**C++面向对象程序设计(甲)（ A ）卷**

**座位号：**

**注意：答案直接写在答题纸上，答在试卷上无效，考试后答题纸和试卷一同上交**

**一、单项选择题（每题2分，总计20分）**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **题 号** | **1** | **2** | **3** | **4** | **5** |
| **答 案** |  |  |  |  |  |
| **题 号** | **6** | **7** | **8** | **9** | **10** |
| **答 案** |  |  |  |  |  |

**二、程序填空题（每空2分，总计20分）**

**（1）**

**（2）**

**（3）**

**（4）**

**（5）**

**（6）**

**（7）**

**（8）**

**（9）**

**（10）**

**三、阅读程序题（每题6分，总计30分）**

**1. 阅读该程序，给出程序的输出结果。**

**2. 阅读该程序，给出程序的输出结果。**

**3. 阅读该程序，给出程序的输出结果。**

**4. 阅读该程序，给出程序的输出结果。**

**5. 阅读该程序，给出程序的输出结果。**

**四、编程题（每题15分，总计30分）**

**座位号：**

**1. 按照要求，编写程序。**

**2. 按照要求，编写程序。**

**座位号：**

**C++面向对象程序设计(甲)（ A ）卷**

**注意：答案直接写在答题纸上，答在试卷上无效，考试后答题纸和试卷一同上交**

**一、单项选择题（每题2分，总计20分）**

|  |  |
| --- | --- |
| **1.** | **下列带缺省值参数的函数说明中，正确的说明是【 】**  **A. int Fun(int x,int y=2,int z=3); B. int Fun(int x=1,int y,int z=3);**  **C. int Fun(int x,int y=2,int z); D. int Fun(int x=1,int y=2,int z);** |
| **2.** | **下面关于类的静态成员的叙述中正确的是【 】**  **A.类的静态成员函数只能访问类的静态数据成员**  **B.类的静态成员函数可直接访问类的所有数据成员**  **C.类的每个对象都有一份自己的静态数据成员**  **D. 静态数据成员只能通过“类名：：”进行访问** |
| **3.** | **当一个函数声明为某个类的友元函数，下列描述正确的是【 】**  **A. 友元函数能访问该类的所有成员 B. 友元函数只能访问该类的公有成员**  **C. 友元函数只能访问该类的成员变量 D. 友元函数只能访问该类的成员函数** |
| **4.** | **下列关于纯虚函数的描述中，错误的是【 】**  **A.只是基类中函数的声明，没有定义**  **B. 可以使用包含纯虚函数的类来创建对象**  **C．当需要使用包含纯虚函数的基类的派生类创建对象时，须在派生类中给出该函数定义。**  **D. 包含纯虚函数的类称为抽象类** |
| **5.** | **下列关于运算符重载描述中，正确的是【 】**  **A. 所有的运算符都能被重载 B. 运算符被重载时，它们的优先级与结合性不会改变**  **C. 可自定义一个新运算符 D. 每个运算符都可以被重载成成员函数和友元函数** |
| **6.** | **可以用p.a的形式访问派生类对象p的基类成员a，其中a是【 】**  **A. 私有继承的公有成员 B. 公有继承的私有成员**  **C. 公有继承的保护成员 D. 公有继承的公有成员** |
| **7.** | **复制初始化构造函数的作用是【 】**  **A. 进行数据类型的转换 B. 用对象调用成员函数**  **C. 用对象初始化对象 D. 用一般类型的数据初始化对象** |
| **8.** | **所谓多态性是指【 】**  **A.不同的对象调用不同名称的函数 B. 不同的对象调用相同名称的函数**  **C. 一个对象调用不同名称的函数 D.一个对象调用不同名称的对象** |
| **9.** | **下列关于创建一个int型变量的引用，正确的是【 】**  **A．int a(3), &ra=a； B．int a(3), &ra=&a；**  **C．double d(3.1)；int &rd=d； D．int a(3)，ra=a；** |
| **10** | **标准模板库（STL）所涉及的4个最主要的基本组件是【 】**  **A．容器、迭代器、算法、函数模板 B．类模板、运算符重载函数、容器、算法**  **C．容器、算法、迭代器、函数对象 D．类、对象、迭代器、函数** |

**二、程序填空题（每空2分，总计20分）**

|  |  |
| --- | --- |
| **1.** | **请在下面程序的横线处填上适当内容，以使程序完整,并使运行结果为：**  **The square’s area is 4**  **#include<iostream.h>**  **class Rectangle**  **{**  **double W,H;**  **public:**  **Rectangle(double w=0,h=0){W=w;H=h;}**  **void show(){cout<<W\*H;}**  **};**  **class Square:public Rectangle {**  **public:**  **Square(double r):\_\_\_\_\_\_\_\_\_\_(1)\_\_\_\_\_\_\_\_\_{};**  **void show()**  **{**  **cout<<”The square’s area is “;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(2)\_\_\_\_\_\_\_\_\_\_\_;**  **}**  **};**  **void main()**  **{**  **Square s(2);**  **s.show();**  **}** |
| **2.** | **下面程序中A是抽象类。请在下面程序的横线处填上适当内容，以使程序完整,并使程序的输出为:**  **B1 called**  **B2 called**  **#include<iostream.h>**  **class A**  **{**  **public:**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_(3)\_\_\_\_\_\_\_\_\_\_\_\_;**  **};**  **class B1:public A**  **{**  **public:**  **void display()**  **{cout<”B1 called”<<endl};**  **};**  **class B2:public A**  **{**  **public:**  **void display()**  **{cout<<”B2 called”<<endl};**  **};**  **void show(\_\_\_\_\_\_\_\_\_(4)\_\_\_\_\_\_\_\_)**  **{**  **p->display();**  **}**  **void main()**  **{**  **B1 b1;**  **B2 b2;**  **A\* p[2]={&b1,&b2};**  **for(int i=0;i<2;i++)**  **show(p[i]);**  **}** |
| **3.** | **请在下面程序的横线处填上适当字句，以使程序完整,并使程序的输出为8。**  **#include<iostream.h>**  **template<class T>**  **void fun(\_\_\_\_\_\_\_\_\_\_\_(5)\_\_\_\_\_\_\_\_\_\_\_)**  **{**  **t=x[0];**  **for(int i=0;i<n;i++)**  **if(a[i]>t)**  **t=a[i];**  **}**  **void main()**  **{**  **int a[]={3,5,8,4,6};**  **int m;**  **fun(a,5,\_\_(6)\_\_\_\_);**  **cout<<m<<endl;**  **}** |
| **4.** | **下面程序通过把类Distance声明为类Point的友元类来实现计算两点之间的距离。请在下面程序的横线处填上适当字句，以使程序完整。**  **#include<iostream>**  **#include<cmath>**  **using namespace std;**  **class Point**  **{**  **double X,Y;**  **public:**  **Point(double x,double y)**  **{X=x;Y=y;}**  **\_\_\_\_\_\_\_\_\_\_\_\_(7)\_\_\_\_\_\_\_\_\_\_\_\_;**  **};**  **class Distance**  **{**  **public:**  **double Dis(Point& p1,Point& p2);**  **{**  **double t;**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(8)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **return t;**  **}**  **}**  **void main(){**  **Point p(10,10),q(20,20);**  **Distance d;**  **cout<<d.Dis(p,q)<<endl;**  **}** |
| **5.** | **下面程序用STL的条件计数算法和自定义的函数对象对一个存放在整数向量类对象中的学生成绩进行统计及格人数并显示结果。请在下面程序的横线处填上适当字句，以使程序完整。**  **#include<iostream>**  **#include<vector>**  **#include<algorithm>**  **#include<functional>**  **using namespace std;**  **class Pass**  **{**  **public:**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(9)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **{**  **return x>=60;**  **}**  **};**  **void main()**  **{**  **vector<int> a;**  **int count;**  **a.push\_back(78);**  **a.push\_back(92);**  **a.push\_back(52);**  **count = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(10)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;**  **cout<<"count="<<count<<endl;**  **}** |

**四、程序阅读题（每题6分，总计30分）**

|  |  |
| --- | --- |
| **1.** | **#include <iostream>**  **using namespace std;**  **int& element(int x[],int index)**  **{**  **return x[index+1];**  **}**  **void main()**  **{**  **int a[]={3,7,2,1,5};**  **element(a,3)=9;**  **for(int i=0;i<5;i++)**  **cout<<a[i]<<’ ’;**  **}** |
| **2.** | **#include<iostream>**  **using namespace std;**  **class A**  **{**  **int num;**  **public:**  **A()**  **{num=0;cout<<"A default constructor"<<endl;}**  **A(int n)**  **{num=n;cout<<”A constructor,num=”<<num<<endl;}**  **~A()**  **{cout<<"A destructor,num="<<num<<endl;}**  **};**  **void main()**  **{**  **A a,\*p;**  **p=new A(2);**  **a=\*p;**  **delete p;**  **cout<<”Exiting main”<<endl;**  **}** |
| **3.** | **#include<iostream>**  **using namespace std;**  **class Format**  **{**  **public:**  **virtual void header()**  **{cout<<"This is a head"<<endl;}**  **virtual void footer()**  **{cout<<"This is a footer"<<endl;}**  **virtual void body()**  **{cout<<"This is a body"<<endl;}**  **void display()**  **{header();body();footer();}**  **};**  **class MyFormat:public Format**  **{**  **public:**  **void header()**  **{cout<<"This is my header"<<endl;}**  **void footer()**  **{cout<<"This is my footer"<<endl;}**  **};**  **void main()**  **{**  **Format \* p;**  **p=new Format;**  **p->display();**  **p=new MyFormat;**  **p->display();**  **}** |
| **4.** | **从键盘输入8 4 3 1 9 6 11 2 5，给出程序的输出结果。**  **#include<iostream>**  **#include<vector>**  **#include<stack>**  **#include<iterator>**  **using namespace std;**  **main()**  **{**  **vector<int> v;**  **stack<int> s;**  **int i,n,x;**  **cin>>n;**  **for(i=0;i<n;i++)**  **{**  **cin>>x;**  **v.push\_back(x);**  **}**  **for(vector<int>::iterator p=v.begin();p!=v.end();p++)**  **if(\*p%2==0)**  **s.push(\*p);**  **while(!s.empty())**  **{**  **cout<<s.top()<<endl;**  **s.pop();**  **}**  **}** |
| **5.** | **#include<iostream>**  **#include<exception>**  **using namespace std;**  **class Fract**  **{**  **int den,num;**  **public:**  **Fract(int d=0,int n=1)**  **{**  **if(n==0)**  **throw exception("divided by zero");**  **den = d;num=n;**  **}**  **int& operator[](int index)**  **{**  **if(index<0||index>1)**  **throw exception("index out of range");**  **if(index==0)**  **return den;**  **if(index==1)**  **return num;**  **}**  **void show()**  **{ cout<<den<<"/"<<num<<endl;}**  **};**  **void main()**  **{**  **try**  **{**  **Fract f(0,2);**  **f.show();**  **cout<<f[1]<<"/"<<f[2]<<endl;**  **}**  **catch(exception e)**  **{**  **cout<<e.what()<<endl;**  **}**  **}** |

**四、编程题（每题15分，总计30分）**

|  |
| --- |
| **1. 按下列要求编程，即实现相关类，并在主函数中使用这些类。**  **圆（Circle）和矩形（Rectangle）都是一种形状（Shape），它们有不同的属性：圆有半径（radius），矩形有长（length）和宽（width），但它们都有计算面积的方法double area()，输入属性值的方法void input()和显示属性值的方法void show()。现有不同形状的圆、矩形若干（假定为3个），将它们的指针放入一个Shape指针数组，用循环来统一输入和显示每个形状的属性值，统计所有形状的总面积并输出。（要求用抽象类和类继承）**  **2、定义一个存储任意类型的动态数组类模板Array，要求如下：**  **(1) 存放数组元素的空间动态申请，定义成员函数时不得造成内存泄露。**  **(2) 定义构造函数、析构函数和复制构造函数。**  **(3) 定义下标运算符[]、赋值运算符=、关系运算符==重载函数。**  **(4) 定义输入运算符>>、输出运算符<<重载函数。**  **（5）编写main函数，对以上所有功能进行测试。** |