

《面向对象程序设计》朋辈课程 · 第十三辑

# 期末总复习

信息学院 赵家宇 丁舒佳

# 目录

01 选择题

02 程序分析（填空题）

03 简答题

04 设计题

## 选择题

1. 以下哪个选项不全是 C++ 中的关键字 ( )
- A. integer, float, double
  - B. auto, inline, switch
  - C. class, union, struct
  - D. virtual, static, namespace

考点：C++ 中一些基本概念（关键字） 第一、二章

2. 下面叙述不正确的是（ ）

- A. 派生类一般都用公有派生
- B. 对基类成员的访问必须是无二义性的
- C. 赋值兼容规则也适用于多重继承的组合
- D. 基类的公有成员在派生类中仍然是公有的

8. 关于友元，下列说法错误的是（ ）

- A. 如果类A是类B的友元，类B是类C的友元，那么类A也是类C的友元
- B. 如果函数fun()被说明为类A的友元，那么在fun()中可以访问类A的私有成员
- C. 友元关系不能被继承
- D. 友元是数据保护和数据访问效率之间的一种折中方案

13. 假设B是A的public继承的派生类，f是A类的public成员函数，g是B类新定义的public成员函数。则以下哪些语句是没有问题的  
( )

A a;

B b;

a. g(); // (1)

A \*p = &b; // (2)

b = a; // (3)

void func1(A \*p); func1(&b); // (4)

A. (1) (2)    B. (2) (3)    C. (2) (4)    D. (3) (4)

考点：继承（派生类&基类、友元） 第七章

3. 下列关于 new 运算符的描述中，哪个是错误的（ ）
- A. 它可以用来动态创建对象和对象数组
  - B. 使用 new 创建的 int 型数组 p[10]，可以用“delete []p;”来释放空间
  - C. 使用它创建对象时要调用构造函数
  - D. new创建的动态变量的空间是在栈区中分配的

**考点：变量的生存期（内存空间） 第四章**

4. 下面代码编译时不会报错的语句是（ ）

```
const int *p;
```

```
int *q;
```

```
const int x=0;
```

```
int y;
```

- |                     |                  |
|---------------------|------------------|
| (1) p = &x; *p = 1; | (2) q = &x;      |
| (3) p = &y;         | (4) p = &x; q=p; |

- A. (1)                  B. (2)                  C. (3)                  D. (4)

**考点：指针类型 第五章**

5. 下列有关重载函数的说法中正确的是 ( )

- A. 重载函数必须具有不同的返回值类型
- B. 重载函数参数个数必须相同
- C. 重载函数必须有不同的形参列表
- D. 重载函数名可以不同

15. 设有一个类为 A，现希望为 A类重载=运算符，并希望能实现如下形式的连续赋值 A a, b, c; a=b=c;且希望有较高的效率，则=运算符的最佳原型应为 ( )

- A. `A A::operator=(A a);`
- B. `A A::operator=(const A& );`
- C. `A& A::operator=(A a);`
- D. `A& A::operator=(const A& a);`

考点：函数名重载、操作符重载 第四、七章

6. 对于下图所示类 A、B、C和D，创建D类对象d时，所调用的构造函数及它们的执行顺序是（ ）

- A. D ( ) 、 B ( ) 、 C ( ) 、 A ( )
- B. D ( ) 、 C ( ) 、 B ( ) 、 A ( )
- C. A ( ) 、 C ( ) 、 B ( ) 、 D ( )
- D. A ( ) 、 B ( ) 、 C ( ) 、 D ( )

```
class A{
    int x;
public:
    A(int i){x=i;}
};
class B: virtual public A{
    int y;
public:
    B(int i): A(1) {y=i;}
};
class C: virtual public A{
    int z;
public:
    C(int i): A(2) {z=i;}
};
class D: public B, publicC {
    int m;
public:
    D(int i, int j, int k): C(j), B(i), A(3) {m=k;}
};
.....
D d(1,2,3)
```



7. 如果myclass类定义了拷贝构造函数和一个整型参数的构造函数，还重载了赋值运算符。那么语句 myclass obj=10; 会（ ）

- A. 调用拷贝构造函数
- B. 调用整型参数的构造函数
- C. 调用赋值运算符
- D. 引起编译错误

10. 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

考点：构造函数   第六章

9. 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

考点：this 指针 第六章

11. 有如下函数模板定义

```
template<class T>
T func( T x, T y)
{ return x*x+y*y; }
```

在下列 func的调用中不正确的是  
( )

- A. func ( 3.5, 5) ;
- B. func ( 3, 5);
- C. func<double> ( 3.5, 5);
- D. func< int > ( 3.5, 5);

考点：模板 第十章

12. 下列有关输入/输出（I/O）的说法中正确的是（ ）
- A. 在 C++中，输入/输出是语言定义的成分
  - B. 在 C++中，输入/输出操作不是一种基于字节流的操作
  - C. 对自定义的类重载插入操作符“<<”和抽取操作符“>>”时，不能作为类的成员函数来重载
  - D. 文件输入操作是指把计算机内存中的数据写入到外存中的文件里

考点：输入/输出 第八章

14. 有如下类定义

```
class MyBASE{
    int    k;
public:
    void set(int n){k=n;}
protected:
    int get() const {return k;}
};
class MyDERIVED: protected MyBASE{
    int j;
public:
    void set(int m, int n){MyBASE::set(m); j=n;}
    int get() const {return MyBASE::get()+j;}
};
```

则类 MyDERIVED 中保护成员个数是（ ）

A. 4 B. 3 C. 2 D. 1

考点：类 、继承 第六、七章

## 程序分析题（填空）

16. 填写程序，利用引用类型实现交换两个 `int *`型指针变量的值。  
（本题 4分）

```
#include <iostream>
using namespace std;
void swap(____ (1) _ __) //交换两个 int *型指针变量的值
{
    int *t;  ____ (2) ____
}
int main()
{
    int a=0,b=1;
    int *p=&a,*q=&b;
    cout << *p << ', ' << *q << endl; //输出: 0, 1
    swap(p,q);
    cout << *p << ', ' << *q << endl; //输出: 1, 0
    return 0;
}
```

考点：引用 第五章

17. 填写程序，实现二维数组的求和操作。（本题 4分）

```
int sum(____(1)____, int num) //2分
{
    int s=0;
    for (int i=0; i<num; i++) s += x[i];
    return s;
}

.....
int a[10][5], b[40][20];
.....
cout << sum(____(2)____, 10*5) ; //1分
cout << sum(____(3)____, 40*20); //1分
```

考点:二维数组、循环语句 第三、五章

18. 阅读程序回答问题。（本题 4分）

```
class A
{
    int x,y;
    char *p;
public:
    A(char *str)
    {
        x = 0;  y = 0;
        p = new char[strlen(str)+1];
        strcpy(p, str);
    }
    ~A() { delete [] p; p=NULL; }
};

.....
A a1( "abcd" );
A a2(a1);
```

- (1) 以上代码存在什么问题？（2分）
- (2) 如何解决？（2分）

考点:拷贝构造函数 第六章



19. 阅读下列程序，写出程序具体调用函数。（本题 6分）

```
#include <iostream>
using namespace std;
class A{
public:
    A() { f(); }
    virtual ~A();
    virtual void f();
    void g();
    void h() { f(); g(); }
};
class B: public A{
public:
    ~B();
    void f();
    void g();
};

void main() {
    B b; //调用 B::B(), A::A() 和 A::f
    A *p;
    p=&b;
    p->f(); //调用 B::f
    p->A::f(); //调用 A::f
    p->g(); //调用 A::g
    p->h(); //调用____(1)____
    p = new B; //调用____(2)____
    ...
    delete p; //调用____(3)____
}
```

考点:构造函数、继承、虚函数 第六、七章

20.填写程序，完成文件输入。（本题 4分）

```
#include __ (1) __
#include <iostream>
using namespace std;
struct Student
{
    int no;
    char name[10];
    int scores[5];
} s1;

void main() {
    //以二进制方式输入数据
    ifstream in_file("d:\\students.dat", __ (2) __);
    if(____ (3) ____ ) {
        cerr<<"Fail to open file"<<endl;
        exit(-1);
    }
    in_file.read(____ (4) ____);
    in_file.close();
}
```

考点:文件的输入输出 第八章

## 考点：栈、一维数组 第四、五章

21. 完成如下程序。（本题 4分）

```
#include <iostream>
using namespace std;
template <__ (1) __>
class Stack
{
    T buffer[size];
    int top;
public:
    Stack() { top = -1; }
    bool push(const T &x) {
        if (top == __ (2) __) {
            cout << "Stack is
            overflow.\n";
            return false;
        }
        else {
            top++; buffer[top] = x ;
            return true;
        }
    }
}
```

```
bool pop(T &x) {
    if (top == -1) {
        cout << "Stack is empty.\n";
        return false;
    }
    else {
        __ (3) __; top--;
        return true;
    }
};

}; int main()
{
    double x;
    Stack< __ (4) __> st1;// st1为元
    素个数为 100的 double型栈
    st1.push(10.0);
    st1.pop(x);
    .....
}
```

## 简答题

- 22. 在面向对象程序设计中，如何理解数据的抽象与封装。
- 23. 拷贝构造函数的作用是什么？何时会调用拷贝构造函数？
- 24. C++怎样实现消息的动态绑定，请简单说明下实现过程。
- 25. C++标准模板库（STL）中包含哪几类模板？它们的作用分别是什么？

重点在类、继承、多态、STL（我记得我们这一届有一题是说写四个STL容器）

建议把第六、七章多看看，一些重要概念可以背一下

# 设计题

一定要会

26. 编写类 String的构造函数、析构函数、赋值函数，以及测试的main函数。

已知类 String的原型为：

```
#include <iostream >
#include <string.h>
class String {
public:    String(const char *str=NULL); // 普通构造函数
        String(const String &other); // 拷贝构造函数
        ~String(); // 析构函数
        String & operator=(const String &other); // 赋值
        函数
        void show() {
            cout<<m_data<<endl;
        }
private: char *m_data; // 用于保存字符串
};
```

27. 定义一个抽象立体图形类(Geometry)，成员有立体图形名称(name)、求体积的抽象方法 `getVolume()`，将它作为**基类**派生出球体类、长方体类和圆柱体类，实现这些**派生类**的方法 `getVolume()`，并在 `main`函数中求它们的体积之和。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。（本题9分）

**理清基类和派生类之间的逻辑关系！  
注意题目要求实现哪些方法**

28. 编写一个学生成绩输入/输出程序。该程序从键盘输入学生基本信息及各门课的成绩，然后把它们保存在文件中。内容包括：学号、姓名、选课门数以及各门课的成绩。要求通过**重载操作符**”>>”和”<<”来实现学生信息的输入/输出。（本题 10分）

学生成绩类的定义如下：

```
const int MAX_NUM_OF_COURSES=30;
const int MAX_ID_LEN=10;
const int MAX_NAME_LEN=8;
class StudentScores {
public:    StudentScores() { initialized = false; }
        bool data_is_ok() const { return initialized; }
private: int scores[MAX_NUM_OF_COURSES], num_of_courses;
        char id[MAX_ID_LEN+1], name[MAX_NAME_LEN+1];
        bool initialized;
        friend istream &operator >>(istream &in, StudentScores &x);
        friend ostream &operator <<(ostream &out, StudentScores &x);
};
```

大概率会考  
去年也考了

# 总结

- 重点在六、七章（类和继承）
- 有很多概念性的东西一定要搞懂，经验是多看书多总结



**祝大家都能取得  
满意的成绩！**