

2010年全国自考C++程序设计模拟试卷(三)

一、单项选择题(本大题共20小题，每小题1分，共20分)在每小题列出的四个备选项中只有一个是符合题目要求的，请将其代码填写在题后的括号内。错选、多选或未选均无分。

1. 设有定义`int i; double j=5;`，则`10+i+j`值的数据类型是 ()

- A. `int`
- B. `double`
- C. `float`
- D. 不确定

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"` ;

3. 类的构造函数被自动调用执行的情况是在定义该类的 ()

- A. 成员函数时
- B. 数据成员时
- C. 对象时
- D. 友元函数时

4. 已知类A是类B的友元，类B是类C的友元，则 ()

- A. 类A一定是类C的友元
- B. 类C一定是类A的友元
- C. 类C的成员函数可以访问类B的对象的所有成员
- D. 类A的成员函数可以访问类B的对象的所有成员

5. 假定一个类的构造函数为“`A(int i=4, int j=0) {a=i;b=j;}`”，则执行“`A x (1);`”语句后，`x.a`和`x.b`的值分别为 ()

- A. 1和0
- B. 1和4
- C. 4和0
- D. 4和1

6. 关于`this`指针使用说法正确的是 ()

- A. 保证每个对象拥有自己的数据成员，但共享处理这些数据的代码
- B. 保证基类私有成员在子类中可以被访问。
- C. 保证基类保护成员在子类中可以被访问。
- D. 保证基类公有成员在子类中可以被访问。

7. 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

8. 友元关系不能 ()

- A. 提高程序的运行效率

B. 是类与类的关系

C. 是一个类的成员函数与另一个类的关系

D. 继承

9. 语句 `ofstream f("TEMP.DAT", ios::app | ios::binary)` 的功能是建立流对象 `f`，试图打开文件 `TEMP.DAT` 并与之连接，并且 ()

A. 若文件存在，将文件写指针定位于文件尾；若文件不存在，建立一个新文件

B. 若文件存在，将其置为空文件；若文件不存在，打开失败

C. 若文件存在，将文件写指针定位于文件首；若文件不存在，建立一个新文件

D. 若文件存在，打开失败；若文件不存在，建立一个新文件

10. 构造函数不具备的特征是 ()

A. 构造函数的函数名与类名相同

B. 构造函数可以重载

C. 构造函数可以设置默认参数

D. 构造函数必须指定类型说明

11. 在公有继承的情况下，基类的公有或保护成员在派生类中的访问权限 ()

A. 受限制

B. 保持不变

C. 受保护

D. 不受保护

12. 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. 3和15

B. 5和4

C. 4和20

D. 20和5

13. C++对C语言做了很多改进，即从面向过程变成为面向对象的主要原因是 ()

A. 增加了一些新的运算符

B. 允许函数重载，并允许设置缺省参数

C. 规定函数说明符必须用原型

D. 引进了类和对象的概念

14. 在类中说明的成员可以使用关键字的是 ()

A. public

B. extern

C. cpu

D. register

15. C++语言中所有在函数中定义的变量，连同形式参数，都属于 ()

A. 全局变量

B. 局部变量

C. 静态变量

D. 函数

16. 在私有继承的情况下，基类成员在派生类中的访问权限 ()

A. 受限制

B. 保持不变

C. 受保护

D. 不受保护

17. 使用地址作为实参传给形参，下列说法正确的是（）

A. 实参是形参的备份

B. 实参与形参无联系

C. 形参是实参的备份

D. 实参与形参是同一对象

18. C++的继承性允许派生类继承基类的（）

A. 部分特性，并允许增加新的特性或重定义基类的特性

B. 部分特性，但不允许增加新的特性或重定义基类的特性

C. 所有特性，并允许增加新的特性或重定义基类的特性

D. 所有特性，但不允许增加新的特性或重定义基类的特性

19. 对于 `int *pa[5];` 的描述，正确的是（）

A. pa是一个指向数组的指针，所指向的数组是5个int型元素

B. pa是一个指向某个数组中第5个元素的指针，该元素是int型变量

C. pa[5]表示某个数组的第5个元素的值

D. pa是一个具有5个元素的指针数组，每个元素是一个int型指针

20. 以下基类中的成员函数表示纯虚函数的是（）

A. `virtual void tt()=0`

B. `void tt(int)=0`

C. `virtual void tt(int)`

D. `virtual void tt(int){}`

二、填空题（本大题共20小题，每小题1分，共20分）请在每小题的空格中填上正确答案。错填、不填均无分。

1. 单目运算符作为类成员函数重载时，形参个数为__0__个。

2. 抽象类中至少要有有一个__纯虚__函数。

3. 设类A有成员函数 `void f(void)`；若要定义一个指向类成员函数的指针变量pf来指向f，该指针变量的声明语句是：__`void (A::*pf)(void);`__。

4. 执行下列程序

```
double a=3.1415926, b=3.14;  
cout<<setprecision(5)<<a<<"", "<<setprecision(5)<<b<<endl;  
程序的输出结果是3.1416, 3.14__。
```

5. vector类中用于删除向量中的所有对象的方法是__。

6. 重载的运算符保持其原有的__操作数__、优先级和结合性不变。

7. 编译时的多态性通过__重载__函数实现。

8. 基类的公有成员在派生类中的访问权限由__继承方式__决定。

9. 假设类X的对象x是类Y的成员对象，则“Y Obj”语句执行时，先调用类__X__的构造函数。

10. 下列程序段的输出结果是__。

11. 下列程序段的输出结果是__。

12. C++中ostream的直接基类__ios__。

13. `int n=0;`

`while (n=1) n++;`

while循环执行次数是__无限次__。

14. C++中有两种继承：单继承和__多重继承__。

15. 在C++中，利用向量类模板定义一个具有10个int的向量A，其元素均被置为1，实现此操作的语句是___。

16. vector类中向向量尾部插入一个对象的方法是___。

17. C++语言中如果调用函数时，需要改变实参或者返回多个值，应该采取引用___方式。

18. 若函数的定义处于调用它的函数之前，则在程序开始可以省去该函数的_声明_语句。

19. 在C++中有两种参数传递方式：传值和_引用___。

20. 将指向对象的引用作为函数的形参，形参是对象的引用，实参是 _对象名___。

三、改错题（本大题共5小题，每小题4分，共20分）

1. class ABC

{int a;

public:

ABC(int aa)a(aa){}

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

};

2. #include <iostream.h>

class Aton

{int X,Y;

protected:

int zx,zy;

public:

void init(int i,int j){zx=i;zy=j;}

Aton(int i,int j,int n=0,int m=0)

{X=i,Y=j,zx=m,zy=n;

}

添加inline int X(){return X};

};

void main()

{Aton A(25,20,3,5);

A.init(5,9);

cout<<A.X()<<endl;

3. #include <iostream.h>

class Bas

{public:

~Bas(){cout<<"Bas construct"<<endl;}

virtual void f()=0;

};

class Dev:public Bas

{public:

~Dev(){cout<<"Bas construct"<<endl;}

virtual void f(){cout<<"Dev::f"<<endl;}

};

void main()

{Bas *a=new Bas();

Bas *a;

Dev p;

a=&p;

a->f();

```
}
```

4. 以下程序实现交换a,b变量的值, 请用下横线标出错误所在行并给出修改意见。

```
#include <iostream.h>
void swap(int &a,int &b)
{a=a+b;
b=a-b;
a=a-b;
}
void main()
{int a=19,b=15;
cout<<"a="<<a<<"",b="<<b<<endl;
swap(&a,&b);          swap(a,b);
cout<<"a="<<a<<"",b="<<b<<endl;
}
```

```
5. #include <iostream.h>
void main()
{int i(3),j(8);
int * const p=&i;      int *p=&i;
cout<<*p<<endl;
p=&j;
cout<<*p<<endl;
}
```

四、完成程序题(本大题共5小题, 每小题4分, 共20分)

1. 在下面程序横线处填上适当内容, 使程序执行结果为: "hello, andylin".

```
#include <iostream>
#include <string.h>
using namespace std;
class mystring
{public:
char * pdata;
mystring(int len)
{pdata=new char [len+1] ;
}
~mystring()
{delete pdata;}
void show() {cout<<pdata<<endl;}
};
void fun(mystring** array,int len)
{mystring*old=*array;
_ *array=new mystring(len) _____;
memcpy(*array, old, len);
}
```

```

}
void main()
{mystring str(20);
mystring*pstr=&str;
mystring**ppstr=&pstr;
strcpy(str.pdata, "hello, andylin");
fun(ppstr, 20);
str.show(); _____
}

```

2. 在下面程序横线处填上适当字句，完成类的定义。

```

class line;
class box
{ private:
int color;
int upx, upy;
int lowx, lowy;
public:
friend int same_color(line l, box b);
void set_color (int c) {color=c;}
void define_box (int x1, int y1, int x2, int y2)
{upx=x1;upy=y1;lowx=x2;lowy=y2;}
};
class line
{ private:
int color;
int startx, starty;
int endx, endy;
public:
friend int same_color(line l, box b);
void set_color (int c) {color=c;}
void define_line (int x1, int y1, int x2, int y2)
{startx=x1;starty=y1;endx=x2;endy=y2;}
};
int same_color(line l, box b)
{if (l.color==b.color) return 1;
return 0;
}

```

3. 下面程序用来求直角三角形斜边长度。

```

#include <iostream.h>
#include <math.h>
class Point
{private:
double x,y;
friend Line; _____
public:

```

```

Point(double i=0,double j=0)
{ x=i;y=j; }
Point(Point &p)
{ x=p.x;y=p.y; }
};
class Line
{ private:
Point p1,p2;
public:
Line(Point &xp1,Point &xp2):p1(xp1),p2(xp2) _____ {}
double GetLength();
};
double Line::GetLength()
{ double dx=p2.x-p1.x;
double dy=p2.y-p1.y;
return sqrt(dx*dx+dy*dy);
}
void main()
{ Point p1,p2(6,8);
Line L1(p1,p2);
cout<<L1.GetLength()<<endl;
}

```

4. 在下面程序的底画线处填上适当的字句，使该程序执行结果为40。

```

#include <iostream.h>
class Test
{ public:
static int x_____;
Test (int i=0)
{ x=i+x; }
int Getnum()
{ return Test::x+7; }
};
int Test::x=33 _____;
void main()
{ Test test;
cout<<test.Getnum()<<endl;;
}

```

5. 在下列程序的空格处填上适当的字句，使输出为：0，2，10。

```

#include <iostream.h>
#include <math.h>
class Magic
{ double x;
public:
Magic(double d=0.00):x(fabs(d))
{}

```

```

Magic operator+(Magic &c_____)
{
return Magic(sqrt(x*x+c.x*c.x));
}
friend ostream& operator<<(ostream & stream, Magic & c)
{ stream<<c.x;
return stream;
}
};
void main()
{Magic ma;
cout<<ma<<"", "<<Magic(2)<<"", "<<ma+Magic(-6)+
Magic(-8)<<endl;
}

```

五、程序分析题(本大题共2小题，每小题5分，共10分)

1. 运行程序，写出程序执行的结果。

```

#include <iostream.h>
void main()
{int a,b,c;
char ch;
cin>>a>>ch>>b>>c;//从键盘上输入1.5×c×10×20, ×表示一个空格
cout<<a<<endl<<ch<<endl<<b<<endl<<c<<endl;
}

```

1

.

5

0

2. 给出下面程序输出结果。

```

#include <iostream.h>
class A
{public:
A()
{cout<<"As cons."<<endl;}
virtual ~A()
{cout<<"As des."<<endl;}
virtual void f()
{cout<<"As f()."<<endl;}
void g()
{f();}
};
class B:public A
{public:
B()
{f();cout<<"Bs cons."<<endl;}
~B()
{cout<<"Bs des."<<endl;}
};

```



```

class C:public B
{public:
C()
{cout<<"Cs cons."<<endl;}
~C()
{cout<<"Cs des."<<endl;}
void f()
{cout<<"Cs f()."<<endl;}
};

void main()
{A *a=new C;
a->g();
delete a;
}

As cons.
As f().
Bs cons.
Cs cons.
Cs f().
Cs des.
Bs des.
As des.

```

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;}

```

2009年全国自考C++程序设计模拟试卷(四)

一、单项选择题(本大题共20小题，每小题1分，共20分)在每小题列出的四个备选项中只有一个是符合题目要求的，请将其代码填写在题后的括号内。错选、多选或未选均无分。

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

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" ;`

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

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”);

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所指的物体类型而确定调用类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

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

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

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

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#ifdef后的常量表达式是否为真，该部分都需要编译

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

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的物体引用，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

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

13. 在编译指令中，宏定义使用哪个指令（）

A. #if

B. #include

C. #define

D. #error

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

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

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

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

18. 下列运算符中，在C++语言中不能重载的是（）

A. *

B. >=

C. ::

D. /

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

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

二、填空题(本大题共20小题，每小题1分，共20分)请在每小题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 string.h。
2. 在函数前面用 inline 保留字修饰时，则表示该函数为内联函数。
3. 向量操作方法中 at 方法返回向量中的第一个对象。
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 性质扩展，即增加派生类的性质。
5. 重载的运算符保持其原有的 操作数、优先级和结合性不变。
6. 编译时的多态性通过 重载 函数实现。
7. 预处理语句有三种，分别是宏定义、文件包含和 条件编译。
8. 构造函数、析构函数和友元函数中，不是该类成员的是 友元函数。
9. 控制格式输入输出的操作中，函数 setw(int) 是设置域宽的。要求给出函数名和参数类型。
10. 派生类的成员一般分为两部分，一部分是 从基类继承的成员，另一部分是自己定义的新成员。
11. C++中ostream的直接基类 ios。
12. vector的 back 方法返回向量中的最后一个对象。
13. 执行下列代码
int i=230;
cout <<"i="<<hex <<i<<endl;
程序的输出结果为 i=E6。
14. 在C++中有两种参数传递方式即值传递和 引用 传递。
15. 使用new为int数组动态分配10个存储空间是 new int [10]；。
16. 面向对象的四个基本特性是多态性、继承性、和封装性 抽象。
17. 定义虚函数所用的关键字是 virtual。
18. 执行下列代码
cout<< "oct:" <<oct<<34;
程序的输出结果是 oct:42。
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 ifstream fin(“Test.txt”);。
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 单继承。

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) {}          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;
}

```

```

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;}
friend 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);
}

```

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=d<c?d:c);
}
void main()
{Test t1;
t1.setValue(34,6,2);
cout<<t1.getmin ()<<endl;          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;
}
```

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:
A (int i=0,int j=0) ____; //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;
p=new A(4,5) ____; //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
}
```

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;
_s=s+i+2_____};
cout<<"s="<<s<<endl;
}
void main (void)
{int i;
for (i=0;i<3;i++_____)
sum(i);
}

```

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) {(*this).x=x;(*this).y=y_____}
void show() {cout<<x<<"+"<<y<<"="<<x+y<<endl;_____}
};
void main()
{Test ptr;
ptr.Setxy(5,10);
ptr.show();
}

```

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)
{
_delete p;_____
}
int Getn(void) const

```



```

{
return n;
}
float & operator [] (int i)
{
return p[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();
}

```

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):a(j),b(i)_____
{}
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++)
b[i].show()_____
}

```

```
}
```

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++));
}

```

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);
}

```

28

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;
}

```

108;

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;}
```