

Here is the definition of a simple class. The bodies of all methods have been replaced with a print statement so we can trace the order in which they are called.

```
class Test {
public:
    // constructor
    Test() { cout << " cons "; ... }

    // copy constructor
    Test(const Test &other) { cout << " cc "; ... }

    // assignment
    Test & operator=(const Test &other) { cout << " op= "; ... }

    // destructor
    ~Test() { cout << " byebye "; ... }
};
```

Now consider the following program that uses class Test:

```
// = copy of obj
Test dup(Test obj) {
    return obj;
}

// main program
int main() {
    Test it;
    Test that = it;
    it = dup(that);

    return 0;
}
```

What happens when this program runs? Show the messages that get printed, and, to help both you and your TA understand your answer, identify which object is associated with the message (i.e., cons that; op= assignment to that, etc.)