郑 州 轻 工 业 大 学

**实 验 报 告**

**课程名称： 面向对象程序设计**

**姓 名： 原彬贺**

**学 号： 542001020223**

**专业班级： 计算机科学与技术20-02**

**任课教师： 王岩**

**2021 年 11 月 3 日**

**实验报告正文**

**实验五**

**文件操作**

**一、 实验目的**

1. 掌握文本文件的输入/输出方法；

2. 掌握二进制文件的输入/输出方法；

3. 掌握控制输出宽度的函数使用方法；

4. 掌握设置进制输出的方法。**二、 问题的本质和抽象描述**

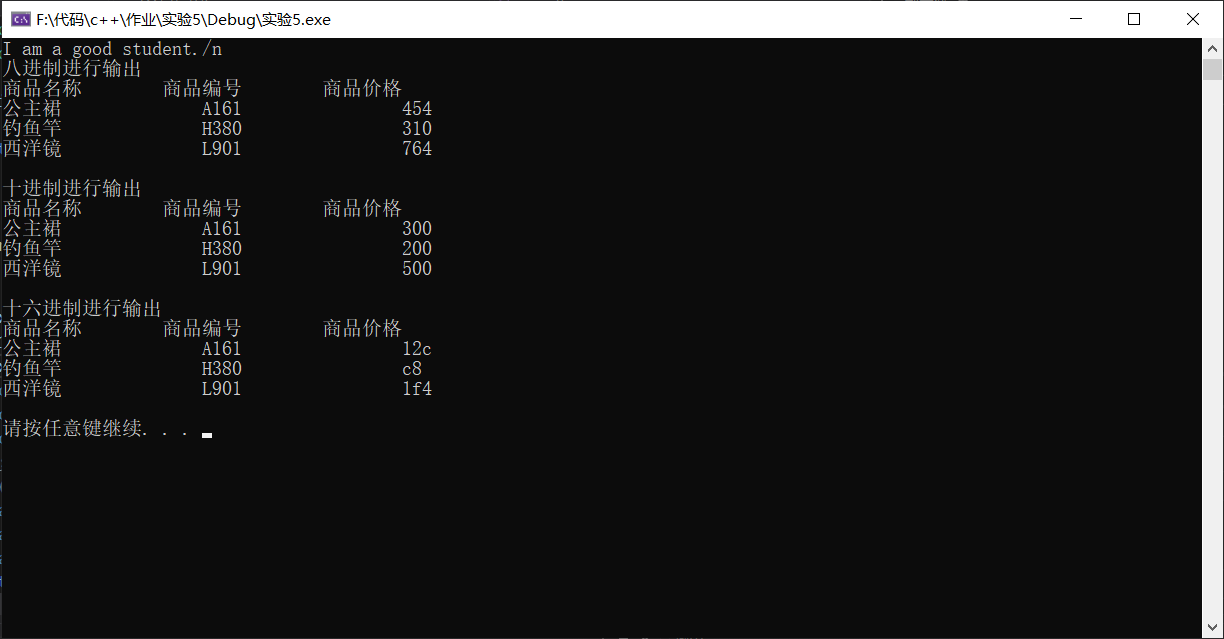
将一个类的数据通过文件操作输出到一个文件，并再使用文件输入操作，将文件中的数据分别按八进制，十进制，十六进制输出到显示屏。

**三、 测试**

**1.方案**

先通过定义一个GoodClass类，里面存入商品信息，再通过文件输出操作，将三个商品的信息传入文件good.dat中。再定义一个商品类的数组，将文件中的数据输入到该数组，通过类的输出函数，将其分别按八进制，十进制，十六进制输出到显示器。

**2.结果**



**四、 总结与讨论**

在进行实验时，对于直接定义GoodClass类型的数组时，由于未定义无参构造函数，导致无法进行直接定义，通过添加一个无参构造函数，使其可以直接进无参数的构造。在进行输出时，通过翻阅书籍，使得数据可以按宽度流输出。

**附：源代码**

#include<iostream>

#include<string>

#include<iomanip>

#include<fstream>

using namespace std;

class GoodClass

{

private:

string Name;

string Num;

int Price;

public:

GoodClass(string name, string num, int price)

{

Name = name;

Num = num;

Price = price;

}

GoodClass()

{

Name = "";

Num = "";

Price = 0;

}

void decshow()

{

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout << Name;

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout << Num;

cout << dec << Price << endl;

}

void hexshow()

{

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout << Name;

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout << Num;

cout << hex << Price << endl;

}

void octshow()

{

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout << Name;

cout.width(20);

cout.setf(ios::left);

cout.fill(' ');

cout<< Num;

cout<<oct << Price << endl;

}

~GoodClass()

{}

};

int main()

{

GoodClass lin[3];

lin[0]=GoodClass("公主裙", "A161", 300);

lin[1] = GoodClass("钓鱼竿", "H380", 200);

lin[2] = GoodClass("西洋镜", "L901", 500);

GoodClass list[3];

ofstream out("good.dat", ios::binary);

if (!out)

{

cout << "can not open the file." << endl;

abort();

}

string text = "I am a good student./n";

string index = "商品名称\t商品编号\t商品价格\t";

string text1="";

string index1 = "";

out.write((char\*)&text, sizeof(text));

out.write((char\*)&index, sizeof(index));

out.write((char\*)lin, 4 \* sizeof(lin[0]));

out.close();

ifstream in("good.dat", ios::binary);

in.read((char\*)&text1, sizeof(text));

in.read((char\*)&index1, sizeof(index));

in.read((char\*)list,4\*sizeof(GoodClass));

cout << text1 << endl;

cout << "八进制进行输出" << endl;

cout << index1 << endl;

for (int i = 0; i < 3; i++)

{

list[i].octshow();

}

cout << endl;

cout << "十进制进行输出" << endl;

cout << index1 << endl;

for (int i = 0; i < 3; i++)

{

list[i].decshow();

}

cout << endl;

cout << "十六进制进行输出" << endl;

cout << index1 << endl;

for (int i = 0; i < 3; i++)

{

list[i].hexshow();

}

cout << endl;

system("pause");

return 0;

}