**Take-Home: Quiz 7 (15 pts) – Inheritance and Polymorphism in C++ Again!**

1. **(8 pts - 2 pts/pillar)** What are the four pillars of object-oriented programming (OOP)? As part of your solution define each one.

Abstraction - implementation details of how a task is done do not matter as long as methods have predictable behavior and outputs and objects communicate to each other through clear, well-defined interfaces.

Encapsulation - "bundling" of data members and methods acting on those members with well-defined behavior in a single package, i.e. the implementation of a well-designed class.

Inheritance - a class can be derived from a "parent", that is, retain the necessary members from its parent while extending or modifying its parents behavior, for extensible code.

Polymorphism - derived class methods can "override" the base class's behavior, allowing for more specialized behavior in the derived class.

1. **(4 pts)** Why do we use *virtual* functions? Explain.

When we want a base class function to be overridden, e.g. if we want a function to do the same general task but with more specific behavior in a derived class and thus want to call the specific method for the derived class instead of the base class method.

1. **(3 pts)** Explain the purpose of a *virtual* *destructor*.

Ensures that objects in inherited classes will garbage collect correctly; makes it so that the compiler will always attempt to look for an overridden destructor in a derived class first, before \*also\* calling the base class destructor.