

CS250 Midterm 3

Date: Friday, April 22, 2005

Name: _____

Total: 90 points

1. (16 points) Select **one** correct answer for each of the following and write a small sentence describing why you made that choice:

1. A derived class will inherit data members from
 - a. only its most immediate ancestor
 - b. all of its ancestors
 - c. none of its ancestors

2. Which of the following statements are true about a variable created with the static modifier?

- a. Once assigned the value of a static variable may not be altered
- b. A static variable created in a function will keep the same value between calls
- c. Only one instance of a static variable will exist for any amount of class instances
- d. The static modifier can only be applied to a primitive value (simple data type)

3. When executing the constructor of a derived class, when is the constructor of the base class invoked?

- a. At the beginning of the derived constructor
- b. At the end of the derived constructor
- c. At the end of the derived constructor's prototype
- d. When it is explicitly called in the derived constructor
- e. When an object goes out of scope

4. Why is the second parameter of the following function passed as a constant reference rather than passed by value?

```
Ostream& operator<<(ostream& out, const addressType &address)
```

- a. Because the function will change its value
- b. Because operator overload functions cannot take a by-value argument
- c. So that the entire class will not be passed on the stack, thus saving memory space and execution time, but at the same time preventing the function from changing the value of the argument
- d. Because objects cannot be passed by-value
- e. Because all operands to a binary function must be constant references

2. (30 points) A class `Date` contains the information about dates. The class stores a day, month, and year as integers.

a. Write the class definition of `Date`. You should include all the necessary data members and a constructor.

b. Write the definition for the overloaded `<<` operator. This should output the date in the form `mm/dd/yyyy`. Make sure you add the function prototype to the class definition you created in part a.

c. Describe how you added the `<<` operator to the class (as a friend, member function) and why you made the choice you did.

d. Write the definition for the overloaded operator `==` that will compare two dates and return true if they are the same and false otherwise. Make sure you add the function prototype to the class definition you created in part a.

e. Write a main program that you can use to test your class. This main should create three date objects. These objects should be set to the dates 3/4/2005, 6/30/2005 and 12/31/2005. Compare two of the dates, and output one of the dates.

3. (24 points) Using the three classes defined next, answer parts a – c.

```
class Automobile
{
public:
    Automobile();
    virtual string getName() const = 0;
    virtual string getType() const;
    string getColor() const;
    ...
};

class Car : public Automobile
{
public:
    Car();
    virtual string getName() const;
    virtual string getType() const;
    virtual string getColor() const;
    virtual void driveMe();
    ...
};

class Sedan : public Car
{
public:
    virtual string getType() const;
    virtual void driveMe();
    ...
};
```

- a. What is the problem with the following statement?
Automobile anAuto;

b. Are there any problems with the code snippet below? Why or why not?

```
void drive( Car someCar )
{
    someCar.driveMe();
}
```

```
int main( )
{
    Sedan sedan;
    drive( sedan );

    return 0;
}
```

c. For the statements listed below (1-4), indicate the class from which the function will be called (i.e. Sedan, Car, or Automobile)

```
Sedan sedan;
Car car;
Automobile *p1( &sedan );
Automobile *p2( &car );
std::cout << sedan.getColor() << endl; // (1)
std::cout << p1->getType() << endl; // (2)
std::cout << p2->getColor() << endl; // (3)
std::cout << car.getColor() << endl; // (4)
```

4. (20 points) Templates.

a. Write a simple function template for a function `isEqualTo` that compares its two arguments of the same data type with the equality operator `==` and returns `true` if they are equal and `false` if they are not.

b. Call the function you just wrote to compare two integers (`45, 23`) and two characters (`'s', 's'`).

5. (7 points extra credit) A dangling pointer is a pointer that is pointing to memory that has been freed for use. Dereferencing this pointer will cause an error since the object being pointed to is no longer valid.

a. Write a C++ code snippet that illustrates dangling pointers using objects.

b. Describe your code and explain why the pointer is a dangling pointer.