C/C++程序设计

课程设计报告书

题目：商品销售系统

院 系：

专业班级：

学 号：

姓 名：

年 月 日

一、课程设计的任务及要求

|  |  |
| --- | --- |
| 题目13 | 结构数组的应用（可以自己设计功能）（管理类） |
| 设计任务 | 模拟一个商店的商品销售管理系统（P209例题7-17） |
| 设计要求 | 1. 库存表数据录入 2. 销售表数据录入 3. 销售商品 4. 库存表查询 5. 销售表查询 6. 退出 |

二、课程设计主要数据和函数介绍

主要数据：commodity 商品物品，包含id、姓名、价格、库存量、销售量等属性

主要数据：commodity\_list 商品物品列表，同时可以作为库存表和销售表。

函数：display\_have 展示库存表的商品

函数：display\_sale 展示销售表的商品

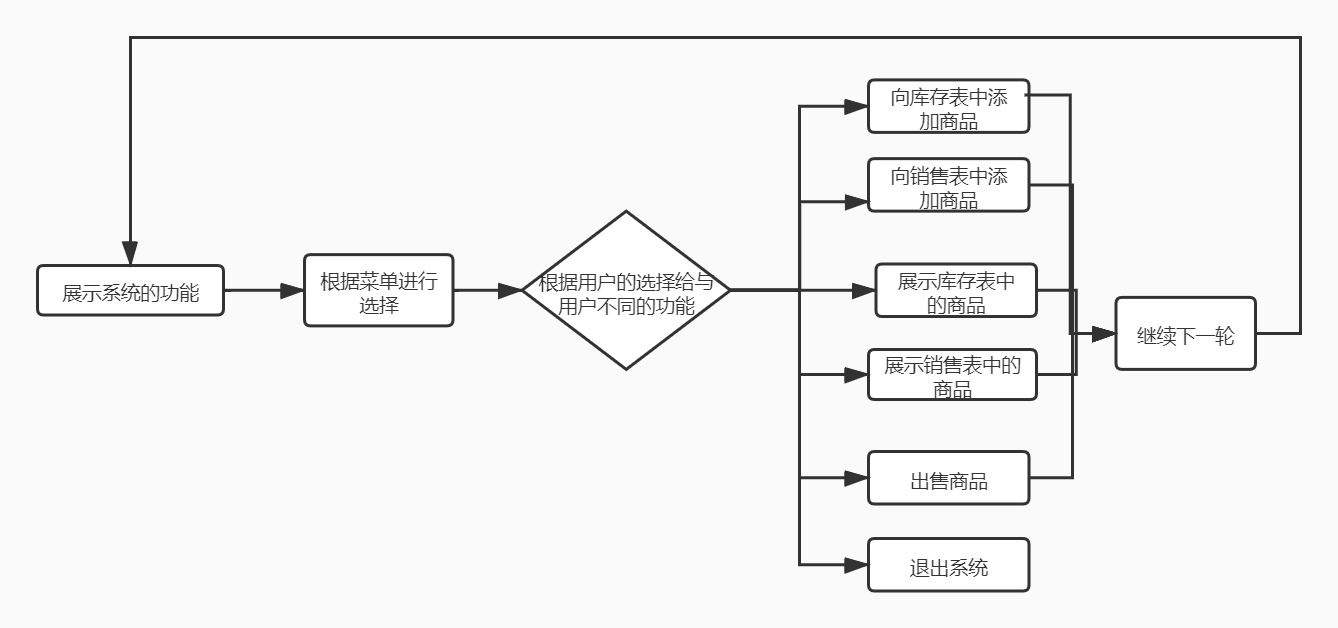
函数：add\_have 向库存表添加商品

函数：add\_sale 向销售表添加商品

函数：buyed 出售商品

三、课程设计的问题分析

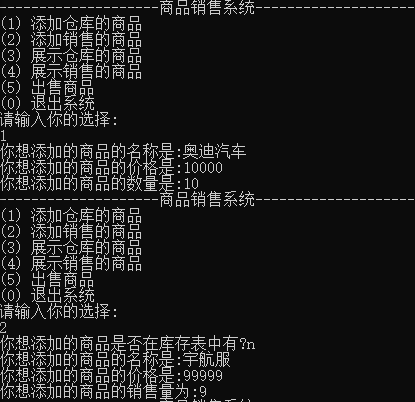
流程图如下：



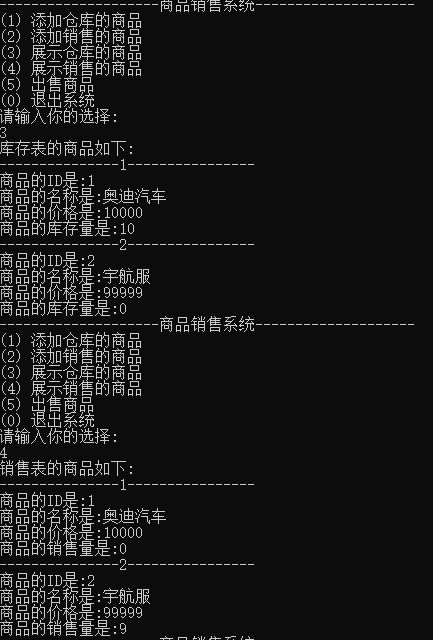
本系统使用1个表来实现系统中的库存表和销售表，使用不同的展示形式来完成不同表的展示，这是更接近于实际生活。然后使用5个函数将本系统拆分成5个模块，这样更加的易于管理。其实向销售表添加商品比较麻烦，首先先判断仓库中是否有这个商品，如果有则直接对商品的销售量进行修改，如果没有则直接添加一个新商品，其库存量为0。在出售商品的时候比较简单，可以直接根据商品的ID，进行库存量-1，商品量+1的操作。

1. 源程序代码及运行结果

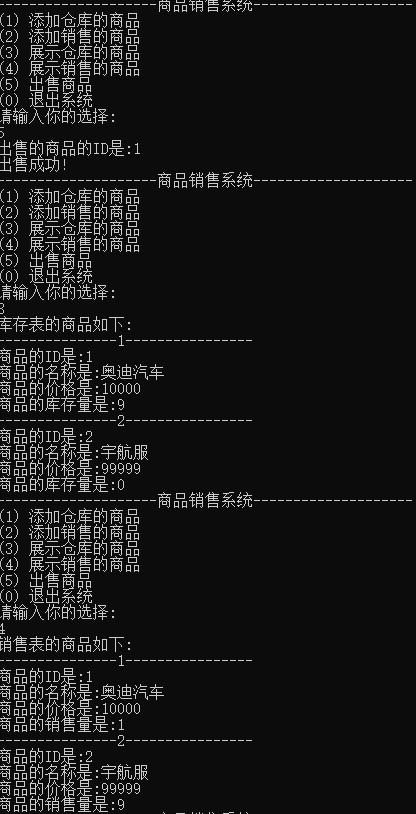
**运行结果：**



首先我们先对仓库和销售表添加奥迪汽车和宇航服两件商品。



接下来我们查看仓库和销售表的信息。可以看到奥迪汽车库存10个，销售量0个。宇航服库存0个，销售9个。



最后我们进行商品的出售，我们可以看到奥迪汽车的库存量减一，销售量加一。

**程序源代码：head.h文件**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

typedef struct T{

int id;

char name[100]; // 商品名称

int money; // 商品价格

int have\_number; // 商品有数量

int sale\_number; // 商品销售数量

}commodity;

commodity commodity\_list[100];

int head=1;

int i,j;

// 展示仓库的商品

void display\_have();

// 展示销售的商品

void display\_sale();

// 添加仓库的商品

void add\_have();

// 添加销售的商品

void add\_sale();

// 出售商品

void buyed();

void add\_have(){

commodity t;

printf("你想添加的商品的名称是:");

scanf("%s",t.name);

printf("你想添加的商品的价格是:");

scanf("%d",&t.money);

printf("你想添加的商品的数量是:");

scanf("%d",&t.have\_number);

t.sale\_number = 0;

t.id = head;

commodity\_list[head-1] = t;

head++;

}

void add\_sale(){

char c;

char t[100];

int data;

printf("你想添加的商品是否在库存表中有?");

scanf("%c",&c);

scanf("%c",&c);

if(c == 'y'){

printf("你要录入的商品名称为?");

scanf("%s",t);

for(i=0;i<head-1;i++){

for(j=0;j<strlen(t);j++){

if(t[j] != commodity\_list[i].name[j]) return;

}

printf("你要将该商品的销售量改为?");

scanf("%d",&data);

commodity\_list[i].sale\_number = data;

}

}else{

commodity tdata;

printf("你想添加的商品的名称是:");

scanf("%s",tdata.name);

printf("你想添加的商品的价格是:");

scanf("%d",&tdata.money);

printf("你想添加的商品的销售量为:");

scanf("%d",&tdata.sale\_number);

tdata.have\_number = 0;

tdata.id = head;

commodity\_list[head-1] = tdata;

head++;

}

}

// 展示仓库的商品

void display\_have(){

printf("库存表的商品如下:\n");

for(i=0;i<head-1;i++){

printf("---------------%d----------------\n",(i+1));

printf("商品的ID是:%d\n",commodity\_list[i].id);

printf("商品的名称是:%s\n",commodity\_list[i].name);

printf("商品的价格是:%d\n",commodity\_list[i].money);

printf("商品的库存量是:%d\n",commodity\_list[i].have\_number);

}

}

// 展示销售表的商品

void display\_sale(){

printf("销售表的商品如下:\n");

for(i=0;i<head-1;i++){

printf("---------------%d----------------\n",(i+1));

printf("商品的ID是:%d\n",commodity\_list[i].id);

printf("商品的名称是:%s\n",commodity\_list[i].name);

printf("商品的价格是:%d\n",commodity\_list[i].money);

printf("商品的销售量是:%d\n",commodity\_list[i].sale\_number);

}

}

// 出售商品

void buyed(){

int id;

printf("出售的商品的ID是:");

scanf("%d",&id);

if( commodity\_list[id-1].have\_number > 0 ){

commodity\_list[id-1].have\_number -= 1;

commodity\_list[id-1].sale\_number += 1;

printf("出售成功!\n");

return;

}

printf("出售失败!\n");

}

**程序源代码：main.c文件**

#include <stdio.h>

#include <stdlib.h>

#include "head.h"

int main()

{

int choose = 0;

while(choose !=6){

printf("--------------------商品销售系统--------------------\n");

printf("(1) 添加仓库的商品\n");

printf("(2) 添加销售的商品\n");

printf("(3) 展示仓库的商品\n");

printf("(4) 展示销售的商品\n");

printf("(5) 出售商品\n");

printf("(0) 退出系统\n");

printf("请输入你的选择:\n");

scanf("%d",&choose);

switch(choose){

case 1:add\_have(); break;

case 2:add\_sale(); break;

case 3:display\_have(); break;

case 4:display\_sale(); break;

case 5:buyed(); break;

case 6:exit(0);

}

}

return 0;

}

五、课程设计收获及体会

在本次课设，我成功搭建了一个完整的系统，在搭建的过程中，我明白了C语言是一个面向过程的语言，所以我在实现中，尽量的使系统更加的模块化，例如头文件和源文件的分类，在头文件中尽量将函数在开始先进行声明，然后在下面在进行实现。这样使代码更加的规范。这次课设对我来说是一个大工程，让我学到了许多。