The Final exam is Tuesday May 1st from 8:00 - 9:50 AM in Wright 9. It will be closed-book and notes, except for **three** 8" x 11" sheets of paper containing any notes that you want. (Plus, the Python Summary Handout) About 75% of the test will cover the following topics (and maybe more) since the second mid-term test, and the remaining 25% will be comprehensive (mostly big-oh analysis and general questions about stacks, queues, priority queues/heaps, lists, and recursion).

Chapter 6: Trees

Terminology: node, edge, root, child, parent, siblings, leaf, interior node, branch, descendant, ancestor, path, path length, depth/level, height, subtree
General and binary tree recursive definitions
Tree shapes and their heights: full binary tree, balanced binary tree, complete binary tree
Applications: parse tree, heaps, binary search trees, expression trees
Traversals: inorder, preorder, postorder
Binary search tree ADT: interface, implementation, big-oh of operations
Balanced binary search trees: AVL tree ADT: interface, implementation, big-oh of operations

File Structures - Lecture 24 handout:

http://www.cs.uni.edu/~fienup/cs1520s18/lectures/lec24_questions.pdf

We talked about how the in memory data structures need to be adapted for slow disks.
From this discussion you should understand the general concepts of Magnetic disks:

- layout (surfaces, tracks/cylinders, sectors, R/W heads)
- access time components (seek time - moving the R/W heads over the correct track, rotational delay - disk spins to R/W head, data transfer time - reading/writing of sector as it spins under the R/W head)

Hash Table as a useful file structure
<u>B+ trees as a useful file structure - see web resources:</u>

http://www.sci.unich.it/~acciaro/bpiutrees.pdf

http://en.wikipedia.org/wiki/B%2B_tree

http://www.ceng.metu.edu.tr/~karagoz/ceng302/302-B+tree-ind-hash.pdf

Chapter 7: Graphs

Terminology: vertex/vertices, edge, path, cycle, directed graph, undirected graph
Graph implementations: adjacency matrix and adjacency list
Graph traversals/searches: Depth-First Search (DFS) and Breadth-First Search (BFS)
General Idea of the following algorithms: topological sort, Dijkstra's algorithm (single-source, shortest path), Prim's algorithm (determines the minimum-spanning tree), TSP (Traveling-Saleperson Problem) Approximation algorithm to solve TSP, general idea of backtracking and best-first search branch-and-bound.
You should understand the graph implementations and algorithms listed above. You should be able to trace the algorithms on a given graph.