

北京工业大学 2022—2023 学年第 1 学期

《面向对象程序设计(C++)》考试试卷 A 卷

考试说明：考试时间：95 分钟 考试形式（开卷/闭卷/其它）：闭卷

适用专业：软件工程专业

承诺：

本人已学习了《北京工业大学考场规则》和《北京工业大学学生违纪处分条例》，承诺在考试过程中自觉遵守有关规定，服从监考教师管理，诚信考试，做到不违纪、不作弊、不替考。若有违反，愿接受相应的处分。

承诺人：_____ 学号：_____ 班号：_____

注：本试卷共 四 大题，共 11 页，满分 100 分，考试时必须使用卷后附加的统一草稿纸，并将答案写在题目下方，如因答案写在其他位置而造成的成绩缺失由考生自己负责。

卷面成绩汇总表（阅卷教师填写）

题号	一	二	三	四	总成绩
满分	10	25	20	45	
得分					

得分

一、单项选择题（本大题共 10 小题，每小题 1 分，共 10 分）

在每小题列出的四个选项中只有一个选项是符合题目要求的，请将正确选项前的字母填在题后的括号内。

1. 下面关于类的描述，()是错误的。

- A. 类是抽象数据类型的实现 B. 类是具有共同行为的若干对象的统一描述体
C. 类是创建对象的模板 D. 类就是 C 语言中的结构类型

2. 下列语句错误的是()。

- A. `const int a;` B. `const int a=10;`
C. `const int *point=0;` D. `const int *point=new int(10);`

3. 对类的构造函数和析构函数描述正确的是()。

- A. 构造函数不能重载，析构函数可以重载

- B. 构造函数可以重载，析构函数不能重载
- C. 构造函数不能重载，析构函数也不能重载
- D. 构造函数可以重载，析构函数也可以重载
4. 下面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共享的数据
- B. 静态数据成员不能通过类的对象调用
- C. 类的不同对象有不同的静态数据成员值
- D. 类的每一个对象都有自己的静态数据成员
5. 下列运算符中，()运算符在 C++中不能用友元函数重载。
- A. && B. == C. = D. +
6. 派生类的对象对它的基类成员中()是可以访问的。
- A. 公有继承的保护成员 B. 公有继承的私有成员
- C. 私有继承的公有成员 D. 公有继承的公有成员
7. C++中，与实现运行时的多态性无关的是()。
- A. 重载函数 B. 虚函数 C. 指针 D. 引用
8. 下列描述中，()是抽象类的特征。
- A. 可以说明虚函数 B. 可以进行构造函数的重载
- C. 可以定义友元函数 D. 不能说明其对象
9. 多继承派生类构造函数构造对象时，()被最先调用。
- A. 派生类自己的构造函数 B. 虚基类的构造函数
- C. 非虚基类的构造函数 D. 派生类中子对象类的构造函数
10. 已知函数 f 的原型是 void f(int *a, long &b);，变量 v1、v2 的定义是：int v1; long v2; 下列调用语句正确的是()。
- A. f(v1, &v2); B. f(v1, v2); C. f(&v1, &v2); D. f(&v1, v2);

得分

二、阅读下列程序，写出执行结果（本大题共 5 小题，每小
题 5 分，共 25 分）

1. 写出下面程序的输出结果。

```
#include <iostream>
using namespace std;
```

资料由公众号【工大喵】收集整理并免费分享

```
class test{
    int k;
public:
    static int n;
    test(int kk)
    {k=kk;n++;}
    void disp()
    {cout<<"n="<<n<<"    k="<<k<<endl;}
};
int test::n=0;
int main(){
    test t1(10);
    t1.disp ();
    test t2(20);
    t2.disp ();
    test::n++;
    t2.disp ();
    return 0;
}
```

运行结果为:

2. 写出下面程序的输出结果。

```
#include <iostream>
using namespace std;
class A{
public:
    A(){cout<<"A`s cons."<<endl;}
    virtual ~A(){cout<<"A`s des."<<endl;}
    virtual void f(){cout<<"A`s f()."<<endl;}
    void g(){f();}
};
class B:public A{
public:
    B(){f();cout<<"B`s cons."<<endl;}
    ~B(){cout<<"B`s des."<<endl;}
};
class C:public B{
public:
    C(){f();cout<<"C`s cons."<<endl;}
    ~C(){cout<<"C`s des."<<endl;}
```

```
        void f(){cout<<"C`s f()."<<endl;}
    };
int main()
{
    A *a=new C;
    a->g();
    delete a;
    return 0;
}
```

运行结果为:

3. 写出下面程序的输出结果。

```
#include <iostream>
using namespace std;
class sample{
    int x;
public:
    sample(){ }
    void setx(int i){x=i;}
    friend int fun(sample B[],int n)
    {
        int m=0;
        for(int i=0;i<n;i++)
            if(B[i].x>m) m=B[i].x;
        return m;
    }
};
int main()
{
    sample A[10];
    int Arr[]={90,87,42,78,97,84,60,55,78,65};
    for(int i=0;i<10;i++)
        A[i].setx (Arr[i]);
    cout<<fun(A,10)<<endl;
    return 0;
}
```

运行结果为:

资料由公众号【工大喵】收集整理并免费分享

4. 写出下面程序的输出结果。

```
#include <iostream>
using namespace std;
int f(int i)
{
    return ++i;
}
int g(int &i)
{
    return ++i;
}
int main()
{
    int a,b;
    a=b=0;
    a+=f(g(a));
    b+=f(f(b));
    cout<<"a="<<a<<"    b="<<b<<endl;
    return 0;
}
```

运行结果为：

5. 写出下面程序的输出结果。

```
#include <iostream>
using namespace std;
class point
{
    int x,y;
public:
    point(){}
    point(int l,int w){x=l;y=w;}
    void disp() {cout<<"面积: "<<x*y<<endl;}
    point operator,(point r)
    {
        point temp;
        temp.x=r.x;
        temp.y=r.y;
        return temp;
    }
    point operator+(point r)
    {
        point temp;
        temp.x=r.x+x;
```

资料由公众号【工大喵】收集整理并免费分享

```
        temp.y=r.y+y;
        return temp;
    }
};
int main()
{
    point r1(3,3),r2(5,8),r3(2,4);
    r1.disp ();
    r2.disp ();
    r3.disp ();
    r1=(r1,r2+r3,r3);
    r1.disp ();
}
```

运行结果为:

得分

三、按要求完成下列各题(共 20 分)

1.阅读以下程序:(每小题 3 分,共 12 分)

```
#include <iostream>
using namespace std;
class A
{
public:
    void f1();
    A(){i1=10;j1=11;}
protected:
    int j1;
private:
    int i1;
};
class B:private A
{
public:
    void f2();
    B(){i2=20;j2=21;}
protected:
    int j2;
private:
```

资料由公众号【工大喵】收集整理并免费分享

```
    int i2;
};
class C:private B
{
public:
    void f3();
    C(){i3=30;j3=31;}
protected:
    int j3;
private:
    int i3;
}
```

回答下列问题:

- (1) 派生类 B 中成员函数 f2()能否访问基类 A 中的成员 f1()、i1 和 j1?
- (2) 派生类 B 的对象 b 能否访问基类 A 中的成员 f1()、i1 和 j1?
- (3) 派生类 C 中成员函数 f3()能否访问直接基类 B 中的成员 f2()和 j2? 能否访问间接基类 A 中的成员 f1()、i1 和 j1?
- (4) 派生类 C 的对象 c 能否访问直接基类 B 中的成员 f2()、i2 和 j2? 能否访问间接基类 A 中的成员 f1()、i1 和 j1?

解:

2. 阅读下面类的定义, 如果正确, 写出运行结果, 如果错误找出程序中的错误, 并说明错误原因。(共 8 分)

```
#include <iostream>
using namespace std;
class sample
{
    char ch1,ch2;
public:
    friend void set(sample &s,char);
};
void set(sample &s,char c)
{
    s.ch1=c;
```

资料由公众号【工大喵】收集整理并免费分享

```
}  
void set(sample &s,char c1,char c2)  
{  
    s.ch1=c1;  
    s.ch2=c2;  
}  
int main()  
{  
    sample obj;  
    set(obj,5);  
    set(obj,7,9);  
    return 0;  
}
```

解:

得分

四、程序设计题（本大题共 4 小题，共 45 分）

注意：必须书写主函数，并给出必要的代码注释。

1. 设计一个类 Rect，要求有下列成员函数：（共 11 分）

- （1）Move()：从一个位置移动到另一个位置。
- （2）Size()：改变矩形的大小。
- （3）Where()：返回矩形左上角的坐标值。
- （4）Area()：计算矩形面积。

2. 编写一个程序，输入 3 个学生的英语和计算机成绩，并按总分从高到低排序（排序算法不限）。要求设计一个学生 Student，其定义如下：（共 11 分）

```
class Student
{
    int english, computer, total;
public:
    void getscore(); //获取一个学生成绩
    void display();  //显示一个学生成绩
    .....
};
#include <iostream>
using namespace std;
class Student{
    int english, computer, total;
public:
    void getscore();//获取该学生的所有成绩
    void display();//输出该学生的所有成绩
    void swap(Student &);//交换两个学生的成绩
    bool compare(Student &);//比较两个学生的成绩，this 大于参数的话返回 1
};
void Student::getscore() {
    cout<<"请输入该学生的英语成绩： ";
    cin>>english;
    cout<<"请输入该学生的计算机成绩： ";
    cin>>computer;
    total=english+computer;
}
void Student::display() {
    cout<<"英语成绩为： "<<english<<endl;
    cout<<"计算机成绩为： "<<computer<<endl;
    cout<<"总分为： "<<total<<endl;
}
void Student::swap(Student &a) {
    Student temp;
    temp=a;
    a=*this;
    *this=temp;
}
bool Student::compare(Student &a) {
    return total<a.total;
```

资料由公众号【工大喵】收集整理并免费分享

```
}  
int main() {  
    Student a[3];  
    for(int i=0;i<3;i++)  
        a[i].getscore();  
    for(int i=0;i<2;i++) {  
        for(int j=0;j<2-i;j++)  
            if(a[j].compare(a[j+1])==1)  
                a[j].swap(a[j+1]);  
    }  
    for(int i=0;i<3;i++)  
        a[i].display();  
}
```

3. 设计一个时间类 Time, 包括 3 个数据成员, 时(h)、分(m)和秒(s), 另外包括存取各数

据成员和设置时间的成员函数，按上、下午各 12 小时或按 24 小时输出时间的成员函数，以及默认构造函数，默认时间值为 0 时 0 分 0 秒。（共 11 分）

```
#include <iostream>

using namespace std;

class Time{
    int h;
    int m;
    int s;
public:
    Time(int newh=0,int newm=0,int news=0){
        h=newh;m=newm;s=news;
    }

    void set(int newh,int newm,int news){
        h=newh;m=newm;s=news;
    }

    void display12(){
        int newh=h;
        if(h>12){newh=h-12;
        cout<<"现在是下午"<<newh<<":"<<m<<":"<<s<<endl;
        return;}
        cout<<"现在是上午"<<newh<<":"<<m<<":"<<s<<endl;
        return;
    }

    void display24(){
        cout<<"现在是"<<h<<":"<<m<<":"<<s<<endl;
    }
};

int main(){
    Time a(24,7,8);
    a.display12();
```

资料由公众号【工大喵】收集整理并免费分享

```
a.display24();  
a.set(1, 2, 3);  
a.display12();  
a.display24();  
}
```

4. 写一个程序，定义抽象类 Shape，由它派生出 5 个派生类：（圆形）、（正方形）、（矩形）、（梯形）、（三角形）。用虚函数分别计算几种图形面积，并求它们的和。要求用基类指针数

组，使它每一个元素指向一个派生类对象。（共 12 分）

```
#include <iostream>

using namespace std;

const double PI=3.1415926;

class shape{
    protected:
        double radius;
    public:
        shape() {}
        double virtual area()=0;
};

class circle:public virtual shape{
    public:
        circle(double radius1) {radius=radius1;}
        double area() {return PI*radius*radius;}
};

class square:public virtual shape{
    public:
        square(double radius1) {radius=radius1;}
        double area() {return radius*radius;}
};

class rectangle:public virtual shape{
    double height;
    public:
        rectangle(double radius1, double radius2) {radius=radius1;height=radius2;}
        double area() {return radius*height;}
};

class tria:public virtual shape{
    double height;
    public:
```

资料由公众号【工大喵】收集整理并免费分享

```
        tria(double radius1, double radius2) {radius=radius1;height=radius2;}

        double area() {return 0.5*radius*height;}

};

class tri:public virtual shape{

    double down;

    double height;

    public:

        tri(double                radius1, double                radius2, double

h) {radius=radius1;down=radius2;height=h;}

        double area() {return (radius+down)*height/2;}

};

int main() {

    shape *p[5];

    circle c(5);square s(5);rectangle r(5,3);tria t1(5,3);tri t2(5,3,8);

    p[0]=&c;p[1]=&s;p[2]=&r;p[3]=&t1;p[4]=&t2;

    double total=0;

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

        total+=p[i]->area();

    cout<<total;

}
```