This homework includes short answer questions and a programming assignment. The short answer questions are worth 5 points each. The programming assignment is worth 30 points. The short answer questions may cover topics that are in the text, but were not covered in the lectures. Push a preliminary version of the programming assignment to Github before Wednesday's class. Push the final version of your source code, Makefile, and a text file with your short answer questions to Github before 5PM Friday.

1. What does the friend keyword do?

The friend keyword lets a class access to functions outside of it. It is able to access the private data and methods found in those external functions.

2. Why should you never use using namespace std;?

If you do this, it will import the namespace in any C++ files that also share the same header. Using this statement goes against the usefulness of namespace because you now can not use namespace to prevent collisions.

3. Why is memory management in C++ hard?

C++ introduced objects to the programming world, but without the concept of ownership. Objects become a problem when you pass a pointer to a method and the method store's a copy of that pointer in the methods object. This allows any of the other object's methods to access the pointer. C++ does not delete memory after the reference goes out of scope, but instead leaves that to the programmer to delete.

4. Describe the concept of "ownership", as it pertains to memory management in C++.

The programmer themselves has to document the ownership of something such as a pointer, instead of having that feature just built in. It is imperative to be careful and keep track of what method owns which things.

5. What are the three ways a class can contain objects in C++?

The three ways a class can contain an object are as an actual object, a reference, and a pointer.

6. What is a forward declaration and why would you use one?

A forward declaration is telling the compiler that there is a header file that it is going to run into, but not yet. Since the compiler only makes one pass through the code, it will be confused if it sees a header file included that it hasn't actually seen. The forward declaration allows you to tell

the compiler that although they haven’t seen the header file yet, it is included in the code that follows. A forward declaration would be used to include a header file in another header file.