SE2811 Final Exam Study Guide

Describe the broad Categories of patterns:

(p. 589)

Behavioral

Structural

Creational

Know these Idioms:

Simple Factory Idiom

Lazy Initialization Idiom

Know these patterns:

Singleton

Composite

Decorator

Strategy

Command

Observer

Compare these patterns:

(The above)

Façade

Adapter/Wrapper

Proxy

Iterator

Factory Method (Knowing simple factory is enough)

Abstract Factory

Know this Composite Pattern:

MVC – Model-View-Controller

What it means to “know” a pattern/idiom/etc.

Compare the Patterns

Implement the Pattern

Compare Patterns

Identify situations where a pattern can be helpful

Select a pattern for a particular problem

Describe similarities and differences between patterns

Categorize patterns (roughly)

Implement the patterns

Draw UML class diagrams

With pattern names/methods

With names/methods for a given application

Write standard methods used by the design pattern (e.g. Singleton.getInstance)

Write code making use of design pattern classes (e.g. main)

Write code to apply a pattern to a specific problem (e.g. WeatherObserver.update)

Refactor (rewrite) code to benefit from the pattern (the 20/100 problem e.g. Ctr. to MVC)

Be able to apply multiple patterns to a single problem (e.g. as in Shape lab)

Threads

Write a double-checked lock.

Know Java’s locking syntax, how to ensure two methods are locking on the same object

Know what a synchronized method locks on (“Excellent” credit)

Know how to use wait & notify without the waiter hanging forever.

Things I hope you’ll learn as you go into industry

Modify existing patterns to make your own pattern for a new problem

Know when to NOT make your own pattern (See pp. 586-587)