C++面向对象程序设计模拟试题二

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. 说明内联函数的关键字是（　　）。
   A）inline　　B）virtual　　C）define　　D）static
2. 假定 CAb 为一个类，则执行 CAb oX；语句时将自动调用该类的（ ）
   A）有参构造函数　　B）无参构造函数
   C）拷贝构造函数　　D）赋值重载函数
3. cin 是某个类的标准对象的引用，该类是（　　）。
   A）ostream　　B）istream　　C）stdout　　D）stdin
4. 下面的哪个保留字不能作为函数的返回类型？（　　）
   A）void　　B）int　　C）template　　D）long
5. 派生类的成员函数不能访问基类的(　　)。
   A）保护成员　　B）公有成员
   C）私有成员　　D）前面各选项都正确
6. 在语句“cout << data;”中，cout 是(　　)。
   A）C++的关键字　　B）类名
   C）对象名　　D）函数名
7. 编译时多态性使用什么获得？（　　）
   A）重载函数　　B）继承　　C）虚函数　　D）B 和 C
8. 拷贝构造函数的参数通常是（　　）。
   A）无特殊要求　　B. 指向对象的指针
   C）本类对象的常引用　　D）对象
9. C++有几种联编？（　　）
   A）1 种　　B）2 种　　C）3 种　　D）4 种
10. 基类和派生类可以分别称为（　　）。
   A）“大类”和“小类”　　B）“父类”和“子类”
   C）“小类”和“大类”　　D）“子类”和“父类”

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. 设函数 max 是由函数模板实现的，并且 max(3.5, 5)和 max(3, 5)都是正确的函数调用，则此函数模板具有（　　）个类型参数。
2. 在 C++中，函数重载与虚函数帮助实现了类的（　　）性。
3. 由 static 修饰的数据成员为该类的所有对象（　　）。
4. 重载函数一般在参数类型或参数个数上不同，但（　　）相同。
5. 使用 new 建立的动态对象在不用时应该用（　　）释放所占用的空间。

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 Point
{
public:
	Point(int a = 0, int b = 0):x(a), y(b) {}
	int GetX() const { return x; }
	int GetY() const { return y; }
	void SetX(int a) { x = a; }
	void SetY(int a) { y = a; }

private:
	int x;
	int y;
};

int main()
{
	Point u;
	const Point v(6, 8);

	cout << u.GetX() << endl;
	u.SetX(16);
	cout << u.GetX() << endl;
	u.SetY(18);
	cout << u.GetY() << endl;

	cout << v.GetX() << endl;
	cout << v.GetY() << 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;

template <class Type>
class Test
{
public:
	Test(Type a[], int iSize):elem(a) { size = iSize; }
	void Print() const { for (int i = 0; i < size; i++) cout << elem[i] << "    "; }
```

```
private:
    Type *elem;
    int size;
};

int main()
{
    int a[] = {1, 0, 8};
    double b[] = {1.6, 1.8};

    Test<int> ar1(a, 3);
    ar1.Print();

    Test<double> ar2(b, sizeof(b) / sizeof(double));
    ar2.Print();

    cout << 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;

class Goods
{
public:
    Goods(int w): weight(w){ totalWeight = totalWeight + w; }
    Goods(const Goods &g)
    {
        weight = g.weight;
        totalWeight = totalWeight + weight;
    }
    ~Goods() { totalWeight = totalWeight - weight; }
    void Print() const;
    static int GetTotalWeight()    { return totalWeight; }

private:
    int weight;
    static int totalWeight;
```

```
};

int Goods::totalWeight = 0;

void Goods::Print() const
{
	cout << this->weight << "    " << this->totalWeight << "    ";
}

int main()
{
	Goods g1(6);
	g1.Print();

	Goods g2(g1);
	g2.Print();

	cout << Goods::GetTotalWeight();
	cout << endl;

	return 0;
}
```

上面程序的输出结果为：

4. 阅读下面程序，写出输出结果。

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

template <class Type>
class Test
{
public:
	Test(Type a = 0, Type b = 0, Type c = 0):z(c) {    x = a; y = b; }
	void Print()
	{
		cout << x << endl;
		cout << y << endl;
	}
	void Print() const
	{
		cout << z << endl;
	}
```

```
private:
    Type x, y;
    const Type z;
};

int main()
{
    Test<float> t1;
    t1.Print();

    Test<int> t2(1, 9, 6);
    t2.Print();

    const Test<double> t3(0, 6, 1.8);
    t3.Print();

    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;

template <class Type>
Type Max(const Type &a, const Type &b)
{
    if (a < b) return b;
    else return a;
}

template <class Type>
Type Min(const Type &a, const Type &b)
{
    if (a < b) return a;
    else return b;
}

int main()
{
    double x = 5.38, y = 6.09;
    cout << Max(x, y) << " " << Max<int>(x, y) << " ";
    cout << Min(x, y) << " " << Min<int>(x, y) << endl;
```

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

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

int main(void)
{
    B obj;

    return 0;
}
```
上面程序的输出结果为：

四、完成程序填题（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; }    // 将数据成员 n 初始化为形参 n
    void Show() const { cout << n << endl; }
};

int main()
{

    A i = 8;
    i.Show();

    return 0;
}
```

2. 将如下程序补充完整。

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

class A
{
private:
    int a;

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

class B: A
{
public:
    B(int m):____[2]____ {}                    // 将数据成员 a 初始化为 m
    void Show() const { A::Show(); }
};

int main()
{
    B obj(8);
    obj.Show();

    return 0;
}
```

3. 将如下程序补充完整。

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

class Test
{
private:
    static int num;

public:
    Test() { num++; }
    ~Test() { num--; }
    static void ShowObjectNum() { cout << num << endl; }
};

_____[3]_____          // 静态数据成员的初始化为 0

int main(void)
{
    Test::ShowObjectNum();
    Test obj;
    Test::ShowObjectNum();

    return 0;
}
```

4. 将如下程序补充完整。

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

class Integer
{
private:
    int i;

public:
    Integer(int x = 0): i(x) { }
    _____[4]_____{ return i; }// 类类型转换函数,将类 Integer 转换为基本类型 int
};

int main()
{
    Integer a, b(18) ;
    cout << a << endl;
    cout << int(b) << endl;
```

```
    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．编写一个函数模板，用于求数组中各元素之和，并编写测试程序进行测试。
函数模板声明如下：

```
template <class Type>
Type Sum(Type a[], int n);
```

2．定义一个抽象类 Shape，它有一个纯虚函数 GetPerimeter()；派生出四边型类 Rectangle 和圆类 Circle，在派生类中重载函数 GetPerimeter()，用于求图形的周长，编写测试程序进行测试。

C++面向对象程序设计模拟试题二参考答案

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．A)　2．B)　3．B)　4．C)　5．C)
6．C)　7．A)　8．C)　9．B)　10．B)

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：2
2．参考答案：多态
3．参考答案：共享
4．参考答案：函数名
5．参考答案：delete

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：
0
16
18
6
8
2．参考答案：1　0　8　1.6　1.8
3．参考答案：6　6　6　12　12
4．参考答案：
0
0
1
9
1.8
5．参考答案：
6.09 6 5.38 5
6．参考答案：
A
B
~B
~A

四、完成程序填题（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 A::n
2．参考答案：[2] A(m)
3．参考答案：[3] int Test::num = 0;

4．参考答案：[4] operator int()

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)
{
	Type tSum = 0;
	for (int i = 0; i < n; i++)
	{
		tSum = tSum + a[i];
	}
	return tSum;
}

int main()
{
	int a[] = {1, 2, 3};
	double b[] = {1.5, 2.8, 8.9, 8};

	cout << Sum(a, 3) << endl;
	cout << Sum(b, 4) << endl;

	return 0;
}
```

2．参考程序：

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

class Shape
{
public:
	virtual double GetPerimeter() const = 0;
};

class Rectangle:public Shape
{
public:
	Rectangle(double w, double h)
	{
		width = w;
```

```
        height = h;
    }

    double GetPerimeter() const
    {
        return 2 * (width + height);
    }

private:
    double width, height;
};

class Circle:public Shape
{
public:
    Circle(double r)
    {
        radius = r;
    }

    double GetPerimeter() const
    {
        return 3.1415926 * radius * radius;
    }

private:
    double radius;
};

int main()
{
    Rectangle oRectangle(2, 3);
    cout << oRectangle.GetPerimeter() << endl;

    Circle oCircle(10);
    cout << oCircle.GetPerimeter() << endl;

    return 0;
}
```