Classes and Objects

- Class, Object
- Abstraction
- State vs. behavior
- Instance variables and methods
- Constructors (multiple)
- Encapsulation
- toString()
- this
- overloading
- object equality

Inheritance

- Code reuse
- Extends
- Superclass vs subclass
- Only extend a single class
- Protected
- Overriding methods
- Constructors not inherited
- Subclass constructors
- Everything extends Object
- Casting
- Instanceof
- Binding
- Polymorphism
- Abstract class

ArrayList

- Store any type of object
- Specify type
- Iterating
- Primitive vs. Wrapper

Recursion

- Pre/post conditions
- Base case vs. recursive case
- Call stack
- Activation record
- Call tree
- Infinite recursion
- Memoization

Interfaces

- Definition
- Interfaces as a type
- Polymorphism
- Relationship to Abstract Classes
- Inheritance
- Comparable
- instanceof

Counting

- Product Rule
- Sum Rule
- Inclusion / Exclusion
- Pigeonhole

Permutations / Combinations

- Calculate permutations, r-permutations
- Calculate combinations, r-combinations
- Use with Counting principles
  - Product, Sum
  - Inclusion/Exclustion, Pigeonhole
- Recognize scenarios

Induction

- Be able to perform mathematical induction proofs
  - Equality
  - Inequality

Programs

- Binary Search
- IFoo
- Spock's Dilemma

Linked Lists

- Java References
- Node class
- Using recursion to traverse a list
- Get / Add / Remove
- Clear / Size / isEmpty
- Doubly linked lists
- DLL w/head node

Iterators

- Why?
- Syntax / forms
- Interface
- Iterator vs. Iterable
- Inner classes

Sorting

- What are N^2 Algorithms?
- Bubble
- Selection
- Insertion
- MergeSort

Exceptions

- Definition
- Try/Catch blocks
- Predefined Exceptions
- Declaring Exceptions
- Checked vs. unchecked
- Inheritance of Exceptions
- Multiple catch blocks
- Finally