Exam 2 Review

- In general know all bold terms in each section
- Understand all Participation Activities and Challenge Activities.
- Understand any programming you did in Homework, Lab Work, and Challenge Activities. Be able to replicate it.
- The exam is made up of short answer, hand execution and analysis of code, writing code, and design.

Recursion (Chapter 7)
- Rules for creating a recursive function
- Definition
- Pros / cons

Objects and Classes (Chapter 8)
- Syntax
- Structure
  - Data members / attributes
  - Member functions / methods
  - Private helper functions
  - Constructor
    - Member initialization
    - Overloading
    - Default constructor
- Access
  - Public / private
- class vs. struct
- UML
  - Aggregation
  - Relationship
- Class vs. object
- Operator overloading
- ‘this’
- Namespaces
- Separate files (organization)

Exceptions (Chapter 9)
- Pros / Cons
- throw
- try
- catch (multiple handlers also)
- Path of exceptions through call stack

Inheritance (Chapter 10)
- Pros / Cons
- Class hierarchies
- Base class / superclass / parent class
- Derived class / subclass / child class
- Access
  - Public / private / protected
  - Data members /
    Member functions /
    Constructors
- const Member functions
- Overriding
- Polymorphism
  - Abstract Classes
  - Virtual Functions
  - Pure Virtual Functions
- UML
  - Inheritance
- Is-a vs. has-a relationships
Calculator
- Analyze / Design / Implement cycle
- Grammar
- Parsing
  - Parse Trees
  - Recursive Descent Parser
- Recursion: Direct vs. Indirect (Mutual)
- Purpose of Tokens
- Correlation of functions to grammar

Pointers Intro (Chapter 12.1-12.3)
- Performance issues: pointers vs. vectors / arrays
- Pointer operators
  - * (both flavors)
  - &
  - new
  - delete
  - ->

Graphics and FLTK
- Graphics coordinate system used by FLTK
- Shape inheritance
  - Inheritance
  - Virtual
  - Use of private / protected / public
- This section might be updated at the end of the week.