1. What is the difference between a class and an object?

In the class the user defines the instructions of how a particular data type (object) will behave. For example, a class will hold private and public member functions and data members, inheritance, and constructors. The object is an instance from that class: the program will use the rules defined in the class to create and manipulate such data type (object).

1. In general, why is it a bad idea to make data members public?

If data members are made public, they can be modified by the program, which can results in errors.

1. What are the similarities and differences of structures and classes? When do you use one versus the other in C++?

Structures and class both have private and public data members, and you can define member functions. A class can have more complexity that a struct (i.e. inheritance). Therefore is good to use a struct when defining something that does not requires too much complexity i.e. a three dimensional point (x, y, z). If defining something with more complexity, use a class.

1. What are the advantages and disadvantages of using defaultable parameters?

Advantages are only defining one constructor for a class, not needing to know all parameters of a function to use it.

Disadvantages are: using a function without knowing all its parameters which can led to errors in the program; the compiler can throw an “is ambiguous” error if when defining a function g’ that is overloading a function g, and the function g can be called in the same way that g’ can be called. For example if we have int g(int x, int y = 2) and try to overload it with int g(int x)

1. Why might it be hazardous to overload comparison operators?

The interpretation of a comparison operator is well establish (i.e. == for equal, != for not equal, etc.). When overloading a comparison operator, the user of the code must know how the overload for that particular comparison operator was defined. If not, this can lead to errors