**实验名称 实验3. 怎样使用类与对象 优**

**一、实验内容及结果**

**1、定义一个Point类，计算给定的两个点p1和p2之间的距离以及两个点分别到原点的距离，要求对象的初始化通过构造函数实现。**

**源代码：#include <iostream>**

**#include<math.h>**

**using namespace std;**

**class point{**

**public:**

**point(int,int,int,int);**

**void distance() ;**

**private:**

**int x1,y1,x2,y2; //point类是点的共性，一个点类中怎么可以有两个点的坐标？应该设计的类，无论几个点都适用**

**};**

**point::point(int X1,int Y1,int X2,int Y2){**

**x1=X1;**

**y1=Y1;**

**x2=X2;**

**y2=Y2;**

**}**

**void point::distance(){//你这还是用的C的思路了**

**float a,b,c;**

**a=sqrt(x1\*x1+y1\*y1);**

**b=sqrt(x2\*x2+y2\*y2);**

**c=sqrt((x1-x2)\*(x1-x2)+(y1-y2)\*(y1-y2));**

**cout<<"p1到原点的距离："<<a<<endl;**

**cout<<"p2到原点的距离："<<b<<endl;**

**cout<<"p1和p2之间的距离："<<c<<endl;**

**}**

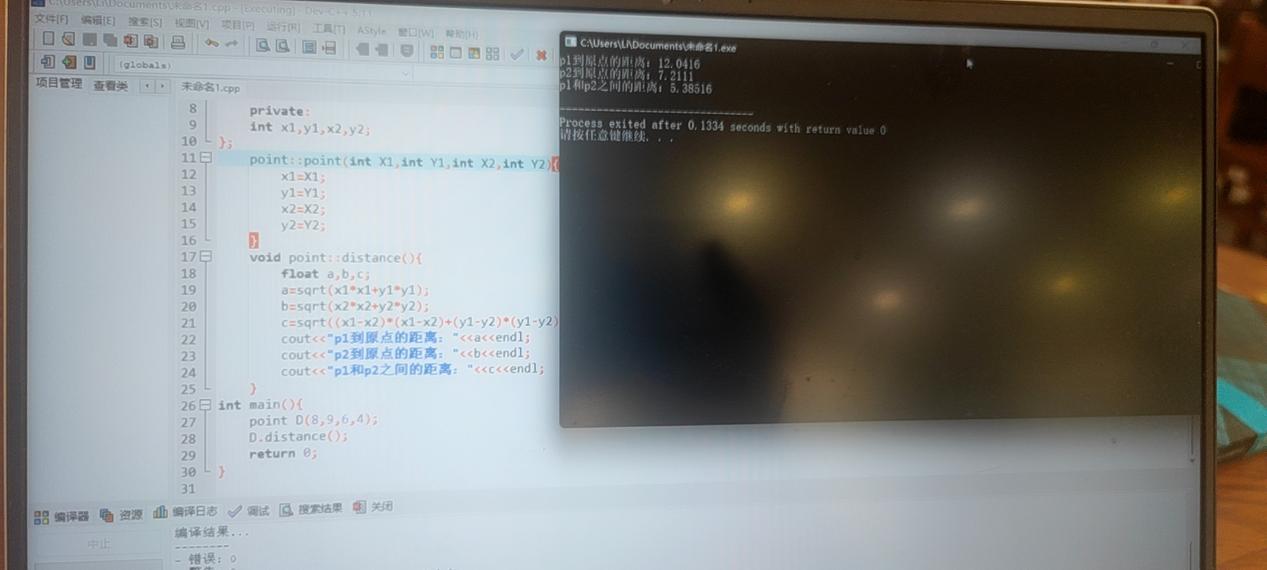
**int main(){**

**point D(8,9,6,4);**

**D.distance();**

**return 0;**

**}**

**运行结果（截图）**：

**2、声明并实现一个股票的类**

**成员函数为：**

获得股票；

增持；

卖出股票；

更新股票价格；

显示关于所持股票的信息；

**数据成员为：**

公司名称；

所持股票的数量；

每股的价格；

股票总值。

**程序如下：**

**要求：**

1. **在划线部分填上合适的语句，使程序完整。**
2. **修改该程序，使用构造函数实现程序功能。**

*#include <iostream>*

*#include <cstring>*

*using namespace std;*

*class Stock*

*{*

*private:*

*char company[50];//公司名称*

*long shares;//所持股票的数量*

*double share\_val;//每股的价格*

*double total\_val;//股票总值*

*void set\_tot()*

*{total\_val=shares\*share\_val;}*

public:

void acquire(char co[50],long n,double pr);//获得股票

void buy(long num, double price);//增持

void sell(long num, double price);//卖出股票

void update(double price);//更新股票价格

void show();//显示所持股票信息

};

void Stock::acquire(char co[50],long n,double pr)

{

→ strcpy(company,co) ;

if(n<0)

{

cout<<"Number of shares can not be negative;"<<company<<" share set to 0.\n";

shares=0;

}

else

shares=n;

share\_val=pr;

set\_tot();

}

void Stock::buy(long num,double price)

{

if(num<0)

cout<<"Number of shares purchased can not be negative. Transaction is aborted.\n";

else

{

shares+=num;

share\_val=price;

set\_tot();

}

}

void Stock::sell(long num,double price)

{

if (num<0)

cout<<"Number of shares sold can not be negative. Transaction is aborted.\n";

else if (num>shares)

cout<<"You can not sell more than you have! Transaction is aborted.\n";

else

{

shares-=num;

share\_val=price;

set\_tot();

}

}

void Stock::update(double price)

{

share\_val=price;

set\_tot();

}

void Stock::show()

{

cout<<"Company: "<<company<<" Shares:"<<shares<<endl;

cout<<"Share Price"<<share\_val<<"$"<<" Total worth:"<<total\_val<<endl;

}

int main()

{

Stock stockl;

stockl.acquire("NanoSmart", 20,12.50);

stockl.buy(15,18.25);

stockl.show();

stockl.sell(400,20.00);

stockl.show();

return 0;

}

**修改后源代码：#include <iostream>**

**#include <cstring>**

**using namespace std;**

**class Stock**

**{**

**private:**

**char company[50];**

**long shares;**

**double share\_val;**

**double total\_val;**

**void set\_tot()**

**{total\_val=shares\*share\_val;}**

**public:**

**Stock(char co[],long n,double pr):shares(n),share\_val(pr){**

**strcpy(company,co);**

**}**

**void acquire();**

**void buy(long num, double price);**

**void sell(long num, double price);**

**void update(double price);**

**void show();**

**};**

**void Stock::acquire(){**

**if(shares<0)**

**{**

**cout<<"Number of shares can not be negative;"<<company<<" share set to 0.\n";**

**shares=0;**

**}**

**else shares=shares;**

**set\_tot();**

**}**

**void Stock::buy(long num,double price)**

**{**

**if(num<0)**

**cout<<"Number of shares purchased can not be negative. Transaction is aborted.\n";**

**else**

**{**

**shares+=num;**

**share\_val=price;**

**set\_tot();**

**}**

**}**

**void Stock::sell(long num,double price)**

**{**

**if (num<0)**

**cout<<"Number of shares sold can not be negative. Transaction is aborted.\n";**

**else if (num>shares)**

**cout<<"You can not sell more than you have! Transaction is aborted.\n";**

**else**

**{**

**shares-=num;**

**share\_val=price;**

**set\_tot();**

**}**

**}**

**void Stock::update(double price)**

**{**

**share\_val=price;**

**set\_tot();**

**}**

**void Stock::show()**

**{**

**cout<<"Company: "<<company<<" Shares:"<<shares<<endl;**

**cout<<"Share Price"<<share\_val<<"$"<<" Total worth:"<<total\_val<<endl;**

**}**

**int main()**

**{**

**Stock stockl("NanoSmart", 20,12.50);**

**stockl.acquire();//这个没必要要了**

**stockl.buy(15,18.25);**

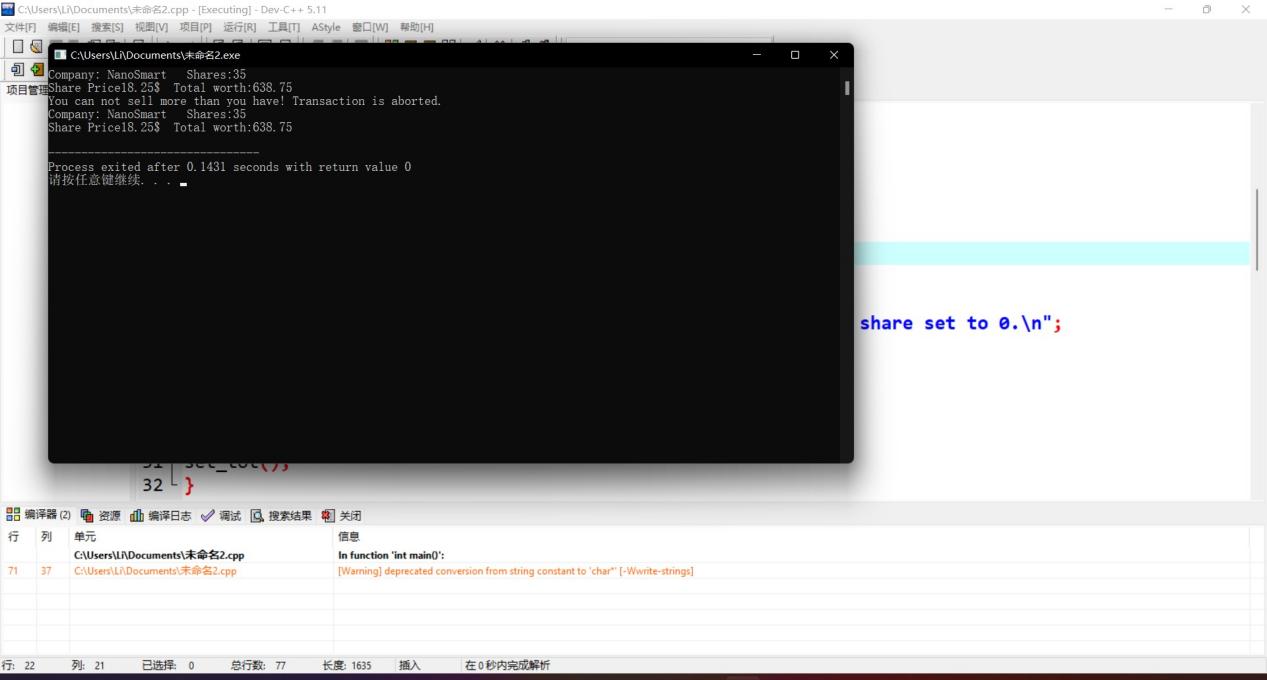
**stockl.show();**

**stockl.sell(400,20.00);**

**stockl.show();**

**return 0;**

**}**

**运行结果（截图）：**

1. **实验体会**

**实验一：做实验一的时候过于粗心大意,定义构造函数是把point写成了class,这样的错误是不应该的，归根结底还是练的太少，平常只求看书学知识而不是去实践。第一题做的不太美丽！**

**实验二：第一次看的时候确实有点眼花，毕竟没怎么见过长程序，但将思路理清了之后发现还是挺容易理解的，但是第一问就做错了，直接用等号定义company和co,运行时发现错误，才想起来得用strcpy函数来将co的值复制到company中去，程序得以运行成功。**

**构造函数是赋初值用的，原程序中acquire函数具有赋初值的功能，因此用构造函数替换掉acquire函数。**