2511 Final Exam Preparation

OO basics:

- Abstraction
- Encapsulation
- Inheritance
- Polymorphism

OO Design Principles

- Principle of least knowledge
- Encapsulate what varies
- Favour composition over inheritance
- Program to an interface not an implementation
- Classes should be open for extension and closed for modification
- Don’t call us, we’ll call you
- A class should only have one reason to change
- Strive for loosely coupled designs between objects that interact
- Depend on abstractions, do not depend on abstract classes

Design Patterns

- Structural Patterns
  - Composite
  - Decorator
- Behaviour Patterns
  - Strategy
  - State
  - Template Method
  - Iterator
  - Observer
- Creational Patterns
  - Factory method
  - Abstract method
  - Builder
  - Singleton
  - Command
  - Visitor

Plan:

- 3 quizzes
- Every tutorial
- Every lab
- Every lecture note
- Emmet’s repository
- Design code smells
    - https://sourcemaking.com/refactoring/smells

Polymorphism

- When a variable has the ability to take on many forms
- Usually occurs when a parent class can take on the form of different sub-classes

Inheritance

- An IS-A relationship where another class contains all the attributes and methods of another class, with also some additional attributes and methods

Abstract Class

- A class that can’t be instantiated