

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 (A)

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 _____MyClass::_____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_____m_____。（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();
    _____p->_____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 _____MyClass_____ 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_____1000_____。（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:     }

```

Answer:

```

Line4 : int yy => int yy = 1
Line8 : void Print() const => void Print()
Line11: int x,y =0; => int x,y;

```

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;
}

```

Answer:

Cons 1

Cons 3

Cons 0

Cons 4

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;
}

```

Answer:

fB

gD

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; // 注意此处, 会被忽略~~~};

int main() {
    Derived d;
    return 0;
}

```

Answer:

Base

Base1

Base2

Base

Derived

~Derived

~Base
~Base2
~Base1
~Base

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;
}
```

Answer:

1
2
Empty:1
Param:9
~2
~1

79 面向对象程序设计_固定试卷模拟 2

一、单选题 (共 20 题,共 40 分)

1. 下列运算符中,可以重载的是 D)
- A.?: B.. C.:: D.++
2. 已知 show()函数是一个类的常成员函数,无返回值,下列表示中正确的 (B)
- A.const void show() B.void show() const
- C.void const show() D.void show(const)

3. 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);

4. 下列符号中不能作为 C++标识符的是 (B)

A.points B.5x C.x5 D._256

5. C++流中重载<<的运算符 (C)

- A.用于输出操作的成员函数 B.用于输入操作的成员函数
- C.用于输出操作的非成员函数 D.用于输入操作的非成员函数

6. 下列关于友元的描述中,错误的是 (B)

- A.如果函数 fun()是类 A 的友元,那么在 fun()中可以访问类 A 的私有成员
- B.如果类 A 是类 B 的友元,那么类 B 也是类 A 的友元
- C.如果类 A 是类 B 的友元,那么类 A 的所有成员函数都是类 B 的友元
- D.友元必须在类体内声明

7. 下列函数中,具有隐含的 this 指针的是 (B)

```
class MyClass{
public:
    int fun1(); //①
    friend int fun3(); //②
    static void fun2(); //③
    static int count; //④
};
```

A.① B.② C.③ D.④

8. 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);

9. 抽象类至少包含一个 (B)

A.友元函数 B.纯虚函 C.虚函数 D.虚析构造函数

10. 下列关于静态成员的描述中,正确的 (A)

- A.静态成员不属于对象,是类的共享成员 B.静态数据成员要由构造函数初始化
- C.静态成员函数只能通过类激活 D.非静态成员函数不能操作静态数据成员

11. 下列关于类和对象的描述中,正确的是 (c)

- A.一个类只能有一个对象 B.类对象可以访问类的全部成员
- C.类和对象的关系是一种数据类型与变量的关系 D.对象是对类的抽

12. 下列对常类型的定义中,错误的是 C

A.MyClass const obj; B.char const *ptr; C.const int &ref=10; D.void fun() const;

13. 下列描述中,不是面向对象系统包含的要素的是 D)

A.对象 B.类 C.继承 D.函数

14. 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")`;

15. 下列关于类定义的描述中，错误的是 D

A.类定义中包括数据成员和成员函数的声明 B.类成员的默认访问权限是私有的

C.可以在类体内对数据成员进行初始化 D.成员函数必须在类体内声明

16. 有如下类定义，obj 是类 D 的对象，下列语句中不违反访问控制权限的是

(A)

```
class B{
    public: void fun1();
    private: void fun2();
    protected: void fun3();
};
class D : public B {
    protected: void fun4();
};
```

A.`obj.fun1()`; B.`obj.fun2()`; C.`obj.fun3()`; D.`obj.fun4()`;

17. 下列对析构函数的描述中，正确的是 C

A.系统不能提供默认的析构函数

B.析构函数必须由用户定义

C.析构函数没有参数

D.析构函数可以设置默认参数

18. 已知函数原型为 `int test(int,int,int)`，则下列重载形式中正确的是 B

A.`char test(int,int,int)`; B.`double test(int,int,double)`;

C.`int test(int,int,int=0)`; D.`float test(int,int,int)` ;

19. 有如下声明：`MyClass * const ptr`，下列描述中，正确的是 (b)

A.ptr 是一个类对象 B.ptr 是一个常量指针

C.ptr 指向一个常量 D.ptr 是一个指向常量的常指针

20. C++中，能够被派生类继承的函数是 (d)

A.析构函数 B.友元函数 C.构造函数 D.成员函数

二、填空题 (共 10 题,共 20 分)

1. 类的三种继承方式是__公有继承__、__私有继承__和__保护继承__。 (2 分)

2. 下列语句序列的输出结果是__ 120000__。 (2 分)

```
class MyClass {
public:
    MyClass(int x=0) { cout<<x; }
    ~MyClass() { cout<<0; }
};
```

```
int main() {
    MyClass arr[3]={MyClass(1),MyClass(2)};
    return 0;
}
```

3. 下列语句序列执行后输出 10，请将划线处的语句补充完整。 (2 分)

```
class MyClass {
public:
    MyClass(int x) : __val(x)___{ }
    void Print() { cout<<val<<endl; }
private:
    int val;
};
int main() {
    MyClass obj(10);
    obj.Print();
    return 0;
}
```

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

```
class Test{
public:
    Test(const __Test__obj);
};
```

5. 在 C++中，__函数重载__是指同一个函数名可以对应多个函数的实现。

6. 请在划线处写出 MyClass 类的析构函数声明。

(2 分)

```
class MyClass {
public:
    __ ~MyClass();__
};
```

7. 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();
    p->Print();
    return 0;
}
```

8. 运算符函数的函数名是由运算符前加关键字 operator 构成的。

9. 请在划线处填写构造函数的声明，使得语句 `Point p1;` 创建的 `p1` 对象为原点。（2 分）

```
class Point {
public:
    __Point(int x = 0, int y = 0): x(x), y(y) {}
private:
    int x,y;
};
```

10. 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 m。

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：

```
Kelly is 10 years old.
Patrick is 35 years old.
```

源文件清单如下：

```
Line1:      #include <iostream.h>
Line2:      class Person {
Line3:      public:
Line4:          Person(char *str, int _age=10) : age(_age)
Line5:              { name=new char[strlen(str)+1]; strcpy(name,str); }
Line6:          void ~Person() { delete[ ] name; }
Line7:          void SetAge(int num) const { age=num; }
Line8:          void Print() const
Line9:              { cout<<name<<" is "<<age<<" years old. "<<endl; }
Line10:     private:
Line11:         char *name;
Line12:         int age=0;
Line13:     };
Line14:     int main() {
Line15:         Person p1("Kelly"),p2("Patrick",25);
Line16:         p1.Print();
```

```
Line17:         p2.SetAge(35);
Line18:         p2.Print();
Line19:         return 0;
Line20:     }
```

Answer:

```
Line12 : int age=0 => int age;
```

Line 7: 去掉 const

Line6: 去掉 void

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 Point{
public:
    Point() { number++; }
    ~Point() { number--; cout<<number<<endl; }
    static int GetNumber() { return number; }
private:
    static int number;
};
int Point::number=0;
int main() {
    Point A,B;
    Point *ptr=new Point[3];
    cout<<Point::GetNumber()<<endl;
    delete[] ptr;
    return 0;
}
```

answer:

```
5
4
3
2
1
0
```

2. 请写出下列程序的输出结果。（10 分）

```
#include <iostream.h>
```

```

class Base1 {
public:
    Base1() { cout<<"Base1"<<endl; }
    ~Base1() { cout<<"~Base1"<<endl; }
};

class Base2 {
public:
    Base2() { cout<<"Base2"<<endl; }
    ~Base2() { cout<<"~Base2"<<endl; }
};

class Derived : public Base2, public Base1 {
public:
    Derived() { cout<<"Derived "<<endl; }
    ~Derived() { cout<<"~Derived"<<endl; }
private:
    Base2 b2;
    Base1 b1;
};

int main() {
    Derived d;
    return 0;
}

```

answer:

```

Base2
Base1
Base2
Base1
Derived
~Derived
~Base1
~Base2
~Base1
~Base2

```

3. 请写出下列程序的输出结果。（6分）

```

#include <iostream.h>

class Base {

```

```

public:
    virtual void f() { cout<<"fB"<<endl; }
    void g() { cout<<"gB"<<endl; }
};

class Derived : public Base {
public:
    virtual void f() { cout<<"fD"<<endl; }
    void g() { cout<<"gD"<<endl; }
};

int main() {
    Derived d;
    Base *p=&d;
    p->f();
    p->g();
    return 0;
}

```

answer:

```

fD
gB

```

4. 请写出下列程序的输出结果。（6分）

```

#include <iostream.h>

class MyClass {
public:
    MyClass(int x):val(x) { cout<<val<<endl; }
    ~MyClass() { cout<<"~"<<val<<endl; }
    int Add() { val=2*val; return val; }
    int Add(int x) { val=val+x; return val; }
    void Print() const { cout<<val<<endl; }
private:
    int val;
};

int main() {
    MyClass obj1(10),obj2(30);
    obj1.Add();
    obj2.Add(10);
    obj1.Print();
}

```

```
obj2.Print();
return 0;
}
```

answer:

10
30
20
40
~40
~20

79 面向对象程序设计_固定试卷模拟 3

一、单选题（共 20 题,共 40 分）

1. 当派生类从基类私有继承时，基类中的公有成员和保护成员成为派生类的（B）

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

2. 下列符号中能够作为 C++标识符的是（A）

A.points B.2b C.const D.-256

3. 下列关于友元的描述中，错误的是（B）

A.如果函数 fun()是类 A 的友元，那么在 fun()中可以访问类 A 的私有成员
B.如果类 A 是类 B 的友元，那么类 B 也是类 A 的友元
C.如果类 A 是类 B 的友元，那么类 A 的所有成员函数都是类 B 的友元
D.友元必须在类体内声明

4. C++中，实现封装性需借助于（B）

A.枚举 B.类 C.数组 D.函数

5. 若有函数调用 fun(x+y, 3, min(a,b))，则 fun 的实参个数为（A）

A.3 B.4 C.5 D.6

6. 下列关于类和对象的描述中，错误的是（B）

A.类具有封装性 B.类对象可以访问类的全部成员
C.类是所有对象的行为和不同的状态的集合体 D.对象是类的具体实例

7. 下列选项中，不是类成员函数的是（D）

A.构造函数 B.析构函数 C.虚函数 D.友元函数

8. 下列程序段中，横线处应填入的内容是（D）

```
class MyClass{
public:
    MyClass() { count++; }
private:
    static int count;
```

```
};
```

```
_____count=10;
```

A.int B.static int C.static MyClass::int D.int MyClass::

9. 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

10. 下列函数中，具有隐含的 this 指针的是（B）

```
void fun1(); //①
class MyClass{
public:
    friend void fun2(); //②
    static int fun3(); //③
    int fun1(); //④
};
```

A.① B.② C.③ D.④

11. 有如下类定义，obj 是类 D 的对象，下列语句中不违反访问控制权限的是（A）

```
class B{public: void fun1(); private: void fun2(); protected: void fun3()};
class D : public B {protected: void fun4()};
```

A.obj.fun1(); B.obj.fun2(); C.obj.fun3(); D.obj.fun4();

12. 下列运算符中，不能被重载的是（C）

C++语言中的不能重载的操作符是. , . * , -> *, ::和:。

A.* B.!= C:: D.++

13. 下列关于常类型的描述中，错误的是（B）

A.不能修改常对象的状态 B.通过常对象可以调用一般成员函数
C.常成员函数中不能修改对象的状态 D.必须使用成员初始化列表初始化常数据成员

14. 下列选项中，istream 类对象是（A）

A.cin B.cout C.cerr D.clog

15. 可以解决程序中代码量不大，但却被频繁调用的函数的调用效率问题的是（D）

A.友元函数 B.重载函数 C.虚函数 D.内联函数

16. 下列格式控制符中，可以设置输出项域宽的是（A）

A.setw B.setfill C.setprecision D.endl

17. 下列对常类型的定义中，错误的是（C）

A.MyClass const obj(1,2); B.char const *ptr="hello";
C.const int &ref; D.void fun() const;

18. 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（A）

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

19. 下列关于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. 多继承时可能会产生二义性

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

20. 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);

二、填空题 (共 10 题,共 20 分)

1. 在 C++ 中，____函数重载____是指同一个函数名可以对应多个函数的实现。 (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 (2 分)

```
class MyClass {
public:
    MyClass(int x):val(x) { }
    void Print() { cout<<val<<endl; }
private:
    int val;
};

int main() {
    MyClass obj(10);
    _____obj._____Print();
    return 0;
}
```

3. 已知一个函数的原型是 double fun(double x);，若要以 4.25 为实参调用该函数，应使用表达式____fun(4.25)_____。

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。

```
class MyClass {
public:
    void Print() const { cout<<"Hello"; }
};

int main() {
    MyClass* p = new MyClass();
    _____p->_____Print();
    return 0;
}
```

5. 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 中先被初始化的是对象_____m_____。

6. 请在划线处填写构造函数的声明，使得语句 Point p1; 创建的 p1 对象为原点。 (2 分)

```
class Point {
public:
    _____Point(int x=0,int y = 0):x(x),y(y){}_____
private:
    int x,y;
};
```

7. 请在划线处写出 MyClass 类的析构造函数声明。

(2 分)

```
class MyClass {
public:
    ~MyClass(){}_____
};
```

8. 设置虚基类的目的是为了解决多继承产生的____二义性____问题。 (2 分)

9. 将一个函数声明为一个类的友元函数必须使用关键字____friend_____ (2 分)

10. 派生新类的类称为____父类_____，而派生出的新类称____子类_____

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：

```
Constructor
The value is 10
Destructor
```

源文件清单如下：

```
Line1:      #include <iostream.h>
Line2:      class MyClass
Line3:      {
Line4:      public:
Line5:          MyClass(int x):value(x) { cout<<"Constructor"<<endl; }
Line6:          void ~MyClass() { cout<<"Destructor"<<endl; }
Line7:          void Print() const;
Line8:      private:
Line9:          int value=0;
Line10:     };
Line11:     void MyClass::Print()
Line12:     {
Line13:         cout<<"The value is "<<value<<endl;
```

```
Line14:      }  
Line15:      int main()  
Line16:      {  
Line17:          const MyClass object(10);  
Line18:          object.Print();  
Line19:          return 0;
```

1. 请写出下列程序的输出结果。

```
Line20:      }
```

Answer:

Line6: 去掉 void

Line11: 需要 const

四、阅读程序题（共 4 题,共 30 分）

```

#include <iostream.h>

class Base {
public:
    virtual void f() { cout<<"fB"<<endl; }
    virtual void g() { cout<<"gB"<<endl; }
};

class Derived : public Base {
public:
    virtual void f() { cout<<"fD"<<endl; }
    virtual void g() { cout<<"gD"<<endl; }
};

int main() {
    Base b,*p=new Derived;
    b.f();
    p->g();
    return 0;
}

```

answer:

fB
gD

2. 请写出下列程序的输出结果。

```

#include <iostream.h>

class Base1 {
public:
    Base1() { cout<<"Base1"<<endl; }
    ~Base1() { cout<<"~Base1"<<endl; }
};

class Base2 {
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:
    Base1 b1;
    Base2 b2;
};

int main() {
    Derived d;
    return 0;
}

```

answer:

Base1
Base2
Base1
Base2
Derived
~Derived
~Base2
~Base1
~Base2
~Base1

3. 请写出下列程序的输出结果。

```

#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);
    const A a2(2);
    a1.print();
    a2.print();
    return 0;
}
```

answer:

1

2

Empty:1

Const:2

~2

~1

4. 请写出下列程序的输出结果。

```
#include <iostream.h>
```

```
class Point{
public:
    Point() { number++; cout<<number<<endl; }
    ~Point() { number--; }
    static int GetNumber() { return number; }
private:
    static int number;
};

int Point::number=0;

int main() {
    Point A,B;
    Point *ptr=new Point[3];
    delete[] ptr;
    cout<<Point::GetNumber()<<endl;
    return 0;
}
```

answer:

1

2

3

4

5

2