Review sheet for Exam 3 and Lab Exam

The following is a list of topics covered since Exam 2. These topics come from the reading, from the slides and from labs and code assignments. Exam 3 is largely conceptual in nature, while the lab exam (given in your last lab), will be largely implementation-based. Please note that you should be prepared to write small pieces of code for Exam 3.

Recursion

- necessary conditions for recursion to work
  - terminating condition
  - forward progress
- understand and write
  - factorial function
  - recursive functions for trees, linked-lists and stacks, as shown below

Linked-Lists

Methods
You should be able to write the following methods and understand how they work:
- all from lab, but particularly:
  - add (2 versions)
  - clear
  - clone (deep)
  - indexOf
  - isEmpty
  - hashCode
  - iterator()

Constructors
- 3 versions

Stack (from linked-lists)

Methods
Given an interface for linked lists, you should be able to write the following methods and understand how they work:
- Push
- Pop
- Top
• isEmpty
• height

Binary trees

Topics
• terminology
  o node
  o root
  o degree
  o parent
  o children
  o simple path
  o siblings
  o height
  o level
• traversals – in-order, pre-order, post-order
  o understand how to implement
  o given tree, give output of these traversals
• building trees –
  o give definition of binary tree node constructor
  o given a picture of a tree and a node constructor, be able to write an assignment statement building an equivalent tree
• binary search trees
  o finding, addition, removal
    • given a tree and a data element, show tree after addition or removal is called
  o implement these methods using recursion

Software Development Methods
• Risks
  o Name and discuss at least 5 risks faced during software development
  o Name and discuss at least 5 ways to reduce risk
• Terms
  o User stories
  o Tasks
  o Estimation (why)
  o Priorization (why and who)
  o Unit testing
  o Acceptance testing
  o Spikes (prototypes)
  o iteration