**Take-Home: Quiz 3 (15 pts) – More OOP**

1. **(4 pts)** What is a *constructor*? Explain.

A constructor is a function that initializes an object when it is instantiated, whether from a function call (e.g. copying into a local scope via pass-by-value) or explicitly instantiated in code. A constructor can initialize an object with passed-in values, default values, or simply the current values from another pre-existing object.

1. **(4 pts)** What is a *destructor*? Explain.

A destructor is a function that is called right before an object is destroyed, either by going out of scope or explicitly using the *delete* keyword. The destructor is responsible for gracefully cleaning up the object and its member attributes (and calling the destructors for any objects contained within the parent object) so that memory issues such as leaks, dangling pointers, etc. do not occur.

1. **(4 pts)** What is a *reference*? Explain.

A reference is an "alias" or another name for a variable/object in a local scope different from the calling scope, which refers back (via a ref table) to the actual variable/object in the calling scope. Put it another way, a reference makes it so that a function effectively uses the original object in its operations, instead of making a copy of the original object in its own stack frame. This means that with a reference, changes to the object in the function will persist!

1. **(3 pts)** What is a class? Explain.

A class is the definition of a collection of variables (attributes) and functions (methods) that are all encapsulated or packaged into a standardized container. Classes enforce information hiding, i.e. the idea that whoever is using the class does not need to know about the process by which the methods operate, only that they do exactly what they are described to do (well-defined behavior). Classes protect their attributes by ensuring that the user cannot alter the data contained within except through pre-approved *interfaces* coded in the class.