Final Exam Study Guide

1 Balanced BSTs (Chapter 19)
- Rotations
  - Single Rotations
  - Double Rotations
  - Inner vs. Outer cases
- AVL Trees
  - Recursive Definition
  - Insertion
- Red-Black Trees
  - Definition
  - Top-Down insertion
  - Top-Down deletion
- AA trees
  - Definition
  - Split and Skew

2 Spatial Data Structures
- Regular Grids
- Quadtrees
- KD-trees
  - basic structure
  - findMin/findMax
  - nearest neighbor search
    * Storing partial results - best so far
    * pruning - reduce search space by eliminating subtrees
    * traversal order - visit most promising subtrees first
  - building a balanced tree
- BSP-trees

3 Transformations
- Translation
- Scale
- Rotation
- Shear
- Linearity
- Composing transformations
- OpenGL matrix stack
- gluLookAt()

4 Lighting/Shading
- Diffuse/Lambertion
- Specular
- Ambient
- Flat vs. Smooth
- Directional vs. Point Light

5 Heaps (Chapter 21)
- Heap order property
- Storage (in an array)
- Insertion/Deletion
- Heapify/BuildHeap

6 Hash Tables (Chapter 20)
- Hash Functions
- Collisions and Collision Handling
  - Linear Probing
– Quadratic Probing (including what is required to ensure that items can always be inserted)
– Separate Chaining

7  Graphs (Chapter 14)

• Definitions
  – Vertices/nodes
  – Edges/arcs
  – Directed vs. Undirected
  – Directed acyclic graph
  – path
  – path length
  – edge cost/weight
  – simple path
• Representation
  – Adjacency Matrix
  – Adjacency List
• Algorithms
  – Breadth First Search
  – Depth First Search
  – Topological Sort
  – Single-Source Shortest paths
    * Unweighted, positive weighted (Dijkstra’s Algorithm), negative weighted
  – Strongly Connected Components
  – Minimum Spanning Trees (Kruskal’s and Prim’s algorithms)

8  Old Stuff

8.1  LinkedList (Chapter 17)

• insert/delete, updating references

8.2  Algorithm Analysis (Chapter 5)

• Definitions of $O$, $\Omega$, $\Theta$, $o$, $\omega$, $\theta$
• finding running times of algorithms

8.3  Recursion (Chapter 7)

• Basic recursion concepts
• Base case
• Inductive hypothesis
• Divide and Conquer Approach
• Pitfalls
• Dynamic Programming

8.4  Sorting (Chapter 8)

• BubbleSort
• InsertionSort
• MergeSort
• QuickSort

8.5  Stacks and Queues (Chapter 16)

• Implementation with Arrays and LinkedLists
• advantages and disadvantages

8.6  Trees (Chapter 18)

• Structure and definitions
• depth, height, size, parent, child, ancestor, descendant, leaf, path, path length
• traversals (inorder, preorder, postorder, level-order)

8.7  Binary Search Trees (Chapter 19)

• Definition
• insertion/removal
• best-case/worst-case running times