

C++ 高级程序设计 2023 年秋（智软）

作业三

2023 年 10 月 29 日

- 题目 1. 什么是操作符重载？操作符重载有哪些基本原则？
- 题目 2. 简述操作符重载的两种实现途径，这两种实现途径有哪些异同？
- 题目 3. 简述拷贝构造函数与赋值操作符“=”重载函数的区别。
- 题目 4. 请从实现和使用的两个方面，比较前置和后置的 operator++ 的不同。
- 题目 5. 给定下列代码，回答问题。

```
class A {
    int n, m;
public:
    A():n(0), m(0) {}
    A(int n, int m):n(n), m(m) {}
    A(const A&a):n(a.n), m(a.m) {}
    ~A() {}
};

class B: public A {
    int x;
public:
    B():x(0) {}
    B(int x):x(x) {}
    B(const B&b) {}
    ~B() {}
};

class C {
public:
    C() {}
    ~C() {}
};

class D: public C {
    B b;
public:
    D() {}
    ~D() {}
};
```

```
int main() {
    D d1;
    D d2(d1);
}
```

- 问题 1: 执行代码 D d1 时, 给出构造过程的执行流 (包含 call 和 return)。执行流示例见图 1。

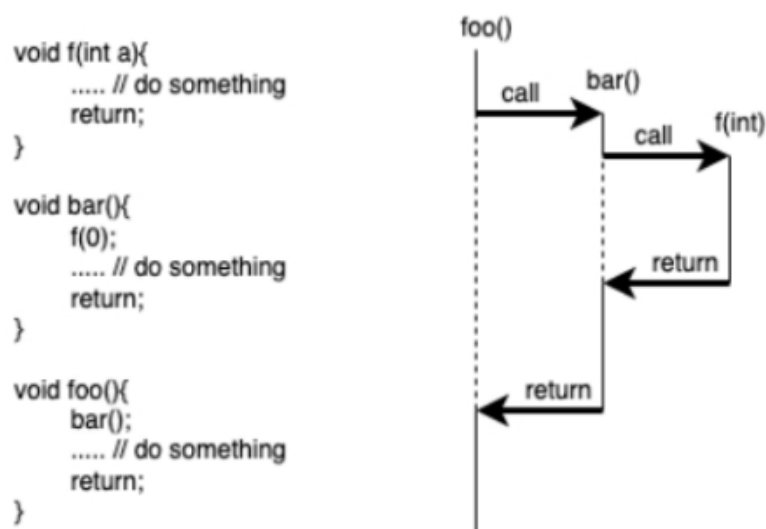


图 1: `foo()` 函数执行时的执行流示例

- 问题 2: 执行代码 D d2(d1) 时, 给出构造过程的执行流 (包含 call 和 return)。
- 问题 3: 给出 d1 析构时, 析构过程的执行流 (包含 call 和 return)。

注意: 说明调用的构造函数时, 需要区分各种重载形式。例如用 `A::A()`, `A::A(int, int)`, `A::A(const A&)` 区分 A 类的默认构造函数及其重载。