1. What is the difference between a shallow copy and a deep copy?
2. What is the rule of 3?
3. The copy constructor for a class X has the form X(const X& b). Why is the parameter passed by reference? Explain why you can’t define a constructor of the form X(X b)?
4. What would be the implication of declaring a copy constructor private?
5. What is the difference between destruction and deletion of an object?
6. What problems could a programmer encounter if they defined a destructor for a class but no assignment operator? Illustrate your description with an example class.
7. What problems could a programmer encounter if they defined a destructor for a class but no copy constructor? Illustrate your description with an example class.
8. Which objects are destroyed when the following function exits? Which values are deleted?

void f(const Fraction& a)

{

Fraction b = a;

Fraction\* c = new Fraction(3, 4);

Fraction\* d = &a;

Fraction\* e = new Fraction(7,8);

Fraction\* f = c;

delete f;

}

1. What error is being committed in the assignment operator for the following class?

class String

{

public:

String(const char right[]);

String& operator=(const String& right);

private:

char\* buffer;

};

String::String(const char right[])

{

len = 0;

while (right[len] != '\0')

len++;

buffer = new char[len + 1];

for (int i = 0; i < len; i++)

buffer[i] = right[i];

buffer[len] = '\0';

}

String& String::operator=(const String& right)

{

int n = right.length();

for (int i = 0; i <= n; i++)

buffer[i] = right.buffer[i];

return \*this;

}

Use the following class for the following problem. The only purpose of the class is to display a message both when the constructor is invoked and when the destructor is executed.

class Trace

{

public:

Trace(string n);

~Trace();

private:

string name;

};

Trace::Trace(string n) : name(n)

{

cout << "Entering " << name << "\n";

}

Trace::~Trace()

{

cout << "Exiting " << name << "\n";

}

1. Extend the class Trace with a copy constructor and an assignment operator, printing a short message in each. Use this class to demonstrate
   1. the difference between initialization

Trace t("abc");

Trace u = t;

and assignment.

Trace t("abc");

Trace u("xyz");

u = t;

* 1. the fact that all constructed objects are automatically destroyed.
  2. the fact that the copy constructor is invoked if an object is passed by value to a

function.

* 1. the fact that the copy constructor is not invoked when a parameter is passed

by reference.

* 1. the fact that the copy constructor is used to copy a return value to the caller.