

## C++面向对象程序设计模拟试题八

一、单项选择题 (本大题共 10 小题, 每小题 2 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。

- 关于函数重载, 下列叙述中错误的是( )。  
A) 重载函数的函数名都相同  
B) 重载函数必须在参数个数或类型上有所不同  
C) 重载函数的返回值类型必须不相同  
D) 重载函数的函数体可以相同
- 有以下函数模板:  
template <class ElemType>  
ElemType Square(const ElemType &x) { return x \* x; }  
其中 ElemType 是( )。  
A) 函数形参      B) 函数实参      C) 模板类型形参      D) 模板类型实参
- 下列选项中, 不属于面向对象程序设计特征的是( )。  
A) 继承性      B) 多态性  
C) 封装性      D) 类比性
- 在 C++ 中, 用于实现运行时多态性的是( )。  
A) 友元函数      B) 重载函数  
C) 静态成员函数      D) 虚函数
- 下列关于类和对象的叙述中, 错误的是( )。  
A) 一个类只能定义一个对象      B) 对象是类的具体实例  
C) 类是某一类对象的抽象      D) 类和对象的关系就像数据类型和变量的关系
- 下面 4 个关键字中, 用于说明虚函数的是( )。  
A) virtual      B) public      C) protected      D) private
- 在语句 “cin >> x;” 中, cin 是( )。  
A) C++ 的关键字      B) 类名  
C) 对象名      D) 函数名
- 运算符重载只能改变运算符原有的( )。  
A) 操作数类型      B) 操作数个数  
C) 优先级      D) 结合性
- 为取代 C 中带参数的宏, 在 C++ 中使用了( )。  
A) 重载函数      B) 内置函数  
C) 递归函数      D) 友元函数
- 有以下类声明:  
class MyClass { int num; };  
则 MyClass 类的成员 num 是( )。  
A) 公有数据成员      B) 公有成员函数      C) 私有数据成员      D) 私有成员函数

二、填空题 (本大题共 5 小题, 每小题 2 分, 共 10 分) 不写解答过程, 将正确的答案写在每小题的空格内。错填或不填均无分。

- 含有纯虚函数的类称为 ( )。

2. 声明静态成员使用的关键字是 ( )。
3. 一个全局函数应声明为类的 ( ) 函数才能访问这个类的 private 成员。
4. 重载运算符 “+” 的函数名为 ( )。
5. 声明全局函数为内置函数的关键字为 ( )。

**三、程序分析题（本大题共 6 小题，每小题 5 分，共 30 分）给出下面各程序的输出结果。**

1. 阅读下面程序，写出输出结果。

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

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

int main()
{
    A obj;
    cout << endl;

    return 0;
}
```

上面程序的输出结果为：

2. 阅读下面程序，写出输出结果。

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

class Test
{
private:
    static int count;

public:
    Test() { count++; }
    ~Test() { count--; }
    static int GetCount() { return count; }
};

int Test::count = 0;

int main()
```

```

{
    Test *p = new Test;
    cout << "count=" << Test::GetCount() << endl;
    delete p;
    cout << "count=" << Test::GetCount() << endl;

    return 0;
}

```

上面程序的输出结果为:

3. 阅读下面程序, 写出输出结果。

```

#include <iostream>
using namespace std;

template <class Type>
Type f(Type x)
{
    return x * 1.6;
}

int main()
{
    cout << f(1) << endl;
    cout << f(1.0) << endl;

    return 0;
}

```

上面程序的输出结果为:

4. 阅读下面程序, 写出在 VC++中运行时的输出结果。

```

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

double Sqrt(double x)
{
    if (x < 0) throw "被开方数为负";
    else return sqrt(x);
}

int main()
{

```

```

    try
    {
        cout << Sqrt(9) << endl;
        cout << Sqrt(-9) << endl;
    }
    catch(char *mess)
    {
        cout << "异常信息:" << mess << endl;
    }

    return 0;
}

```

上面程序的输出结果为:

5. 阅读下面程序, 写出输出结果。

```

#include <iostream>
using namespace std;

class A
{
public:
    virtual void f() const { cout << "A::f()" << endl; }
};

class B: public A
{
public:
    void f() const { cout << "B::f()" << endl; }
};

int main()
{
    B obj;
    A *p = &obj;
    p->f();

    return 0;
}

```

上面程序的输出结果为:

6. 阅读下面程序, 写出输出结果。

```

#include <iostream>

```

```

using namespace std;

class Array
{
private:
    int *elem;
    int size;

public:
    Array(int a[], int sz): elem(a), size(sz) {}
    int GetSize() const { return size; }
    int &operator[](int pos){ return elem[pos - 1]; }
};

int main()
{
    int a[] = {1, 2, 3, 4, 5, 6};
    Array obj(a, 5);

    for (int i = 1; i <= 5; i++) cout << obj[i] << " ";
    cout << endl;

    return 0;
}

```

上面程序的输出结果为:

#### 四、完成程序填空题（本大题共 4 个小题，每小题 3 分，共 12 分）下面程序都留有空白，请将程序补充完整。

1. 将如下程序补充完整。

```

#include <iostream>
using namespace std;

class A
{
private:
    int n;

public:
    A(int n) { ____[1]____ = n; }           // 初始化数据成员 n
    void Show() const { cout << n << endl; }
};

int main()

```

```

{

    A i(18);
    i.Show();

    return 0;
}

```

2. 将如下程序补充完整。

```

#include <iostream>
using namespace std;

class A
{
protected:
    int a;

public:
    A(int x) { a = x; }
    void Show() const { cout << "a:" << a << endl; }
};

class B: public A
{
private:
    int b;

public:
    B(int x, int y):___[2]___ { b = y; }          // 将 a 初始化为 x, b 初始化为 y
    void Show() const { cout << "a:" << a << endl << "b:" << b << endl; }
};

int main()
{
    B obj(5, 18);
    obj.Show();

    return 0;
}

```

3. 将如下程序补充完整。

```

#include <iostream>
using namespace std;

```

```

class A
{
public:
    void Show(int x = 0) const { cout << x; }
};

class B: public A
{
public:
    void Show(int x = 0) const
    {
        _____[3]_____ // 显式调用基类的函数 Show()
        cout << x << endl;
    }
};

int main()
{
    B obj;
    obj.Show(1);

    return 0;
}

```

4. 下列程序的输出结果为 1，试将程序补充完整。

```

#include <iostream>
using namespace std;

class A
{
public:
    _____[4]_____ Show() const { cout << 1 << endl; }
};

class B: public A
{
public:
    void Show() const { cout << 2 << endl; }
};

int main()
{
    A *p = new B;
    p->Show();
}

```

```
delete p;  
  
return 0;  
}
```

## 五、编程题（本大题共 2 小题，第 1 小题 12 分，第 2 小题 16 分，共 28 分）

1. 试使用函数模板实现求一个数组各元素的和，要求编写测试程序。

函数模板声明如下：

```
template <class Type>  
Type Sum(Type a[], int n);           // 求数组 a 各元素的和
```

2. 设计一个基类 Shape，Shape 中包含成员函数 Show()，将 Show()声明为纯虚函数。Shape 类公有派生矩形类 Rectangle 和圆类 Circle，分别定义 Show()实现其主要几何元素的显示。使用抽象类 Shape 类型的指针，当它指向某个派生类的对象时，就可以通过它访问该对象的虚成员函数 Show()，要求编写测试程序。



## C++面向对象程序设计模拟试题八参考答案

一、单项选择题 (本大题共 10 小题, 每小题 2 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. C) | 2. C) | 3. D) | 4. D) | 5. A)  |
| 6. A) | 7. C) | 8. A) | 9. B) | 10. C) |

二、填空题 (本大题共 5 小题, 每小题 2 分, 共 10 分) 不写解答过程, 将正确的答案写在每小题的空格内。错填或不填均无分。

1. 参考答案: 抽象类或抽象基类
2. 参考答案: static
3. 参考答案: 友元
4. 参考答案: operator+
5. 参考答案: inline

三、程序分析题 (本大题共 6 小题, 每小题 5 分, 共 30 分) 给出下面各程序的输出结果。

1. 参考答案:  
A  
~A
2. 参考答案:  
count=1  
count=0
3. 参考答案:  
1  
1.6
4. 参考答案:  
3  
异常信息:被开方数为负
5. 参考答案: B::f()
6. 参考答案: 1 2 3 4 5

四、完成程序填空题 (本大题共 4 个小题, 每小题 3 分, 共 12 分) 下面程序都留有空白, 请将程序补充完整。

1. 参考答案: [1] this->n 或 Test::n
2. 参考答案: [2] A(x)
3. 参考答案: [3] A::Show();
4. 参考答案: [4] void

五、编程题 (本大题共 2 小题, 第 1 小题 12 分, 第 2 小题 16 分, 共 28 分)

1. 参考程序:  

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

```

template <class Type>
Type Sum(Type a[], int n)           // 求数组 a 各元素的和
{
    Type s = 0;
    for (int i = 0; i < n; i++)
        s += a[i];
    return s;
}

int main()
{
    int a[] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
    cout << Sum(a, 9) << endl;

    return 0;
}

```

## 2. 参考程序:

```

#include <iostream>
using namespace std;

const double PI = 3.1415926;

class Shape
{
public:
    virtual ~Shape() { }
    virtual void Show() const = 0;
};

class Rectangle: public Shape
{
private:
    double height;
    double width;

public:
    Rectangle(double h, double w): height(h), width(w) { }
    void Show() const
    {
        cout << "矩形:" << endl;
        cout << "高:" << height << endl;
        cout << "宽:" << width << endl;
        cout << "周长:" << 2 * (height + width) << endl;
    }
}

```

```

        cout << "面积:" << height * width << endl << endl;
    }
};

class Circle: public Shape
{
private:
    double radius;

public:
    Circle(double r): radius(r) { }
    void Show() const
    {
        cout << "圆形:" << endl;
        cout << "半径:" << radius << endl;
        cout << "周长:" << 2 * PI * radius << endl;
        cout << "面积:" << PI * radius * radius << endl << endl;
    }
};

int main()
{
    Shape *p;
    p = new Circle(1);
    p->Show();
    delete p;
    p = new Rectangle(1, 2);
    p->Show();
    delete p;
    return 0;
}

```