Final Review

Here are the main topics we have seen in class. I expect you to be able to answer basic questions about each, including definitions, relationship between various notions. I also expect you to be able to recognize any of the notions below, and to use them in code fragments that I may ask you to write.

- ADT and specification: signatures, algebraic specifications, reasoning with algebraic specifications, implementing specifications, public versus private members of a class, static methods for creators, dynamic methods for operations (understand the difference), code against an ADT to allow substituting implementations, defining equality for ADTs, implicit specifications for equality.

- Errors and testing: classification of errors, exceptions, use of the type system to guarantee some classes of errors do not occur, exceptions, black-box versus white-box testing, unit testing versus integration testing, tests generation, testing via a specification.

- Subclassing: definition of subclass, abstract classes, concrete classes, design pattern to derive an implementation from an ADT specification, nested classes, tree subclassing hierarchies, non-tree subclassing hierarchies, interfaces, casting, dynamic dispatch.

- Polymorphism (aka generics): polymorphic interfaces, polymorphic classes, polymorphic methods.

- Inheritance: definition of inheritance, differences with subclassing, inheritance and subclassing in Java, multiple inheritance, inheritance versus delegation, protected members of a class.

- Design patterns: Iterator design pattern, distinction between functional iterators and Java iterators, Adapter design pattern, adapters for functional iterators and Java iterators, Map and Reduce design patterns, Publish-Subscribe design pattern, laziness, MVC design pattern.

- Mutation: memory model of Java, static fields, sharing, shallow copy, deep copy.