### 题目1:

1. 先阅读下列程序，写出执行结果。然后输入程序，调试程序，比较结果的正确性。

#include <iostream> // 预处理命令
  
using namespace std; // 使用标准命名空间std
  
   
// 基类
  
class Base
  
{
  
public:
  
// 公有函数:
  
 virtual void f() { cout << "调用Base::f()" << endl; } // 虚函数
  
};
  
   
// 派生类
  
class Derived: public Base
  
{
  
public:
  
// 公有函数:
  
 void f() { cout << "调用Derived::f()" << endl; } // 虚函数
  
};
  
   
int main(void) // 主函数main(void)
  
{
  
 Derived obj; // 定义派生类对象
  
 Base \*p = &obj; // 基类指针
  
 p->f(); // 调用函数f()
  
   
 system("PAUSE"); // 调用库函数system( )，输出系统提示信息
  
 return 0; // 返回值0, 返回操作系统
  
}

### 题目2：

1. 先阅读下列程序，写出执行结果。然后输入程序，调试程序，比较结果的正确性。

#include <iostream> // 预处理命令
  
using namespace std; // 使用标准命名空间std
  
   
// 基类
  
class Base
  
{
  
public:
  
// 公有函数:
  
 virtual void Show() const{ cout << "调用Base::Show()" << endl; } // 虚函数
  
};
  
   
// 派生类
  
class Derived: public Base
  
{
  
public:
  
// 公有函数:
  
 void Show() const{ cout << "调用Derived::Show()" << endl; } // 虚函数
  
};
  
   
void Refers(const Base &obj) // 基类引用
  
{
  
 obj.Show(); // 调用函数Show()
  
}
  
   
int main(void) // 主函数main(void)
  
{
  
 Base obj1; // 定义对象
  
 Derived obj2; // 定义对象
  
   
 Refers(obj1); // 调用函数Refers()
  
 Refers(obj2); // 调用函数Refers()
  
   
 system("PAUSE"); // 调用库函数system( )，输出系统提示信息
  
 return 0; // 返回值0, 返回操作系统
  
}

题目3：

// 文件名: main.cpp
  
#include <iostream> // 预处理命令
  
using namespace std; // 使用标准命名空间std
  
   
// 基类
  
class Base
  
{
  
private:
  
// 数据成员:
  
 int m; // 数据成员
  
   
public:
  
// 公有函数:
  
 Base(int a): m(a){ } // 构造函数
  
 virtual void Show() const { cout << m << endl; } // 虚函数
  
};
  
   
// 派生类
  
class Derived: public Base
  
{
  
private:
  
// 数据成员:
  
 int n; // 数据成员
  
   
public:
  
// 公有函数:
  
 Derived(int a, int b): Base(a), n(a){ } // 构造函数
  
 void Show() const // 虚函数
  
 {
  
 cout << n << ","; // 显示n及","
  
 Base::Show(); // 调用基类的Show()
  
 }
  
};
  
int main(void) // 主函数main(void)
  
{
  
 Base obj1(168); // 定义基类对象
  
 Derived obj2(158, 98); // 定义派生类对象
  
 Base \*p; // 定义基类指针
  
 p = &obj1; // 指向obj1
  
 p->Show(); // 调用Show()
  
 p = &obj2; // 指向obj2
  
 p->Show(); // 调用Show()
  
 p-> [1] ; // 调用基类的Show()
  
 return 0; // 返回值0, 返回操作系统
  
}

### 题目4：

1. 改正下面程序中的错误，使其能正常运行。

#include <iostream> // 预处理命令 //2
  
using namespace std; // 使用标准命名空间std //3
  
 //4
  
// 基类 //5
  
class A //6
  
{ //7
  
public: //8
  
// 公有函数: //9
  
 void ShowA() const{ cout << "基类A" << endl; } //10
  
}; //11
  
 //12
  
// 派生类 //13
  
class B: public A //14
  
{ //15
  
public: //16
  
// 公有函数: //17
  
 void ShowB() const{ cout << "派生类B" << endl; } //18
  
}; //19
  
 //20
  
int main(void) // 主函数main(void) //21
  
{ //22
  
 B obj; // 定义派生类对象 //23
  
 A \*p = &obj; // 定义基类指针指向派生类对象 //24
  
 p->ShowB(); // 调用ShowB() //25
  
 //26
  
 system("PAUSE"); // 调用库函数system( )，输出系统提示信息 //27
  
 return 0; // 返回值0, 返回操作系统 //28
  
}

### 实验作业1

编写程序，定义抽象基类Shape(形状)，由它派生出2个派生类：Circle(圆形) 和Rectangle(矩形)，用函数Show()分别显示各种图形的相关信息，最后还要显示所有图形的总面积。

### 实验作业2

编写程序，定义抽象基类Shape(形状)，由它派生出3个派生类: Circle(圆形)、Rectangle(矩形)和Square 正止方形)，用函数函数ShowArea()分别显示各种图形的面积，最后还要显示所有图形的总面积。要求用基类指针数组，使它的每一个元素指向一一个派生类对象。