Lab 4: Chapter 7

Answer the following questions: 1, 3, 4, 7, 8, 10, 11

1. Does a superclass have access to the members of a subclass? Does a subclass have access to the members of a superclass?

A subclass has access to the nonprivate members of its superclass, but not the other way around; superclasses cannot access any part of their subclass(es).

3. How do you prevent a subclass from having access to a member of a superclass?

By declaring a superclass’ member as private, a subclass cannot access it.

4. Describe the purpose and use of both versions of super:

The two general forms of super are used as follows; the first is used to call a superclass constructor from within a subclass, while the second is used to access a member of a subclass’ super that has been protected/hidden.

7. What is an abstract class?

An abstract class is one that is never instantiated itself, but used to serve as a ‘base class framework’ for subclasses to inherit and then assign values to on an individual basis.

8. How do you prevent a method from being over-ridden? How do you prevent a class from being inherited?

Methods can be ‘protected’ from being over-ridden by using a *final* modifier in their definition (once declared final, they’re final and unchangeable). If a method is declared private in a class and a subclass is derived from it, it won’t have access to that class, which will thereby remove the opportunity to be changed. A method declared as *static* will not be open to over-riding, because it is resolved in during compiling, unlike overriding, which is resolved during runtime.

10. What class is a superclass of every other class?

The *Object* class is the superclass of all others – all classes (eventually) inherit from it.

11. A class that contains at least one abstract method must itself be declared abstract. T or F?

It is true that a class containing even one abstract method must be declared abstract (there are no objects of an abstract class, since an a.c. is never fully instantiated).