

C++程序设计试题及答案解析（四）

C++程序设计模拟试卷(四)

一、单项选择题

1. 当一个类的某个函数被说明为virtual时，该函数在该类的所有派生类中（）

- A. 都是虚函数
- B. 只有被重新说明时才是虚函数
- C. 只有被重新说明为virtual时才是虚函数
- D. 都不是虚函数

答案：A

解析：(P170)在基类声明为virtual的函数为虚函数，在派生类中只要有相同的函数（函数名相同、返回值相同、形参类型和个数相同）即使不用virtual说明，也都是虚函数。

2. 要禁止修改指针p本身，又要禁止修改p所指向的数据，这样的指针应定义为（）

- A. const char *p="ABCD";
- B. char *const p="ABCD";
- C. char const *p="ABCD";
- D. const char * const p="ABCD";

答案：D

解析：(P12)const char *p说明禁止通过p修改所指向的数据。char * const p则说明不能修改指针p的地址。因此const char * const p="ABCD"; 它禁止修改指针p本身，又禁止修改p所指向的数据。

3. 函数调用func((exp1,exp2),(exp3,exp4,exp5))中所含实参的个数为（）

- A. 1
- B. 2
- C. 4
- D. 5

答案：B

解析：(exp1,exp2)、(exp3,exp4,exp5)表示是两个逗号表达式，值是最后一个值，相当于两个参数。因此实参的个数是2。

4. 设有函数模板

```
template <class Q>
```

```
Q Sum(Q x,Q y)
```

```
{return (x)+(y);}
```

则下列语句中对该函数模板错误的使用是（）

- A. Sum(10,2);
- B. Sum(5.0,6.7);
- C. Sum(15.2f,16.0f);
- D. Sum("AB","CD");

答案：D

解析：(P40)由Q Sum(Q x,Q y)可知形参和函数返回值都是同一种数据类型。A、B、C三项都正确。而D项用字符串作为实参，字符串的操作与数值类型不同，要用特殊方法进行字符串的连接和运算。

5. 类B是类A的公有派生类，类A和类B中都定义了虚函数func(),p是一个指向类A对象的指针

，则p->A::func()将（）

- A. 调用类A中的函数func()
- B. 调用类B中的函数func()
- C. 根据p所指的對象类型而确定调用类A中或类B中的函数func()
- D. 既调用类A中函数，也调用类B中的函数

答案: A

解析: (P117)指向类成员指针的使用，A::func()是明确调用A类的func函数，所以不管p指向基类或者派生类对象，都执行基类虚函数。注意p->A::func()和p->fun();进行区分。如果使用p->fun()，因为p指向派生类对象，由动态多态性可知要调用派生类的虚函数。

6. 在面向对象的程序设计中，首先在问题域中识别出若干个 ()

- A. 函数
- B. 类
- C. 文件
- D. 过程

答案: B

解析: (P31)面向过程的和面向对象都具有、函数、文件和过程这些概念，而面向对象程序才有类和对象的特征。所以选择B。

7. 已知f1和f2是同一类的两个成员函数，但f1不能直接调用f2，这说明 ()

- A. f1和f2都是静态函数
- B. f1不是静态函数，f2是静态函数
- C. f1是静态函数，f2不是静态函数
- D. f1和f2都不是静态函数

答案: C

解析: (P107)普通成员函数可以调用静态函数，相反静态函数不能调用普通成员函数，这与普通函数与常成员函数相同。因此选择C项。

8. 下列有关模板和继承的叙述正确的是 ()

- A. 模板和继承都可以派生出一个类系
- B. 从类系的成员看，模板类系的成员比继承类系的成员较为稳定
- C. 从动态性能看，继承类系比模板类系具有更多的动态特性
- D. 相同类模板的不同实例一般没有联系，而派生类各种类之间有兄弟父子等关系

答案: D

解析: (P145)类是相同类型事物的抽象，具有不同的操作。而模板是不同类型的事物，具体相同的操作的抽象。类模板的实例化后，各个对象没有任何关系。而类对象是通过派生、继承等关系的关系。

9. 有关C++编译指令，以下叙述正确的是 ()

- A. C++每行可以写多条编译指令
- B. #include指令中的文件名可含有路径信息
- C. C++的编译指令可以以#或//开始
- D. C++中不管 # if后的常量表达式是否为真，该部分都需要编译

答案: B

解析: (P96)编译指令以 # 作为开头，只能一行写一条，# if有选择进行编译，所以选择B项。

10. 在C++中不返回任何类型的函数应该说明为 ()

- A. int
- B. char
- C. void

D. double

答案: C

解析: 无形参或无返回值都可以用void来声明, int char double分别是整型、字符型和实型。

11. 若Sample类中的一个成员函数说明如下:

void set(Sample& a), 则Sample& a的含义是 ()

A. 指向类Sample的名为a的指针

B. a是类Sample的对象引用, 用来作函数Set () 的形参

C. 将a的地址赋给变量Set

D. 变量Sample与a按位与的结果作为函数Set的参数

答案: B

解析: (P53)成员函数使用对象的引用作为形参。该函数的功能是将已知对象的所有数据成员的值拷贝给相应对象的所有数据成员, 不会建立临时对象, 这里是对象的引用所以选择B。

12. 下列关于静态数据成员的描述中正确的是 ()

A. 静态数据成员是类的所有对象所共有的

B. 静态数据成员要在构造函数内初始化

C. 类的每个对象有自己的静态数据成员

D. 静态数据成员不能通过类的对象调用

答案: D

解析: (P107)静态成员属于类的即所有对象所共享的, 只能在外部进行初始化。使用时可以使用形式有两种, 类名::静态成员或者对象.静态成员。所以选择D项。

13. 在编译指令中, 宏定义使用哪个指令 ()

A. #if

B. #include

C. #define

D. #error

答案: C

解析: (P7)#if条件编译, # include文件包含, # error错误处理。

14. 类的析构函数是对一个对象进行以下哪种操作时自动调用的是 ()

A. 建立

B. 撤销

C. 赋值

D. 引用

答案: B

解析: (P80)删除对象或结束程序时, 自动调用析构函数。

15. 关于new运算符的下列描述中, 错误的是 ()

A. 它可以用来动态创建对象和对象数组

B. 使用它创建的对象或对象数组可以使用运算符delete删除

C. 使用它创建对象时要调用构造函数

D. 使用它创建对象数组时必须指定初始值

答案: D

解析: (P107)静态成员的特性是静态成员只有一个拷贝 (副本), 这个副本被所有属于这个类的对象共享。这种共享与全局变量或全局函数相比, 既没有破坏数据隐藏的原则, 又保证了安全性。静态成员表示整个类范围的信息, 其声明以关键字static开始, 包括静态数据成员和静态成员函数。在对静态数据成员初始化时应注意:

- (1)应在类体外对静态数据成员进行初始化（静态数据成员的初始化与它的访问控制权限无关）。
- (2)静态数据成员初始化时前面不加static关键字，以免与一般静态变量或对象混淆。
- (3)由于静态数据成员是类的成员，因此在初始化时必须使用作用域运算符(::)限定它所属的类。因此选择D项。

16. 如果类A被说明成类B的友元，则（）

- A. 类B不一定是类A的友元
- B. 类B的成员即类A的成员
- C. 类A的成员即类B的成员
- D. 类A的成员函数不得访问类B的成员

答案: A

解析: (P113)友元关系不能被继承，友元关系是单向的，友元关系不具有传递性。但是友元函数不是类的成员，所以选择A项。

17. 假定一个类的构造函数为A(int aa,int bb){a=aa++;b=a*++bb;}，则执行A x(4,5);语句后，x.a和x.b的值分别为（）

- A. 4和5
- B. 4和20
- C. 4和24
- D. 20和5

答案: C

解析: (P75)执行构造函数将数据成员进行赋值，aa++是后加，先赋值a = 4，++bb,bb变量值先自加为6，再与a相乘，所以b = 24。

18. 下列运算符中，在C++语言中不能重载的是（）

- A. *
- B. >=
- C. ::
- D. /

答案: C

解析: (P186)除了类属关系运算符"."、成员指针运算符".'"、作用域运算符"::"、sizeof运算符和三目运算符"?:"以外，C++中的所有运算符都可以重载。

19. C++语言建立类族是通过（）

- A. 类的嵌套
- B. 类的继承
- C. 虚函数
- D. 抽象类

答案: B

解析: (P174)类族即同一个类派生出来的类，各个类是兄弟或父子关系。

20. 在C++语言中，数据封装要解决的问题是（）

- A. 数据的规范化
- B. 便于数据转换
- C. 避免数据丢失
- D. 防止不同模块之间数据的非法访问

答案: D

解析: (P39)封装是指把对象属性和操作结合在一起，构成独立的单元，它的内部信息对外界是

隐蔽的，不允许外界直接存取对象的属性，只能通过有限的接口与对象发生联系。类是数据封装的工具，对象是封装的实现。类的访问控制机制体现在类的成员中可以有公有成员、私有成员和保护成员。对于外界而言，只需要知道对象所表现的外部行为，而不必了解内部实现细节。封装体现了面向对象方法的“信息隐蔽和局部化原则”。

二、填空题(本大题共20小题，每小题1分，共20分)请在每小題的空格中填上正确答案

。错填、不填均无分。

1. 若要使用string类，需要引入的头文件是___。

答案：(P40)string.h

【解析】编译时要将头文件引入才能使用标准库中的方法或成员。

2. 在函数前面用___保留字修饰时，则表示该函数表为内联函数。

答案：(P59)inline

【解析】内联函数，用来提高程序运行速度。在类内部定义的函数也是内联函数。

3. 向量操作方法中___方法返回向量中的第一个对象。

答案：(P151)front

【解析】考察向量方法的使用。front():返回向量中的第1个对象。back ()：返回向量中的最后一个对象。operator [] (size_type,n):返回向量中的第n + 1个对象（下标为n的向量元素）。

4. C++派生类使用两种基本的面向对象技术：第一种称为性质约束，即对基类的性质加以限制；第二种称为___，即增加派生类的性质。

答案：(P129)性质扩展

【解析】派生类通过继承可以从基类中获得成员，也可以自定义成员。

5. 重载的运算符保持其原有的___、优先级和结合性不变。

答案：(P183)操作数

【解析】运算符重载时要遵循以下规则：

(1)除了类属关系运算符“.”、成员指针运算符“.”、作用域运算符“::”、sizeof运算符

和三目运算符“?:”以外，C++中的所有运算符都可以重载。

(2)重载运算符限制在C++语言中已有的运算符范围内的允许重载的运算符之中，不能创建新的运算符。

(3)重载之后的运算符不能改变运算符的优先级和结合性，也不能改变运算符操作数的个数及语法结构。

6. 编译时的多态性通过___函数实现。

答案：(P165)重载

【解析】编译多态性，实现的方法主要通过函数的重载或运算符的重载。

7. 预处理语句有三种，分别是宏定义、文件包含和___。

答案：(P7)条件编译

【解析】宏定义 # define,文件包含 # include和条件编译#if等。

8. 构造函数、析构函数和友元函数中，不是该类成员的是___。

答案：(P109)友元函数

【解析】友元函数不是类成员，但可以访问类中成员。

9. 控制格式输入输出的操作中，函数___是设置域宽的。要求给出函数名和参数类型）。

答案：(P193)setw(int)

【解析】setw(int n):用来设置n输出宽度。

10. 派生类的成员一般分为两部分，一部分是___，另一部分是自己定义的新成员。

答案：(P127)从基类继承的成员

【解析】派生类成员一个来自继承基类成员，一个来自本身增加的成员。

11. C + + 中ostream的直接基类___。

答案: (P193)ios

【解析】 istream和ostream的直接基类是ios。

12. vector的___方法返回向量中的最后一个对象。

答案: (P151)back

【解析】 front():返回向量中的第1个对象。back () : 返回向量中的最后一个对象。

operator [] (size_type,n):返回向量中的第n + 1个对象（下标为n的向量元素）。

13. 执行下列代码

```
int i=230;
cout <<"i"<<hex <<i<<endl;
```

程序的输出结果为___。

答案: (P193)i=e6

【解析】 流类库中使用格式符，输出十六进制数据。

14. 在C++中有两种参数传递方式即值传递和___传递。

答案: (P51)引用

【解析】 函数参数传递有传值和传引用两种。

15. 使用new为int数组动态分配10个存储空间是___。

答案: (P10)new int [10] ;

【解析】 new delete动态开辟空间和删除空间。new int [10] , 注意不要写成new

int (10) , 使用小括号只能开辟一个空间, 使用10来初始化该值。

16. 面向对象的四个基本特性是多态性、继承性、和封装性___。

答案: (P37)抽象

【解析】 考察面向对象的四个特征。程序由一组抽象的对象组成，一组对象的共同特征抽象出类的概念，类是对象的抽象，对象是类的实例。封装即将数据和操作紧密结合提供访问的接口，外部通过接口实现访问数据，提供安全性。继承继承解决了类的扩展性。多态性不同对象调用相同的函数名，但调用不同的函数，实现不同的功能，解决了接口统一的问题。

17. 定义虚函数所用的关键字是___。

答案: (P170)virtual

【解析】 在成员函数前加virtual修饰的函数就是虚函数。但不是所有成员函数都可以定义为虚函数的。比如构造函数，不能定义虚函数。

18. 执行下列代码

```
cout<<"oct:"<<oct<<34;
```

程序的输出结果是___。

答案: (P193)Oct:42

【解析】 oct表示八进制，hex表示十六进制，但它们只能输出整型的数据。

19. 在C++中要创建一个文件输入流对象fin，同时该对象打开文件“Test.txt”用于输入，则正确的声明语句是___。

答案: (P200)ifstream fin(“Test.txt”);

【解析】 文件操作中ifstream用于文件的输入，可以调用它的构造函数与要打开的文件进行关联

20. 如果一个派生类只有一个唯一的基类，则这样的继承关系称为___。

答案: (P130)单一

【解析】 根据派生类所拥有的基类数目不同，可以分为单继承和多继承。一个类只有一个直接基类时，称为单继承；而一个类同时有多个直接基类时，则称为多继承。

基类与派生类之间的关系如下：

(1)基类是对派生类的抽象，派生类是对基类的具体化，是基类定义的延续。

(2)派生类是基类的组合。多继承可以看作是多个单继承的简单组合。

(3)公有派生类的对象可以作为基类的对象处理。

三、改错题(本大题共5小题，每小题2分，共10分)下面的类定义中有一处错误，请用下横线标出错误所在行并给出修改意见。

1. class ABC

```
{int a;  
  
public:  
  
ABC(int aa)a(aa){}  
  
};
```

答案：ABC(int aa)a(aa){}初始化列表格式错误。

【修改】 ABC(int aa): a(aa){}

2. #include <iostream.h>

```
class T  
  
{protected:  
  
int p;  
  
public:  
  
T(int m){p=m;}  
  
};  
  
void main()  
{ T a(10);  
  
cout<<a.p<<endl;  
  
}
```

答案： 【修改】 public

【解析】 protected保护类型的成员，不能在类外访问。

3. #include <iostream>

```
using namespace std;  
  
class Date;  
  
class Time  
{public:  
  
Time(int h,int m,int s)  
{hour=h,minute=m,sec=s;}  
  
void show(Date & d);  
  
private:  
  
int hour,minute,sec;  
  
};  
  
class Date  
{public:  
  
Date(int m,int d,int y)  
{month=m,day=d,year=y;}  
  
void Time::show(Date &);  
  
private:  
  
int month,day,year;
```

```
};

void Time::show(Date & d)

{cout<<d.month <<" "<<d.day<<" "<<d.year<<endl;

cout<<hour<<" ":"<minute<<" ":"<sec<<endl;

}

void main()

{Time t1(9,23,50);

Date d1(12,20,2008);

t1.show(d1);
```

答案： void Time::show(Date &);成员函数作为友元函数，要加friend。

[修改] friend void Time::show(Date &);

4. 输出最小值，有一处错误。

```
#include <iostream.h>

class Test

{int a,b;

int getmin()

{return (a<b?a:b);}

public:

int c;

void setValue(int x1,int x2,int x3)

{a=x1;b=x2;c=x3;}

int GetMin();

};

int Test::GetMin()

{int d=getmin();

return (d<c?d:c);

}

void main()

{Test t1;

t1.setValue(34,6,2);

cout<<t1.getmin ()<<endl;

}
```

答案： cout<<t1.getmin()<<endl;采用默认访问权限即私有的，在外部无法访问。

[修改] cout<<t1.GetMin()<<endl;

5. 实现数值、字符串的交换。

```
#include <iostream>

#include <string>

using namespace std;

template<class T>

void Swap(T& a,T& b)

{T temp;

temp=a,a=b,b=temp;

}

void main()
```



```

{int a=5,b=9;

char s1 [] ="Hello",s2 [] ="hi";

Swap(a,b);

Swap(s1,s2);

cout<<"a="<<a<<"",b="<<b<<endl;

cout<<"s1="<<s1<<"",s2="<<s2<<endl;

}

```

答案: char s1 [] ="Hello",s2 [] ="hi";使用Swap(s1,s2)调用交换的是地址。字符指针作实参，形参值发生改变，实参也就发生变化。

[修改] char *s1="Hello",*s2="hi";

四、完成程序题(本大题共5小题，每小题4分，共20分)

1. 在下划线处填上缺少的部分。

```

#include <iostream.h>

class A

{int a,b;

public:

_____; //定义构造函数，使参数i和j的默认值为0

{a=i;b=j;}//在函数体中用i初始化a，用j初始化b

};

main()

{A *p;

_____;//调用带参构造函数生成由p指向的动态对象

//使a和b成员分别被初始化为4和5

}

```

答案: A(int i=0,int j=0), p=new A(4,5)

[解析] 构造函数带默认参数为0，使用new运算符动态分配对象空间，同时初始对象成员值

4，5。

2. 在下面程序横线处填上适当内容，使程序执行结果为：

```

S=2

S=5

S=9

#include <iostream.h>

void sum(int i)

{static int s;

_____;

cout<<"s="<<s<<endl;

}

void main (void)

{int i;

for (i=0;_____)

sum(i);

}

```

答案: s=s+i+2;, i<3,i++

[解析] 根据结果和调用形式，得出规律。注意静态成员能保留上次运行的结果。循环了3次

, 退出循环的条件。

3. 下面程序运行的结果是：5+10=15。

```
#include <iostream.h>

class Test

{ private:

int x,y;

public:

Test() {x=y=0;}

void Setxy(int x,int y) {_____}

void show(){_____}

};

void main()

{Test ptr;

ptr.Setxy(5,10);

ptr.show();

}
```

答案：(*this).x=x; (*this).y=y;; cout<<x<<"+"<<y<<"="<<x+y<<endl;

【解析】形参同数据成员同名，使用this来访问。

4. 完成下面类中成员函数的定义。

```
#include <iostream.h>

#include <iomanip.h>

class Arr

{protected:

float *p;

int n;//数组大小 (元素个数)

public:

Arr(int sz=10)

{ n=sz;

p=new float [n] ;

}

~Arr(void)

{

_____

}

int Getn(void) const

{

return n;

}

float & operator [] (int i)

{

_____

}

void Print();

};
```

```

void Arr::Print()

{int i;

for(i=0;i< this->Getn();i++)

{if (i%10==0)

cout << endl;

cout<<setw(6)<<p [i] ;

}

cout<<endl;

}

void main()

{Arr a(20);

for (int i=0;i<a.Getn();i++)

a [i] =i* 2;

a.Print();

}

```

答案： delete p;, return p [i] ;

【解析】在析构函数中释放对象空间。第二个是对 [] 运算符的重载，函数返回类型是实型，形参i，取得下标为i的元素的值。

5. 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:

```

11,10

13,12

#include <iostream.h>

class A

{int a;

public:

A(int i=0){a=i;}

int Geta(){return a;}

void show(){cout<<a<<endl;}

};

class B

{A a;

int b;

public:

B(int i,int j)_____

{}

void show(){cout<<a.Geta()<<" "<<b<<endl;}

};

void main()

{B b [2] ={B(10,11),B(12,13)};

for(int i=0;i<2;i++)

_____

}

```

答案： :a(j),b(i), b [i] .show();

【解析】在构造函数中对数据成员初始化，从结果先输出a，后b，所以对a = j， b=i;在循环中

输出成员，调用show成员。

五、程序分析题(本大题共4小题，每小题5分，共20分)

1. 给出下面程序输出结果。

```
#include<iostream.h>

class a

{public:

a(int i=10){x=i;cout<<"a:"<<x<<endl;}

int x;

};

class b:public a

{public:

b(int i):A(i){x=i;cout<<"b:"<<x<<" "<<a::x<<endl;}

private:

a A;

int x;

};

void main()

{b B(5);

}
```

答案：a:10

a:5

b:5,10

【解析】定义对象B，先调用基类构造函数，在b构造函数中使用的是A(i)，注意大小写，不是a(i),也就是说调用基类的构造函数时没有实参值，所以采用默认值；在初始化类成员A，即A(i)，i = 5，所以输入为a:5；最后是b类的构造函数，x=5,来自基类x = 10，输出b:5,10。

2. 运行程序，写出程序执行的结果。

```
#include<iostream.h>

class Location

{public:

int X,Y;

void init(int initX,int initY);

int GetX();

int GetY();

};

void Location::init (int initX,int initY)

{X=initX;

Y=initY;

}

int Location::GetX()

{return X;

}

int Location::GetY()

{return Y;

}
```

```

void display(Location& rL)

{cout<<rL.GetX()<<" "<<rL.GetY()<< \ n;

}

void main()

{Location A [5] ={{5,5},{3,3},{1,1},{2,2},{4,4}};

Location *rA=A;

A [3] .init(7,3);

rA->init(7,8);

for (int i=0;i<5;i++)

display(*(rA++));

}

```

答案： 7 8

3 3

1 1

7 3

4 4

【解析】对象数组的使用。使用数组对象修改了A [3] 元素的值，又使用指针修改指针所指向的第一个元素的值，因此修改了A [0] 和A [3] 元素的值。

3. 给出下面程序输出结果。

```

#include <iostream.h>

int a [8] ={1,2,3,4,5,6,7};

void fun(int *pa,int n);

void main()

{int m=8;

fun(a,m);

cout<<a [7] <<endl;

}

void fun(int *pa,int n)

{for (int i=0;i<n-1;i++)

*(pa+7)+=*(pa+i);

}

```

答案： 28

【解析】数组名与指针都表示地址，只是数组名是常地址，不能改变；指针是地址变量，使用时可以当数组名使用。

4. 给出下面程序输出结果。

```

#include <iostream.h>

class A

{int *a;

public:

A(int x=0):a(new int(x)){}

~A() {delete a;}

int getA() {return *a;}

void setA(int x) {*a=x;}

};

```

```

void main()

{A x1,x2(3);

A *p=&x2;

(*p).setA(x2.getA()+5);

x1.setA(10+x1.getA());

cout<<x1.getA()<<" "<<x2.getA()<<endl;

}

```

答案： 108

【解析】 p指向对象x2， x2.getA()+5该值为8 即x2.a = 8;10+x1.getA()为10， x1.a = 10。

六、程序设计题(本大题共1小题，共10分)

1. 已知交通工具类定义如下。

要求：（1）实现这个类；（2）定义并实现一个小车类car，是它的公有派生类，小车本身的私有

属性有载人数，小车的函数有init(设置车轮数，重量和载人数)，getpassenger(获取载人数

), print（打印车轮数，重量和载人数）。

```

class vehicle

{protected:

int wheels;//车轮数

float weight;//重量

public:

void init(int wheels,float weight);

int get_wheels();

float get_weight();

void print();

};

void vehicle::init(int wheels,float weight)

{this->wheels=wheels;

this->weight=weight;

cout<<wheels<<endl;

}

int vehicle::get_wheels()

{return wheels;

}

float vehicle::get_weight()

{return weight;}

void vehicle::print()

{cout<<"车轮数: "<<wheels<<","<<"重量: "<<weight<<endl;}

```

答案： class car:public vehicle

```

{private:int passengers;

public:

void init(int wheels,float weight,int pass);

int getpassenger();

void print();};

void car::init(int wheels,float weight,int pass)

{vehicle::init(wheels,weight);

```

```
passengers=pass;}

int car::getpassenger()

{return passengers;}

void car::print()

{vehicle::print();

cout<<"可载人数: "<<passengers<<endl;

}
```

C++程序设计及答案解析（五）