Aamir Khan

Prof. Paul Conrad

CSC-18A

12 July 2022

Assignment 8 - Inheritance Report:

Inheritance promotes software reusability because it provides us with a way to create related classes that extend other classes with ease. Through inheritance, a subclass can inherit all the members of a superclass, apart from its private members. This makes it very easy to extend and add onto classes with a simple “extends” keyword. We essentially only need to define a base class that other classes will be related to, and we can create subclasses that will reuse the members and fields of our base class. After we inherit from another class, we may add onto the derived class to make it unique and have its own methods besides those that were inherited. Since we don’t need to code an entirely new class with the same methods and attributes as another class, we save time programming. For example, say we wanted to create several classes that had most of the same members, we could simply use inheritance to quickly do this as opposed to writing the same code for each individual class. Moreover, if we define our superclass properly, we can keep our data protected to reduce the number of errors in our code. By using protected members, we can have members that are basically treated as private within the class but can be inherited in derived classes. This is useful if we want to have access to certain members in subclasses. This can also be considered a disadvantage because it allows us to have access to the parent class’s members from subclasses. This isn’t the best practice because we generally want to hide the internal state of our classes. This doesn’t promote the idea of encapsulation to its full extent. On the other hand, using private members is always a good way to approach the design of your classes because it allows you to hide the internal state of your class and protect your data from unnecessary access. However, this prevents subclasses from inheriting these members, so they only reside within the base class. This can be considered a disadvantage in some cases because you may want to have access to your base class’s members.