《面向对象程序设计 C++》期末考试试卷（A）

班级：_____姓名：_____学号：_____分数：_____

题号	一	二	三	四	总分
得分					

试卷说明：本套试题共四个大题，全部题目都答在答题纸上，写在其他地方均无效。

（答题纸在本套试卷的第 10 页上）

一、选择题（每小题 2 分，共 40 分）

1、C++是（ C ）。

A. 面向对象的程序设计语言

B. 面向过程的程序设计语言

C. 既支持面向对象的程序设计又支持面向过程的程序设计的混合型语言

D. 非结构化的程序设计语言

2、面向对象程序设计思想的主要特征中不包括（ D ）。

A. 封装性　　B. 多态性　　C. 继承性　　D. 功能分解，逐步求精

3、若定义：string str; 当语句cin>>str; 执行时，从键盘输入：

Microsoft Visual Studio 6.0!

所得的结果是str=（ B ）。

A. Microsoft Visual Studio 6.0!　　B. Microsoft

C. Microsoft Visual　　D. Microsoft Visual Studio 6.0

4、考虑下面的函数原型声明：void testDefaulParam(int a,int b=7,char z='*');

下面函数调用中，不合法的是（ C ）。

A. testDefaulParam(5);　　B. testDefaulParam(5,8);

C. testDefaulParam(5,'#');　　D. testDefaulParam(0,0,'*');

5、下列语句中，将函数int sum(int x, int y)正确重载的是（ C ）。

A. float sum(int x, int y);　　B. int sum(int a, int b);

C. float sum(float x, float y);　　D. double sum(int y, int x);

6、下列表示引用的方法中，（ A ）是正确的。

已知：int a=1000;

A. int &x=a;　　B. char &y;　　C. int &z=1000;　　D. float &t=&a;

7、在一个函数中，要求通过函数来实现一种不太复杂的功能，并且要求加快执行速度，选用（ A ）。

A. 内联函数　　B. 重载函数　　C. 递归调用　　D. 嵌套调用

8、下列有关C++类的说法中，不正确的是（ D ）。

A. 类是一种用户自定义的数据类型

B. 只有类中的成员函数或类的友元函数才能存取类中的私有成员

C. 在类中，如果不做特别说明，所有成员的访问权限均为私有的

D. 在类中，如果不做特别说明，所有成员的访问权限均为公用的

9、已知X类，则当程序执行到语句：X array[3];时，调用了（ D ）次构造函数。

A. 0　　B. 1　　C. 2　　D. 3

10、下面说法中，正确的是（ B ）

A. 一个类只能定义一个构造函数，但可以定义多个析构函数

B. 一个类只能定义一个析构函数，但可以定义多个构造函数

C. 构造函数与析构函数同名，只要名字前加了一个求反符号（~）

D. 构造函数可以指定返回类型，而析构函数不能指定任何返回类型，即使是void类型也不可以

11、已知：print( )函数是一个类的常成员函数，它无返回值，下列表示中，（ A ）是正确的。

A. void print( ) const;　　B. const void print( );

C. void const print( );　　D. void print(const);

12、下面描述中，表达错误的是（ B ）

A. 公用继承时基类中的 public 成员在派生类中仍是 public 的

B. 公用继承时基类中的 private 成员在派生类中仍是 private 的

C. 公用继承时基类中的 protected 成员在派生类中仍是 protected 的

D. 私有继承时基类中的public成员在派生类中是private的

13、设置虚基类的目的是（B　　）。

A. 简化程序　　B. 消除二义性　　C. 提高运行效率　　D. 减少目标代码

14、下面（ B　）的叙述不符合赋值兼容规则。

A. 派生类的对象可以赋值给基类的对象

B. 基类的对象可以赋值给派生类的对象

C. 派生类的对象可以初始化基类的对象

D. 派生类的对象的地址可以赋值给指向基类的指针

15、关于虚函数的描述中，（ C　　）是正确的。

A. 虚函数是一个static类型的成员函数

B. 虚函数是一个非成员函数

C. 基类中说明了虚函数后，派生类中与其对应的函数可不必说明为虚函数

D. 派生类的虚函数与基类的虚函数具有不同的参数个数和类型

16、下面关于友元的描述中，错误的是（　D　）。

A. 友元函数可以访问该类的私有数据成员

B. 一个类的友元类中的成员函数都是这个类的友元函数

C. 友元可以提高程序的运行效率

D. 类与类之间的友元关系可以继承

17、下列关于静态数据成员的说法，不正确的是（　C　）。

A. 类中定义的公用静态数据成员，可以通过类的对象来访问

B. 类中定义的所有静态数据成员，都必须在类外初始化

C. 静态数据成员不是所有对象所共用的

D. 普通的成员函数可以直接访问类中的静态数据成员

18、如果表达式++i*k中的“++”和“*”都是重载的友元运算符，若采用运算符函数调用格式，则表达式还可以表示为（ B　　）。

A. operator*(i.operator++(),k)　　B. operator*(operator++(i),k)

C. i.operator++().operator*(k)　　D. k.operator*(operator++(i))

19、下面对模板的声明，正确的是(C)。

A. template<T>　　B. template<class T1, T2>

C. template<class T1, class T2>　　D. template<class T1; class T2>

20、下列的各类函数中，（　C　）不是类的成员函数

A. 构造函数　　B. 析构函数　　C. 友元函数　　D. 复制构造函数

二、填空题（前14个空，每空1分，后3个空，每空2分，共20分）

1、类和对象的关系可表述为：类是对象的_____，而对象则是类的_____。

2、静态成员函数没有隐含的_____，所以，在C++程序中，静态成员函数主要用来访问静态数据成员，而不访问非静态成员。

3、在图1中，A，B，C，D，E，F均是类，其中属于单继承的派生类有_____，属于多继承的派生类有_____，类F的基类有_____，类A的派生类有_____。

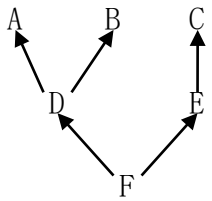

图 1 类的继承层次图

4、如果只想保留公共基类的一个复制，就必须使用关键字_____把这个公共基类声明为虚基类。

5、从实现的角度来讲，多态性可以划分为两类：_____ 和_____。

6、如果一个类包含一个或多个纯虚函数，则该类称为_____。

7、若要把void fun()定义为类A的友元函数，则应在类A的定义中加入语句_____ 。

8、列出C++中的两种代码复用方式：_____ 和_____。

9、析构函数的作用是 _____。

10、假定A是一个类名，则该类的拷贝构造函数的原型说明语句为：_____。

11、后置自增运算符“++”重载为类的成员函数（设类名为A）的形式为：_____。

三、阅读下面4个程序，写出程序运行时输出的结果。（共13分）

1、

```
#include<iostream >
using namespace std;
void fun(int &a, int &b)
{       int p;
        p=a; a=b; b=p;
}
void exchange(int &a, int &b, int &c)
{       if( a<b ) fun(a, b);
        if( a<c ) fun(a, c);
        if( b<c ) fun(b, c);
}
int main()
{       int a=12,b=89,c=56;
        exchange(a, b, c);
        cout<<"a="<<a<<",b="<<b<<",c="<<c<<endl;
        return 0;
}
```

2、

```
#include <iostream>
using namespace std;
class A {
public:
      A( ) { cout << "A"; }
```

```
};
class B {
public:
    B( ) { cout <<"B"; }
};
class C: public A {
public:
    C( ) { cout << "C"; }
private:
    B b;
};                    ABC
int main ( ) {
   C obj;
   return 0;
}
```

3、

```
#include <iostream>
using namespace std;
class A
{public:
    A(){cout<<"A::A() called.\n";}
    virtual ~A(){cout<<"A::~A() called.\n";}
};
class B: public A
{public:
    B(int i)
    {   cout<<"B::B() called.\n";
```

```
        buf=new char[i];
    }
    virtual ~B()
    {   delete []buf;
        cout<<"B::~B() called.\n";
    }
private:
    char *buf;
};
int main()
{   A *a=new B(15);
    delete a;
    return 0;
}
```

4、

```
#include <iostream>
using namespace std;
class A
{public:
    void f(){cout<<"Call A's function f()"<<endl;}
};
class B
{public:
    void f() {cout<<"Call B's function f()"<<endl;}
    void g() {cout<<"Call B's function g()"<<endl;}
};
class C: public A, public B
```

```
{public:
    void g(){cout<<"Call C's function g()"<<endl;}
};
int main()
{   C cc;
    cc.B::f();
   cc.B::g();
    cc.g();
    return 0;
}
```

四、编程题（27 分）

1、（10分） 已知复数类Complex的声明如下：

```
class Complex
{public:
        Complex();                      //无参构造函数
        Complex(double );               //转换构造函数
        Complex(double, double);        //有两个形参的构造函数
        friend Complex operator+(Complex&, Complex&);     //对“+”运算符进行重载
        friend ostream& operator<<(ostream&, Complex&);   //对“<<”运算符进行重载
        friend istream& operator>>(istream&, Complex&);   //对“>>”运算符进行重载
private:
        double real,imag;
};
```

要求：（1）写出该类的所有构造函数的类外定义代码。

（2）写出对运算符“+”、“<<”、“>>”进行重载的运算符重载函数的定义。

2、（17分）下列Shape类是一个表示形状的抽象类，area( )为求图形面积的函数，total( )则是一个通用的用以求不同形状的图形面积总和的函数。

```
class Shape
{public:
    virtual double area()=0;
};
double total(Shape *s[ ], int n)
{   double sum=0.0;
     for(int i=0; i<n; i++) sum+=s[i]->area( );
    return sum;
}
```

要求：（1）从 Shape 类派生圆类(Circle)、正方形类（Square），圆类新增数据成员半径（radius），正方形类新增数据成员边长（a），圆类和正方形类都有构造函数，修改、显示数据成员值的函数，求面积函数。

（2）写出 main()函数，计算半径为 5.5 的圆和边长为 9.9 的正方形的面积和（必须通过调用 total 函数计算）。

《面向对象程序设计 C++》期末考试试卷（A）标准答案

班级：_____姓名：_____学号：_____分数：_____

题号	一	二	三	四	总分
得分					

一、单项选择题（每小题 2 分，共 40 分）

1-5．C D B C C　　6-10．A A D D B

11-15．A B B B C　　16-20．D C B C C

二、填空题（前 14 个空，每空 1 分，后 3 个空，每空 2 分，共 20 分）

1．抽象　实例　2．this 指针

3．E　D、F　A、B、C、D、E　D、F

4．virtual　5．静态多态性　动态多态性

6．抽象类　7．friend void fun(A &a)

8．继承　组合或模板

9．在对象被系统释放之前做一些内存清理工作

10．A(const A&)　11．A operator++(int)

三、阅读程序（13 分）

1、a=89,b=56,c=12

2、ABC

3、A::A() called.

B::B() called.

B::~B() called.

A::~A() called.

4、Call B's function f()

Call B's function g()

Call C's function g()

四、编程题（共27分）

1、（10分）

```
Complex::Complex(){real=0;imag=0;}
Complex::Complex(double r){real=r;}
Complex::Complex(double r,double i){real=r;imag=i;}
Complex operator+(Complex &c1,Complex &c2)
{    Complex c;
     c.real=c1.real+c2.real; c.imag=c1.imag+c2.imag;
     return c;
}
ostream& operator << (ostream& output,Complex& c)
{   output<<"("<<c.real<<"+"<<c.imag<<"i)"<<endl;
    return output;
}
istream& operator >> (istream& input,Complex& c)
{   cout<<"input real and imaginary part of complex number:";
    input>>c.real>>c.imag;
    return input;
}
```

2、（17分）

```
     class Circle:public Shape
```

```
{public:
    Circle(double r){radius=r;}
    void set()
    {   cout<<"Please input the value of the circle:"<<endl;
        cin>>radius;
    }
    void show()
    {   cout<<"the radius of the circle="<<radius<<endl;    }
    double area() {return 3.14159*radius*radius;}
private:
    double radius;
};
class Square:public Shape
{public:
    Square(double a){this->a=a;}
    void set()
    {   cout<<"Please input the value of a:"<<endl;
        cin>>a;
    }
    void show()
    {   cout<<"a="<<a<<endl;    }
    double area(){return a*a;}
private:
    double a;
};
int main()
{
```

```
    Circle C(5.5);
    C.show();
    Square S(9.9);
    S.show();
    Shape *s[]={&C,&S};
    cout<<"total="<<total(s,2)<<endl;
    return 0;
}
```