Lecture 25

Final Exam Review!
General Info

• Wednesday, December 21st, 2016 8:00 AM - 9:50 AM

• CIWW 102 (Our class)

• I’ll be having office hours as usual next week if you have questions, or you can email

• **IF there is any reason why you won’t be there for the final, you need to come talk to me or email me TODAY. You’d have to take it earlier and we need to get it approved / worked out.**
Format

• Format is similar to the midterms - there will be short answer:
  • T/F
  • Written short answer
  • Explain the output
  • Find the errors

• And programming questions:
  • Writing complete classes
  • Test classes using given classes
  • methods or parts of methods

• There will be extra credit as well
What’s on the test?

• EVERYTHING

• (pretty much)
Basic Stuff from Midterm I

- Primitive data types
- methods
  - void and ones that have a return type
  - passing by value, vs passing by reference
- if statements
- switch statements (Just kidding. No switch statements)
- min, max abs
- random number usage (make sure you know how to use this correctly! Lots of off-by-one errors last exam)
- Reading from the console using Scanner
- relational and logical operators (<=, && || !)
Basic Stuff from Midterm I

• comparing, searching in strings

• casting and parsing numbers and strings (back and forth)

• For, Do, Do…While loops (and break and continue keywords, nested loops)

• Arrays
  • create, copy, pass to methods, using java.util.Arrays methods
  • 2 and multi-dimensional arrays
  • looping

• scope of variables
Processing

• standard methods

• drawing shapes, changing colors

• keyboard and mouse interaction

• I won’t make you memorize the syntax this time
Objects

- how to set up a class
- using classes from other classes
- static data fields and methods(!!! Section 9.7 in the book)
- constructors
  - multiple constructors, private, protected
- instance variables and default values (null for objects, 0 for primitive data types, etc)
- primitive vs reference types (especially when passing to methods)
Static modifier

- Static methods and variables belong to the class, not to the instance

- They are the common property shared by all objects of a class, and are not unique to any one instance.

- Static methods can only access static data

- Static methods can only call other static methods

- Static methods cannot use `this()` or `super()`
Static modifier

- Examples of when you might use static variables or methods:
  - Keeping a count of all objects created by a class (static int totalCount)
  - creating a helper class with useful methods that you pass data to (Math class - Math.min(3,2))
  - “Does it make sense to call this method or access this variable, even if no objects have been created from it yet?”
• visibility modifiers (public, private, protected)

• data field encapsulation (getters and setters for private data, why we do this)

• Arrays of objects

• hidden variables and the this keyword

• class relationships
Objects

- Inheritance
- Polymorphism
- Exceptions
- use of keyword super
- Check out section 13.10 in the book
Other stuff

• Abstract classes!

• File I/O (using File, Scanner, and PrintWriter classes)

• ArrayLists

• Overriding vs. Overloading

• Random vs random
Interfaces

- Comparable interface
- how to create and use your own interface
- single inheritance, multiple-interfaces
Thank you!