

2008-2009 学年 C++（下）期末考试模拟试题

一、选择题（每题 2 分，共 30 分）

1. 下列程序段的运行结果是（ C ）。Strlen 函数不包括\0  
char c[]="t\0will\n";cout<<strlen(c);  
A. 4      B. 3      C. 9      D. 字符串中含有非法字符，输出结果不确定
2. 已知定义类 A，该类的构造函数的参数都没有缺省值。执行语句：  
A a1(2),a2(1,2), \*p=new A[3];  
则自动调用该类缺省的构造函数的次数为（D）。  
A. 1                      B. 2                      C. 3                      D. 5
3. 下列叙述正确的是（D）。  
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. 包含虚函数的基类为虚基类  
D. 包含存纯函数的基类为抽象类
4. 下列关于赋值兼容性规则的叙述不正确的是（A）。  
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对象的引用  
D. 可以将派生类对象的地址赋值给其基类的指针变量
5. 设有定义 int a[][3]={1,3,5,7,9,11,13,15,17};int \*p=a[2];则\*(p+1)的值为（ A）。  
A. 15                      B. 7                      C. 14                      B. 6
6. C++中多态性包括两种：编译时的和运行时的。运行时多态性是通过（ B ）实现的。  
① 函数重载    ② 运算符重载    ③ 类继承关系    ④动态链接    ⑤ 虚函数  
A. ①和②      B. ③和⑤      C. ④和⑤      D. ③和④
7. 关于动态内存分配，对 delete 运算符的下列说法中，（ C ）是错误的。  
A. 应与 new 运算符配合使用。  
B. 可以在类的成员函数中使用。  
C. 对同一个指针变量可任意多次使用该运算符。  
D. 若 a 是一个二维数组，则 delete []a; 可删除 a 所占用的存储空间。
8. 设 int x; int \*p2=&x, \*p1=&x; \*p1=a; \*p2=\*b; 则变量 a 和 b 的类型分别是（ c ）。  
A. int 和 int      B. int \*和 int      C. int 和 int \*      D. int \*和 int \*
9. 下列关于成员函数的叙述不正确的是（ c ）。  
A. 成员函数一定是内联函数  
B. 成员函数可以重载  
C. 成员函数可以设置参数的默认值  
D. 成员函数可以是静态的

10. 以下不属于类的访问权限的是 ( c )。
- A. public                  B. protected                  C. static                  D. private
11. 若要使用 ifstream 流定义一个流对象并打开一个文件时,应在程序开始使用#include 命令,使之包含 (B)。
- A. iostream.h                  B. fstream.h                  C. stdio.h                  D. strstream.h
12. 下列叙述不正确的是 ( D )。
- A. 对象之间可以相互赋值                  B. 对象可以用作函数参数  
C. 对象可以用作数组的元素                  D. C++中可以有指向对象的指针
13. C++中不能被派生类继承的是 ( B )。
- A. 虚函数                  B. 析构函数和构造函数  
C. 静态成员函数                  D. 运算符重载函数
14. 不属于构造函数的特点是 (B)。
- A. 具有与类名相同的函数名                  B. 返回类类型返回值  
C. 允许重载                  D. 在定义类的对象时自动调用
15. 类型相同的两个指针不能进行的运算是 ( A )。
- A. +                  B. -                  C. =                  D. ==

## 二、填空题 (每题 2 分, 共 20 分)

- 函数参数传递方式有三种, 分别为: 值传递、地址传递和 ( 引用传递 )。
- 面向对象的程序设计的三大特征是: ( 封装)、继承性和多态性。
- 假定 A 为一个类, 则语句 A(A &x);是此类 (拷贝 (复制)) 构造函数的原型说明。
- 在类的继承与派生中, 缺省的继承方式是 ( 私有继承)。
- C++的两种多态性分别是 ( 编译时) 多态性和 ( 运行时) 多态性。
- 运算符重载有两种实现方法, 一种是通过友元函数来实现, 另一种通过 (成员 ) 函数来实现。
- 根据数据的组织形式不同, 可将文件分为二进制文件和 ( 文本 ) 文件。
- 若希望类中的成员只能被该类的成员函数及该类的派生类的成员函数访问, 则应加上访问权限: (protected)。
- 假定类 A 中有一个公用属性的静态数据成员 b, 在类外不通过对象名访问该成员 b 的写法为 (A::b)。

## 三、阅读程序题 (每题 5 分, 共 30 分)

1. 下列程序运行结果为 ( 3 )。

```
#include <iostream.h>
#include <string.h>
char s(char *w,int y)
{
    char t, *s1, *s2;
```

```

s1=w;
s2=w+y+5;
(s1<s2)
while(s1>s2)
{
    t=*s1++;*s1=*s2--;*s2=t;
    return *s1;
}
return *s1+2;
}
void main(void)
{
    char *p={"123"};
    cout<<s(p,strlen(p));
}

```

2. 下列程序的输出结果是 ( x=11 )。

```

#include <iostream.h>
class A
{
public:
    int X;
    A(int x){X=x*x;}
};
class S
{ int x;
  A a;
public:
    S(int b,int c):a(c)
    {x=b+a.X;}
    void show(){cout<<"x="<<x<<endl;}
};
void main(void)
{ S s1(2,3);
  s1.show();
}

```

3. 下列程序的运行结果是 ( ABAC )。

```

#include <iostream.h>
class A
{
public:
    int n;
    A(){cout<<"A";}
}

```

```

};
class B:public A
{
public:
    B(){cout<<"B";}
};
class C:public B
{
    A a;
public:
    C():a(),B(){cout<<"C";}
};
void main(void)
{ C c;
}

```

4. 下列程序的运行结果是 ( 50,101  
20,102 )

```

#include <iostream.h>
class Base
{
protected:
    int *p;
public:
    Base(int a=0)
    {p=new int (a);}
    ~Base()
    {delete p;}
    virtual void print()
    {cout<<"p->"<<*p<<endl;}
};
class Derived:public Base
{
    static int y;
public:
    Derived(int b=0)
    { *p=b;y++;}
    void print()
    {cout<<*p<<','<<y<<endl;}
};
int Derived::y=100;

void main(void)
{

```

```

    Derived d(50);
    Base *pb=&d;
    pb->print();
    Derived d1(20);
    d1.print();
}

```

5. 下列程序的运行结果是 ( 8        7 )

```

#include <iostream.h>
class B
{
    int a,b;
public:
    B(int aa=0,int bb=0){a=aa;b=bb;}
    void operator ++(); //++a
    void operator ++(int); //a++,只能是 int
    void show(){ cout<<a<<"\t"<<b;}
};

void B:: operator ++()
{
    a+=2;
    b+=5;
}
void B:: operator ++(int)
{
    a+=5;
    b+=2;
}
void main(void)
{
    B x(3,5);
    x++;
    x.show();
}

```

6. 下列程序的运行结果是 (0        1)。

```

#include <iostream.h>
class A
{
public:
    A(int n){num=n;}
    int compare(A a)

```

```

    {
        if(this->num==a.num)return 1;
        else return 0;
    }
private:
    int num;
};

void main(void)
{
    A aa(5);
    A bb(10);
    A cc(5);
    cout<<aa.compare(bb)<<"t";
    cout<<cc.compare(aa)<<endl;
}

```

#### 四、完善程序（每空 2 分，共 20 分）

1. 下面的程序中定义了圆类 **Circle**、长方形类 **Rect**、圆角长方形类 **RoundRect**，其中 **RoundRect** 共有继承于 **Circle** 和 **Rect** 类。试完成程序。

```

#include <iostream.h>
#include <string.h>
class Circle
{
    double radius;
public:
    Circle(double r):_____ { }
    double area(){return 3.14*radius*radius;}
};
class Rect
{
    double width, height;
public:
    Rect(double w, double h){ width=w; height=h;}
    double area(){return height*width;}
};
class RoundRect: public Circle,_____ Rect
{
    char color[5];
public:
    RoundRect(_____):Circle(r),Rect(w,h)
    {
        strcpy(color,c);
    }
}

```

```

    char *getcolor(){return color;}
};
void main(void)
{
    RoundRect rr(0.8,1,0.25,"白色");
    cout<<rr.getcolor()<<endl;
}

```

2. 下列程序通过重载运算符“\*”，直接实现两个一维数组对应元素相乘运算。设数组 a，b 分别为：

```
int a[10]={1,2,3,4,5,6,7,8,9,10};
```

```
int b[10]={1,2,3,4,5,6,7,8,9,10};
```

相乘后的结果为{1,4,9,16,25,36,49,64,81,100}。同时通过重载运算符“=”，直接实现两个一维数组的赋值运算。试完善程序。

```

#include <iostream.h>
#include <string.h>
class A
{
    int x[10];
public:
    A(){for(int i=0;i<10;i++)x[i]=0;}
    A(int *p)
    {
        for(int i=0;i<10;i++)_____
    }
    A operator *(A a)
    {
        A t;
        for(int i=0;i<10;i++)_____
        return t;
    }
    A operator =(A a)
    {
        for(int i=0;i<10;i++)_____
        return *this;
    }
    void show(){for(int i=0;i<10;i++)cout<<x[i]<<'\\t';}
};

```

```

void main(void)
{
    int a[10]={1,2,3,4,5,6,7,8,9,10};
    int b[10]={1,2,3,4,5,6,7,8,9,10};
    A a1(a),a2(b),a3;
}

```

```

    a3=a1*a2;
    a3.show();
}

```

3. 下面的程序的功能是比较两个字符串是否相等，若相等返回 1，否则返回 0。请完成程序。

```

#include <iostream.h>
int f(char *s1, char *s2)
{
    char *p=s1,*q=s2;
    while(_____)
    {
        q++;
        p++;
    }
    return (_____);
}

void main(void)
{
    char s1[100],s2[100];
    cin.getline(s1,100);
    cin.getline(s2,100);
    cout<<f(s1,s2);
}

```

4. 下列函数的功能是将一个新生成的链表按照从小到大的顺序插入到现有的链表中，请完成函数。

```

struct node{
    int data;
    node * next;
};

```

node \* Insert(node \*head, node \*p)//将 p 指向的结点插入到 head 代表的链表中

```

{
    node *p1,*p2;
    if(head==0){//空链表,插入到链表首部
        head=p;
        _____
        return head;
    }
    if(head->data>=p->data){//非空链表,p 的数据项比首节点的数据项小,插入到链首
        p->next=head;
        _____
    }
}

```



```

        return head;
    }

    p2=p1=head;
    while(_____){//找到要插入的位置
        p1=p2;p2=p2->next;
    }
    if(p2->data<p->data){//p 代表的结点的数据项最大,插入到表尾
        p2->next=p;p->next=0;
    }
    else{//插入到 p1 和 p2 指向的结点之间
        _____
    }
}

```

#### 五、上机编程题（每题 50 分，共 100 分）

1. 建立一个 `str` 类，完成对字符串循环左移的功能。即从第二个字符开始，以后的每个字符都依次前移一个字符，而原来的第一个字符放在串中的最后一个字符。当要平移 `m` 个字符时，则要进行 `m` 次的循环左移。例如：“ABCD”循环左移 2 个字符后为“CDAB”。要求如下：

(1)私有数据成员

`char * w;`指向待左移字符串。

`int m, len;` 分别用于存储循环左移的次数及字符串的长度。

(2)公有成员函数

构造函数 `str(char *p, int n, int k)`：初始化各数据成员。

`void move()`：循环左移。

`void show()`：输出字符串数据成员

(3)编写一个程序测试该类，使用以下测试数据：`a[]="ABCD"`，左移 3 个字符。要求输出字符串。

2. 请将压缩包中的文本文件 `in.dat` 的内容读入到数组 `A[3][3]`中，并将转置后的数组 `A` 的内容写入到文本文件 `out.dat` 中。请完成程序。