79面向对象程序设计\_固定试卷模拟1

**一、单选题 （共20题,共40分）**

1. 下列符号中能够作为C++标识符的是 （D）

A.4p B.how many C.friend D.\_256

2. 为了解决程序中函数调用的效率问题，可以将一些函数体代码不是很大，但被频繁调用的函数定义为 （B）

A.重载函数 B.内联函数 C.递归函数 D.友元函数

3. 抽象类至少包含一个 （C）

A.静态函数 B.虚析构函数 C.纯虚函数 D.友元函数

4. C++中重载的运算符>>是 （B）

A.用于输入操作的成员函数 B.用于输入操作的非成员函数

C.用于输出操作的成员函数 D.用于输出操作的非成员函数

5. 下列函数中，具有隐含this指针的是 （B）

class MyClass{

public:

MyClass(); //①

friend void fun2(); //②

static int fun3(); //③

private:

int count; //④

};

A. ① B.② C.③ D.④

6. 已知函数f的原型是void f(int &x, int y);，变量v1、v2的定义是：int v1, v2;，下列调用语句中，正确的是 （C）

A.f(v1,&v2); B.f(v1,v2); C.f(&v1,v2); D.f(&v1,&v2);

7. 下列有关继承和派生的描述中，错误的是 （C）

A.派生类要向基类的构造函数传递参数

B. 多继承时可能产生二义性

C. 派生类可以访问基类的所有数据成员，也能调用基类的所有成员函数

D. 一个基类可以有多个派生类，一个派生类可以有多个基类

8. 下列语句中，会产生编译错误的是 （B）

int i=0, j=1;

int &r=i; //①

&r=j; //②

r=r+5; //③

int \*p=&i; //④

A.① B.② C.③ D.④

9. 下列语句中，会产生编译错误的是 （B）

A.int x=5; int &ref=x; B.const int size;

C.int x=5; int &ref=x; x++; D.const int size=100;

10. 不论派生类以何种方式继承基类，都不能使用基类的 （C）

A.保护成员和私有成员 B.公有成员 C.私有成员 D.保护成员

11. 下列关于类和对象的描述中，错误的是 （B）

A.类是创建对象的模板 B.一个类只能有一个对象

C.一个对象一定属于某个类 D.对象是状态和操作的封装体

12. 如果'-'作为友元函数重载，在使用显式函数调用时，表达式x-y可以表示为 （A）

A.x.operator-(y) B.operator-(x,y) C.y.operator-(x) D.operator-(y,x)

13. 已知函数原型如下：int Fun(int x, int y=2, int z=3)，则下列函数调用错误的是 （D）

A.Fun(4,5,6); B.Fun(1,2); C.Fun(1); D.Fun("1");

14. 下列关于友元的描述中，错误的是 （D）

A.关键字friend用于声明友元 B.一个类的成员函数可以是另一个类的友元

C.友元函数访问对象的成员不受访问特性影响 D.友元函数也是成员函数

15. 在C++中，要实现动态联编，调用虚函数时必须使用 （B）

A.派生类指针 B.基类指针 C.类 D.对象

16. 在下列函数原型中，可以作为类A构造函数的是 （D）

A.void A(int); B.int A(); C.A(int) const; D.A(int);

17. 下列关于常类型的描述中，正确的是 （C）

A.可以通过成员函数修改常对象的状态 B.通过常对象可以调用一般成员函数

C.必须使用成员初始化列表初始化常数据成员 D.常对象中的成员函数都是常成员函数

18. 释放一个类的对象时，系统自动调用 （B）

A.构造函数 B.析构函数 C.成员函数 D.友元函数

19. 当一个派生类继承自一个基类时，基类中的所有公有成员都成为派生类的 （C）

A.可访问成员 B.公有成员 C.私有成员 D.保护成员

20. 下列描述中，作为重载函数调用时选择依据的是 （C）

A.参数名字 B.函数体 C.函数名字 D.返回类型

**二、填空题 （共10题,共20分）**

1. 类成员的访问权限分为公有继承、私有继承和保护继承三类。 （2分）

2. 下列语句序列执行后输出10，请将划线处的语句补充完整。

class MyClass {

public:

MyClass(int x):val(x) { }

void Print();

private :

int val ;

};

void 　　　　Print() { cout<<val<<endl; }

int main() {

MyClass obj(10);

obj.Print();

return 0;

}

3. 有两个类M和C，其中类C定义如下：class C { public: M m;} ;。若建立类C的对象object，则对象m和对象object中后被初始化的是对象\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。 （2分）

4. 下列语句序列执行后输出Hello，请将划线处的语句补充完整。 （2分）

class MyClass {

public:

void Print() const { cout<<"Hello"; }

};

int main() {

MyClass\* p = new MyClass();

　　　　Print();

return 0;

}

5. 请在划线处填写正确内容，使类MyClass的复制构造函数的声明完整。 （2分）

class MyClass {

public:

MyClass(const \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ obj);

};

6. 为了解决多继承产生的\_二义性\_\_问题，在C++中引入了虚基类。 （2分）

7. 派生新类的类称为\_\_父类\_\_，而派生出的新类称\_\_子类\_。（2分）

8. 运算符函数的函数名是由运算符前加关键字\_operator\_构成的。 （2分）

9. 下列语句序列的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_。 （2分）

class MyClass {

public:

MyClass(int x) { cout<<x; }

~MyClass() { cout<<0; }

};

int main() {

MyClass obj1(1),obj2(obj1),&ptr=obj1;

return 0;

}

10. 所有在类内部定义的成员函数都是 inline 函数。 （2分）

**三、改错题 （共1题,共10分）**

下列程序中有三个错误，请改正错误（注意不要修改主函数），使程序的输出结果为：

The point is (0,1)

The point is (3,5)

源文件清单如下：

Line1: #include <iostream.h>

Line2: class Point {

Line3: public:

Line4: Point(int xx=0, int yy) : x(xx), y(yy) { }

Line5: void Move(int xOff, int yOff) const {

Line6: x+=xOff; y+=yOff;

Line7: }

Line8: void Print() const

Line9: { cout<<"The point is ("<<x<<', '<<y<<')'<<endl; }

Line10: private:

Line11: int x,y=0;

Line12: };

Line13: int main() {

Line14: Point p1,p2(2,1);

Line15: p1.Print();

Line16: p2.Move(1,4);

Line17: p2.Print();

Line18: return 0;

Line19: }

**四、阅读程序题 （共4题,共30分）**

1. 请写出下列程序的输出结果。 （8分）

#include <iostream.h>

class Test{

public:

Test(int x=0):val(x) { cout<<"Cons "<<val<<endl; }

Test(const Test &p) { val=p.val; cout<<"Copy "<<val<<endl; }

friend Test operator + (Test &t1,Test &t2);

private:

int val;

};

Test operator + (Test &t1,Test &t2) {

Test temp(t1.val+t2.val);

return temp;

}

int main() {

Test A(1),B(3);

Test C;

C=A+B;

return 0;

}

2. 请写出下列程序的输出结果。 （6分）

#include <iostream.h>

class Base {

public:

void f() { cout<<"fB"<<endl; }

virtual void g() { cout<<"gB"<<endl; }

};

class Derived : public Base {

public:

void f() { cout<<"fD"<<endl; }

virtual void g() { cout<<"gD"<<endl; }

};

int main() {

Base \*p=new Derived;

p->f();

p->g();

return 0;

}

3. 请写出下列程序的输出结果。 （10分）

#include <iostream.h>

class Base{

public:

Base() { cout<<"Base"<<endl; }

~Base() { cout<<"~Base"<<endl; }

};

class Base1 : virtual public Base{

public:

Base1() { cout<<"Base1"<<endl; }

~Base1() { cout<<"~Base1"<<endl; }

};

class Base2 : virtual public Base{

public:

Base2() { cout<<"Base2"<<endl; }

~Base2() { cout<<"~Base2"<<endl; }

};

class Derived : public Base1, public Base2 {

public:

Derived() { cout<<"Derived "<<endl; }

~Derived() { cout<<"~Derived"<<endl; }

private:

Base b;

};

int main() {

Derived d;

return 0;

}

4. 请写出下列程序的输出结果 。 （8分）

#include <iostream.h>

class A {

public:

A(int i):r1(i) { cout<<r1<<endl; }

~A()　　　　 { cout<<'~'<<r1<<endl; }

void print() {cout<<"Empty:"<<r1<<endl;}

void print() const {cout<<"Const:"<<r1<<endl;}

void print(int x) {cout<<"Param:"<<x\*x<<endl;}

private:

int r1;

};

int main() {

A a1(1),a2(2);

a1.print();

a2.print(3);

return 0;

}