**实验1 类的设计与对象使用**

【实验目的】

1. 熟悉VS开发环境。
2. 掌握多文件结构程序项目的建立过程。
3. 初步掌握面向对象程序的多文件结构项目的开发运行步骤。
4. 掌握类的定义和使用以及类对象的声明和使用方法，理解具有不同访问属性的成员的访问方式。
5. 掌握构造函数和析构函数的作用和编写方法。
6. 了解并掌握静态成员、常成员的使用。

【实验内容】

1．下面的程序在编译时会出错，请修改程序，使之能正确运行。然后得到的正确程序分成3个文件。文件MyClass.h包括类的定义，文件MyClass.cpp包括类成员函数的实现，文件App.cpp包括类的使用。调试、运行该程序，使之获得正确结果。

#include<iostream>

using namespace std;

class MyClass

{

public:

MyClass() { x=0;y=0; }

MyClass(int a = 0, b = 0);

Print();

private:

int x, y;

};

MyClass::MyClass(int a = 0, int b =0)

{

x = a;

y = b;

}

void MyClass::Print()

{

cout << "x = " << x << endl;

cout << "y = " << y << endl;

}

int main()

{

MyClass obj1,obj2(5,8);

obj1.x =1;

obj1.y =3;

obj1.Print();

obj2.Print();

return 0;

}

2．已知两个矩形的长和宽,用面向对象的概念编程求它们的面积和周长。假设矩形 1 的长和宽分别为 20 和 50；矩形 2 的长和宽分别为 3.6 和 4.5。(先定义矩形类再实例化两个对象)

3. 下面程序中定义了一个整型动态数组，编写了3个构造函数。

#include<iostream>

using namespace std;

class IntArray {

public:

IntArray(int sz); // 数组初始化为sz个元素，初值全为0

IntArray( int \*array, int sz ) ; // 用静态数组array的前sz个元素初始化新数组

IntArray( const IntArray &rhs ) ; // 拷贝构造函数

void printAll();

private:

int \*ia;

int size;

};

IntArray::IntArray( int sz ) {

size = sz; // 设置数据成员

ia = new int[ size ];

for ( int ix=0; ix < size; ++ix )

ia[ix] = 0; // 初始化数组元素

}

IntArray:: IntArray( int \*array, int sz ) {

size = sz; // 设置数据成员

ia = new int[ size ];

for ( int ix=0; ix < size; ++ix )

ia[ix ] = array[ ix ]; // 拷贝数据成员

}

IntArray:: IntArray( const IntArray &rhs )

{

size = rhs.size;

ia = new int[ size ];

for (int ix = 0; ix < size; ix++ )

ia[ ix ] = rhs.ia[ ix ];

}

void IntArray::printAll()

{

for ( int ix=0; ix < size; ++ix )

cout<<ia[ix]<<" ";

cout<<endl;

}

int main()

{

int a[10]={1,2,3,4,5,6,7,8,9,10};

IntArray arr1(10), arr2(a,5),arr3(arr2);

arr1.printAll();

arr2.printAll();

arr3.printAll();

return 0;

}

（1）为IntArray加上析构函数并编译运行程序，体会类的构造函数和析构函数的作用。

（2）程序中，类IntArray的三个构造函数都是以相似的方式来实现的。一般来说，当两个或多个函数重复相同的代码时，可将这部分代码抽取出来，形成独立的函数，以便共享。以后如果需要改变这些实现，则只需改变一次，而且这种实现的共享本质更容易为大家所理解。怎样把构造函数中的代码抽取出来形成独立的共享函数呢？请给出一种可能的实现。按你给出的实现方法修改程序并重新编译运行。

4.设有一个点myPoint类的定义如下：

class myPoint {

public:

myPoint(double x0=0.0,double y0=0.0):x(x0),y(y0) {}

myPoint(myPoint &np):x(np.x),y(np.y) {}

double GetX() { return x;}

double GetY() {return y;}

void SetX(double x0) {x=x0;}

void SetY(double y0) {x=y0;}

void SetPoint(double x0,double y0) {x=x0;y=y0;}

void SetPoint(myPoint &np) { x=np.x; y=np.y;}

double GetLength(myPoint p) {

return sqrt((x-p.x)\*(x-p.x) +(y-p.y)\*(y-p.y));

}

void Printit() { cout<<" ("<<x<<","<<y<<") ";}

private:

double x ,y;

};

试定义一个三角形Triangle类，在Triangle类中以点myPoint类的3个对象p1、p2、p3作为数据成员，表示三角形的三个顶点。Triangle类具有计算三角形的周长和面积的功能。请编写程序上机调试并运行。