空。  
 2.8退出系统管理的创建：关闭正运行的系统回到到超市管理系统 。

1. **数据分析**

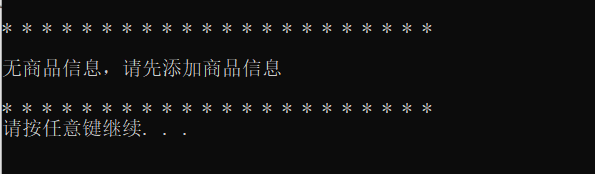
****

1.3.2 设计步骤展示

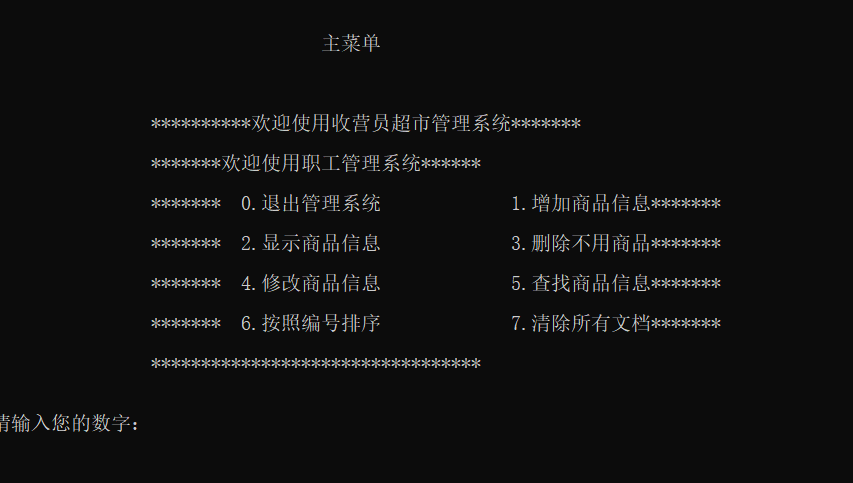
1. 设计一个菜单驱动程序，输入相应内容可以进入相应的选择项目，并执行对应程序。
2. 初始进入若商品数量为零则会提示一个无商品界面。
3. 这个程序用来增加，显示，删除，修改，查找，排序，清除商品信息，主要信息为商品编号、商品名、商品种类。
4. 商品信息可以按照食物类、饮料类、百货类归类显示，通过TXT（记事本）将信息录入和显示这样可以让信息保存时间更长，最后可以通过按序号（0）退出管理系统，也可以通过（7）清除所有信息。
5. **总体设计**

2.1初始界面展示

2.1.1当商品数量为零时页面展示



2.1.2 当商品数量不为零是页面展示

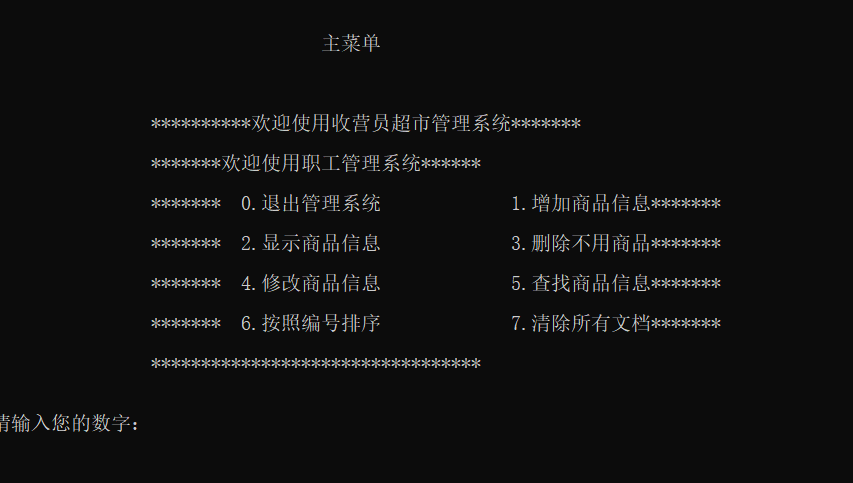
****

2.2 Sup主函数信息展示



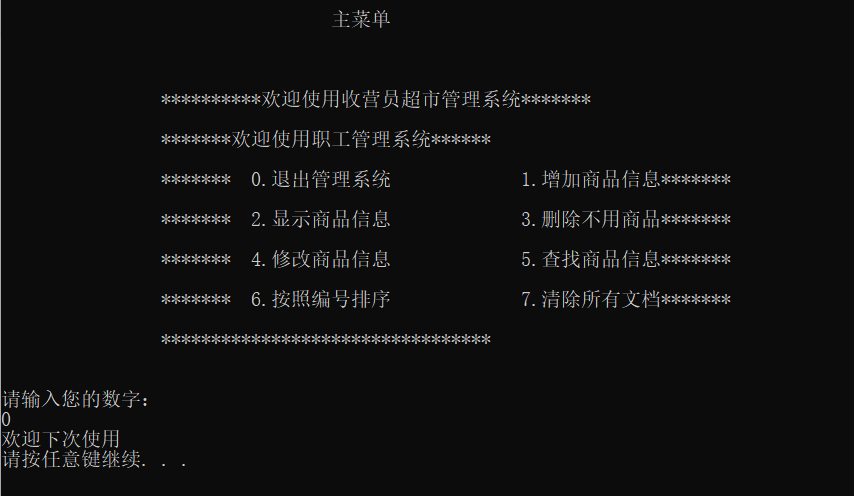
1. 函数功能介绍

3.1程序主界面

****

3.2各功能模块的流程

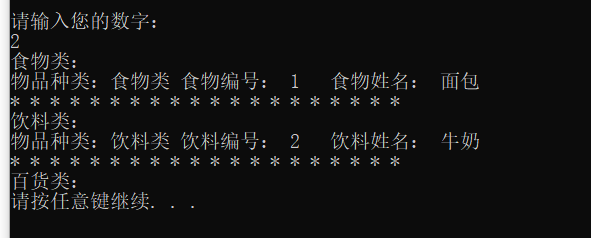
3.2.1退出管理系统



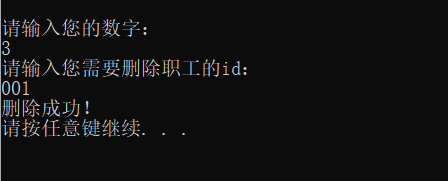
3.2.2增加商品信息



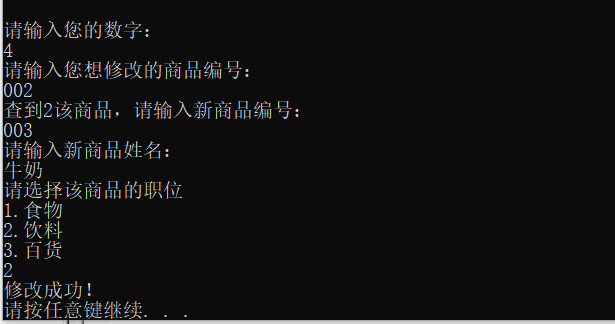
3.2.3显示商品信息



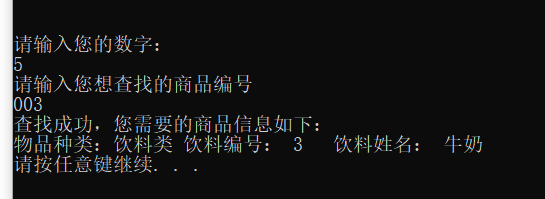
3.2.4删除不用商品



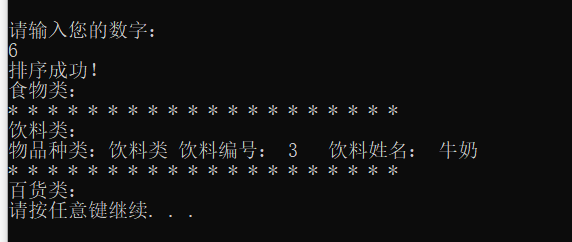
3.2.5修改商品信息



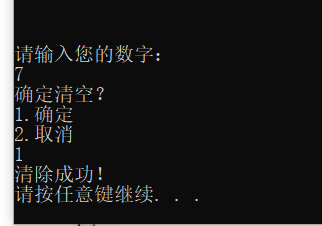
3.2.6查找商品信息



3.2.7按序号排序



3.2.7清除所有信息



1. **函数功能介绍**
2. ****显示函数：显示保存的数据****
3. ****添加函数：此函数实现的功能主要是，添加一个商品的信息，也就是说调用一次这个函数只能添加一个商品的信息。在此函数中也使用文件处理的方式，但使用的是以追加的方式打开文件，使输入的商品信息写在文件里原有信息的后面。****
4. ****删除函数：此函数实现的功能主要是，删除一个商品的信息。在此函数中，先以读的方式打开文件，读出文件里的信息在显示在屏幕上，用户再根据提示信息输入想要删除第几个商品的信息，关闭文件。再以写的方式打开文件，以一个if条件语句控制，再重新将商品信息写入文件（除了打算删除的那个学生的信息），由于是以写的方式打开的文件，所以文件里原有的输入就都没有了，存入的新的数据。****
5. ****修改函数：此函数实现的功能主要是，修改一个学生的信息。此函数的设计方法和删除函数的设计思想相类似，应用的思想都是将学生信息重新写入文件的方法，将原有数据覆盖。****
6. ****查询函数：在查询功能中，可以按多条件查询，本系统分为两个查询条件，即按学号查询和按姓名查询。使用if语句来选择查询条件。****

****5.1 按编号查询****

****根据提示信息输入编号后，然后使用一个for循环语句，从第一个商品开始检索，如果该商品的编号和输入的编号相等，则输出该商品的信息，由于商品的编号是唯一固定的，所以并使用break语句跳出循环。****

****当无查询结果时，会输出一条提示信息“无此商品信息！” 。****

****5.2 按姓名查询****

****根据提示信息输入姓名后，然后使用一个for循环语句，从第一个商品开始检索，如果该商品的名和输入的名一样，则输出该商品的信息，由于商品的姓名有可能重复出现，所以不能使用break语句跳出循环，应该从第一个商品开始检索直到最后一个商品。****

****当无查询结果时，会输出一条提示信息“无此商品信息！” 。****

1. ****显示函数：此函数实现的功能主要是，显示所有学生的信息。先是从文件中读取学生信息，然后在输出到显示窗口中。用一个for循环语句，来把所有学生的信息都输出。****
2. ****退出系统：当输入“0”时，就能退出系统，这是通过while语句来控制实现的。****

4.2各种选项功能介绍

1. （0） 则按任意键即可退出该系统；
2. （1） 先输入商品编号，再输入商品名，再选择商品种类，则录入成功；
3. （2） 显示所在商品，若没有则无该商品信息
4. （3） 删除所显示的商品，若没有则无法删除
5. （4） 修改所拥有的商品按照编号进行修改，若无则无法找到
6. （5） 按照编号找到该商品
7. （6） 按照编号排序
8. （7） 清除所有信息

4.3结果分析

根据菜单提供的选项逐步操作，一项一项地实现目的，简洁明了。系统流程清晰可见。可根据此系统达成管理学生信息的目的。

5.**总结**

5.1个人收获

1. **C++很难，似乎是学不完的，你觉得自己了解一点，是总会发现还有很多自己所不懂的东西，还有很多自己所没探知的领域，很多知识运用到实践中很有一定的难度，看着自己和同学的程序运行结果，突然发现C++真的很神奇，居然能运行出那么漂亮的系统，看着运行结果**
2. **一个大的程序是由许多小的程序组成的，总的程序就是对这些小的程序的调用，来实现各项功能。只要处理好这些小的程序，做出一个大的程序并不是很困难。**
3. **根据C++课程所学的概念、理论和方法，按照C++程序设计的基本步骤，设计出一个适当规模的程序；进一步加深对C++语言的理解和掌握。理论联系实际，加深和巩固所学的理论知识，提高实践能力和计算机的综合运用能力。我们编写程序的过程是辛苦与快乐的，程序的编写原则很重要，只要我们在编程，就必须不断改进，才能更好提高编程能力。**

5.2个人小结

1.这个课程设计的主题都是自己逐步调试功能并运行出来的，然后有联想到超市商品，最后决定写一个城市管理系统，在网上参考了CSDN网站和billbill软件用来借鉴。

2.在最开始的时候就要整体的布局，系统有那几个功能，需要那些函数的使用。我初步确定的是五个操作大体，每个购物信息包含五个子信息。五个主体操作是 1.)增加商品信息2.)显示商品信息 3.)修改商品信息 4.)删除商品信息 5.)查找商品信息 6.）退出系统。首先我想到的是对数据的输出操作，感觉这个应该会比较简单，然后在做的时候发现了很多问题，字符串的读取开始准备用string实现的，但用的时候发现了很多问题，最后没办法，只有选择用字符串数组来实现数据的处理。在前面的主题布局完场以后，现在开始实现每一个比较小的共能模板块，我要商品信息这个板块比较简单，很快就做好，输出信息这个模板块是使用指针实现的，在后面基本上是把CSDN的一些类容拿来借鉴使用的，定位，查找，这些都比较难，然后都是借鉴的，在修改信息的时候就用到了定位，查找功能的函数，这个只是对内容的修改，这个最后完成的还是比较快的。主要是在后面一个删除信息的，因为我在开始用的是指针定位，与查找，但在用到删除信息的时候，就不知道该怎么用了。然后又把师生管理系统拿来看，因为里面有删除师生，看时候发现里面是用数组进行存储，然后再删除的时候，数组下标实现前移，这样就掩盖了要删除的信息，这样的方法让我没办法，我最后准备将所有的数据进行数组存储的时候，发现了很多不能修改的问题，没办法，就在网上找资料，看有什么好的办法是删除文件夹的数据的，最后还是没找到，最后，在实在没办法的时候，就想到了替换，将那些要删除的信息。这就是这个课程设计最大的缺憾，因为时间用了很长，两条就为了这个系统的设计。不过最后的整体功能还是实现了，基本的要求也达到了。

参考文献

[1] 杜青、解芳、李春颖、王丹华.C++面向对象程序设计教程.2017

[2]史蒂芬 普拉达.C++ Primer plus.2015

[3] CSDN网站和billbill软件

附录

# include <iostream>

# include<string>

# include<fstream>

# define FILENAME "supFile.txt"

using namespace std;

class Sup;

class SupManager

{

public:

friend Sup;

SupManager();

void ShowMenu(); //菜单栏

void Exitsystem(); //退出系统

void Add\_Emp(); //增加商品

void Save(); //用来保存文件

int get\_EmpNum(); //统计商品数量

bool m\_FileIsEmpty; //用来判断文件是否为空

void init\_Emp(); //初始化商品

void Show\_Emp(); //显示商品

void Del\_Emp(); //删除商品

int IsExist(int id); //判断商品是否存在

void Mod\_Emp(); //修改员工信息

void Find\_Emp(); //查找商品

void Sort\_Emp();//对商品进行排序

void Clean\_File();//清空所有内容

~SupManager();

int m\_EmpNum; //记录原始商品人数

Sup \*\* m\_EmpArray; //两个\*\*是可以记录动态数组的大小

};

class Sup //制作一个商品的类 运用多态的方法

{

public:

virtual void ShowInfo() = 0; //显示商品信息

virtual string getDeptName() = 0; //显示商品名

int m\_ID; //商品数量

string m\_Name; //商品名

int m\_depID; //商品号码

};

class Food:public Sup,public SupManager

{

public:

Food(int id, string name, int did);

virtual void ShowInfo();

virtual string getDeptName();

};

class Drink:public Sup

{

public:

Drink(int id, string name, int did);

virtual void ShowInfo();

virtual string getDeptName();

};

class Goods:public Sup

{

public:

Goods(int id, string name, int did);

virtual void ShowInfo();

virtual string getDeptName();

};

void Food::ShowInfo()

{

cout<<"物品种类："<<this->getDeptName()

<<" 食物编号： "<<this->m\_ID

<<"\t食物姓名： "<<this->m\_Name

<<endl;

}

void Drink::ShowInfo()

{

cout<<"物品种类："<<this->getDeptName()

<<" 饮料编号： "<<this->m\_ID

<<"\t饮料姓名： "<<this->m\_Name

<<endl;

}

void Goods::ShowInfo()

{

cout<<"物品种类："<<this->getDeptName()

<<" 百货编号： "<<this->m\_ID

<<"\t百货姓名： "<<this->m\_Name

<<endl;

}

string Food::getDeptName()

{

return "食物类";

}

string Drink::getDeptName()

{

return "饮料类";

}

string Goods::getDeptName()

{

return "百货";

}

Food::Food(int id, string name, int did)

{

this->m\_ID = id;

this->m\_Name = name;

this->m\_depID = did;

}

Drink::Drink(int id, string name, int did)

{

this->m\_ID = id;

this->m\_Name = name;

this->m\_depID = did;

}

Goods::Goods(int id, string name, int did)

{

this->m\_ID = id;

this->m\_Name = name;

this->m\_depID = did;

}

void SupManager::ShowMenu()

{

if(this->m\_FileIsEmpty)

{

cout<<endl<<endl<<endl;

cout<<"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*"<<endl<<endl;

cout<<"无商品信息，请先添加商品信息"<<endl<<endl;

cout<<"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*"<<endl;

system("pause");

system("cls");

}

cout<<"\n\n\t\t\t\t 主菜单"<<endl<<endl<<endl ;

cout<<"\n\t\t\*\*\*\*\*\*\*\*\*\*欢迎使用收营员超市管理系统\*\*\*\*\*\*\* "<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\*欢迎使用职工管理系统\*\*\*\*\*\*"<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\* 0.退出管理系统 1.增加商品信息\*\*\*\*\*\*\*"<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\* 2.显示商品信息 3.删除不用商品\*\*\*\*\*\*\*"<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\* 4.修改商品信息 5.查找商品信息\*\*\*\*\*\*\*"<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\* 6.按照编号排序 7.清除所有文档\*\*\*\*\*\*\*"<<endl;

cout<<"\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl<<endl;

}

void SupManager::Exitsystem()

{

cout<<"欢迎下次使用"<<endl;

system("pause");

exit(0);

}

void SupManager::Add\_Emp()

{

int addnum = 1;

if(addnum > 0)

{

int newsize = this->m\_EmpNum + addnum; //记录空间大小

Sup \*\* newspace = new Sup\*[newsize]; //开辟新空间

if(m\_EmpNum != 0) //将原有数据记录下来

{

for(int i=0; i<this->m\_EmpNum; ++i)

{

newspace[i] = m\_EmpArray[i];

}

}

for(int i=0; i<addnum; ++i)

{

int id;

string name;

int dselsct = 0;

cout<<"请输入新商品编号："<<endl;

cin>>id;

cout<<"请输入新商品名："<<endl;

cin>>name;

cout<<"请选择该商品的种类"<<endl;

cout<<"1.食物"<<endl;

cout<<"2.饮料"<<endl;

cout<<"3.百货"<<endl;

cin>>dselsct;

Sup \* sup = NULL;

switch(dselsct)

{

case 1:sup = new Food(id,name,1);

break;

case 2:sup = new Drink(id,name,2);

break;

case 3:sup = new Goods(id,name,3);

break;

default:

break;

}

newspace[this->m\_EmpNum+i] = sup; //添加新商品

delete[]this->m\_EmpArray; //释放原有空间

this->m\_EmpNum = newsize;

this->m\_EmpArray = newspace;

}

cout<<"成功添加该新商品"<<endl;

this->m\_FileIsEmpty = false;

}

else

{

cout<<"添加数据有误，请重新输入！"<<endl;

}

system("pause");

system("cls");

}

void SupManager:: Save()

{

ofstream ofs;

ofs.open(FILENAME,ios::out);

for(int i=0; i<this->m\_EmpNum; ++i)

{

ofs<< this->m\_EmpArray[i]->m\_ID<<" "

<<this->m\_EmpArray[i]->m\_Name<<" "

<<this->m\_EmpArray[i]->getDeptName()<<endl;

}

ofs.close();

}

SupManager::SupManager()

{

ifstream ifs;

ifs.open(FILENAME, ios::in);

//1.当文件不存在的时候

if( !ifs.is\_open() )

{

this->m\_EmpArray = NULL;

this->m\_EmpNum = 0;

this->m\_FileIsEmpty = true;

ifs.close();

return; //这个很重要如果执行了1后面的将不会再执行了

}

//2.当文件为空的时候

char ch;

ifs>>ch;

if( ifs.eof() ) //ifs.eof() 当文件为空返回true，认为文件读完了

{

this->m\_EmpArray = NULL;

this->m\_EmpNum = 0;

this->m\_FileIsEmpty = true;

ifs.close();

return;

}

//3.当文件存在且不为空时

this->m\_FileIsEmpty = false;

int num = this->get\_EmpNum();

cout<<"商品的数量为："<<num<<endl;

this->m\_EmpNum = num;

this->m\_EmpArray = new Sup\*[this->m\_EmpNum];

this->init\_Emp();

}

SupManager::~SupManager()

{

if(this->m\_EmpArray != NULL)

{

delete[] this->m\_EmpArray;

this->m\_EmpArray = NULL;

}

}

int SupManager::get\_EmpNum()

{

ifstream ifs;

ifs.open(FILENAME,ios::in);

int id;

string name;

int did;

int num = 0;

while(ifs>>id && ifs>>name && ifs>>did)

{

num++;

}

ifs.close();

return num;

}

void SupManager::init\_Emp()

{

ifstream ifs;

ifs.open(FILENAME, ios::in);

int id;

string name;

int did;

int index = 0;

while(ifs>>id && ifs>>name && ifs>>did)

{

Sup \* sup = NULL;

if(did = 1)

{

sup = new Food(id,name,did);

}

else if(did = 2)

{

sup = new Drink(id,name,did);

}

else if(did = 3)

{

sup = new Goods(id,name,did);

}

this->m\_EmpArray[index] = sup;

index++;

}

ifs.close();

}

void SupManager::Show\_Emp()

{

if(this->m\_FileIsEmpty)

{

cout<<"文件不存在或者为空！"<<endl;

}

else

{

cout<<"食物类："<<endl;

for(int i=0; i<this->m\_EmpNum; i++)

{

if(this->m\_EmpArray[i]->getDeptName() == "食物类")

{

this->m\_EmpArray[i]->ShowInfo();

}

}

cout<<"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*"<<endl;

cout<<"饮料类："<<endl;

for(i=0; i<this->m\_EmpNum; i++)

{

if(this->m\_EmpArray[i]->getDeptName() == "饮料类")

{

this->m\_EmpArray[i]->ShowInfo();

}

}

cout<<"\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*"<<endl;

cout<<"百货类："<<endl;

for(i=0; i<this->m\_EmpNum; i++)

{

if(this->m\_EmpArray[i]->getDeptName() == "百货类")

{

this->m\_EmpArray[i]->ShowInfo();

}

}

}

system("pause");

system("cls");

}

void SupManager::Del\_Emp()

{

if(this->m\_FileIsEmpty)

{

cout<<"文件不存在或者文件为空!"<<endl;

}

else

{

cout<<"请输入您需要删除职工的id："<<endl;

int id = 0;

cin>>id;

int index = this->IsExist(id);

if(index != -1)

{

for(int i=index; i<this->m\_EmpNum-1; ++i)

{

this->m\_EmpArray[i] = this->m\_EmpArray[i+1];

}

this->m\_EmpNum--;

this->Save();

cout<<"删除成功！"<<endl;

}

else

{

cout<<"删除失败，未找到该商品！"<<endl;

}

}

system("pause");

system("cls");

}

int SupManager::IsExist(int id)

{

int index = -1;

for(int i=0; i<this->m\_EmpNum; ++i)

{

if(this->m\_EmpArray[i]->m\_ID == id)

{

index = i;

break;

}

}

return index;

}

void SupManager::Mod\_Emp()

{

if(this->m\_FileIsEmpty)

{

cout<<"商品不存在或者为空"<<endl;

}

else

{

cout<<"请输入您想修改的商品编号："<<endl;

int id = 0;

cin>>id;

int ret = this->IsExist(id);

if(ret != -1)

{

delete this->m\_EmpArray[ret];

int newid = 0;

string newname = "";

int dselect = 0;

cout<<"查到"<<id<<"该商品，请输入新商品编号："<<endl;

cin>>newid;

cout<<"请输入新商品姓名："<<endl;

cin>>newname;

cout<<"请选择该商品的职位"<<endl;

cout<<"1.食物"<<endl;

cout<<"2.饮料"<<endl;

cout<<"3.百货"<<endl;

cin>>dselect;

Sup \* sup = NULL;

switch(dselect)

{

case 1:

sup = new Food(newid, newname, dselect);

break;

case 2:

sup = new Drink(newid, newname, dselect);

break;

case 3:

sup = new Goods(newid, newname, dselect);

break;

default:

break;

}

this->m\_EmpArray[ret] = sup;

cout<<"修改成功！"<<endl;

this->Save();

}

else

{

cout<<"修改失败，无法找到该商品！"<<endl;

}

}

system("pause");

system("cls");

}

void SupManager::Find\_Emp()

{

if(this->m\_FileIsEmpty)

{

cout<<"文件为空或不存在！"<<endl;

}

else

{

int id = 0;

cout<<"请输入您想查找的商品编号"<<endl;

cin>>id;

int ret = this->IsExist(id);

if( ret != -1 )

{

cout<<"查找成功，您需要的商品信息如下："<<endl;

this->m\_EmpArray[ret]->ShowInfo();

}

else

{

cout<<"查找失败，无法找到该商品！"<<endl;

}

}

system("pause");

system("cls");

}

void SupManager::Sort\_Emp()

{

if(this->m\_FileIsEmpty)

{

cout<<"文件为空或不存在！"<<endl;

}

else

{

for(int i=0; i<this->m\_EmpNum; ++i)

{

int num = i;

for(int j=i+1; j<this->m\_EmpNum; ++j)

{

if(this->m\_EmpArray[num]->m\_ID > this->m\_EmpArray[j]->m\_ID)

{

num = j;

}

}

}

cout<<"排序成功！"<<endl;

this->Save();

this->Show\_Emp();

}

system("pause");

system("cls");

}

void SupManager::Clean\_File()

{

cout<<"确定清空？"<<endl;

cout<<"1.确定"<<endl;

cout<<"2.取消"<<endl;

int dselect = 0;

cin>>dselect;

if(dselect = 1)

{

ofstream ofs(FILENAME, ios::trunc); //trunc如果存在删除文件并重新创建

ofs.close();

if(this->m\_EmpArray != NULL)

{

for(int i=0;i<this->m\_EmpNum; ++i)

{

if(this->m\_EmpArray[i] != NULL)

{

delete this->m\_EmpArray[i];

}

}

this->m\_EmpNum = 0;

delete[]this->m\_EmpArray;

this->m\_EmpArray = NULL;

this->m\_FileIsEmpty = true;

}

cout<<"清除成功！"<<endl;

}

system("pause");

system("cls");

}

int main()

{

SupManager wm;

int choice = 0;

while( true )

{

wm.ShowMenu();

cout<<"请输入您的数字："<<endl;

cin>>choice;

switch(choice)

{

case 0: wm.Exitsystem();

break;

case 1: wm.Add\_Emp();

break;

case 2: wm.Show\_Emp();

break;

case 3: wm.Del\_Emp();

break;

case 4: wm.Mod\_Emp();

break;

case 5: wm.Find\_Emp();

break;

case 6: wm.Sort\_Emp();

break;

case 7: wm.Clean\_File();

break;

default:

system("cls"); //清屏

break;

}

}

return 0;

}

|  |
| --- |
| **成绩及评语：**      指导教师（签字）：    系主任（签字）：  2020年 月 日 |