Aamir Khan

Prof. Paul Conrad

CSC-18A

12 July 2022

Assignment 9 – Abstract Classes & Interfaces Report:

An abstract class is a special type of class that is declared but can’t be instantiated directly. It serves as a model or template for inherited classes to specify what these classes can do. In this way, it promotes software reusability because many classes can be derived from the same abstract class. However, using abstract classes gives us a way to make each subclass implement the same methods differently. When using abstract classes, abstract methods can be declared. These are methods that have no body and must be overridden in our subclasses. The use of abstract classes is useful if you need to have your subclasses override its members and implement its methods differently. It also saves us time because we can have subclasses inherit the traits and methods of these classes as opposed to copy-pasting all the code. Since we can inherit these members, we can avoid any typing errors given that we don’t need to retype our code. We also can have our data protected to ensure data encapsulation, so this also prevents us from unnecessarily handling our data. Having data encapsulation in place keeps our data protected and the internal state of our class hidden. Interfaces on the other hand promote multiple inheritance. Just like abstract classes, interfaces are like templates that specify the actions that a class can do but not how it must do them. This allows us to override the same methods in different subclasses to perform different implementations of the same method. Moreover, we can implement multiple interfaces, but we can only directly inherit from one superclass. Having interfaces is very useful if we need to create a class that needs to inherit from multiple classes. It’s very powerful in this sense because it allows us to create classes that implement different class templates. We can essentially inherit the methods and attributes of an interface and make it our own with overridden implementations that then specify how we perform actions.