CIS101 Handout #13
David Helmbold
November 3, 1998

Written Assignment 3 - Midterm study

Study problems, do not turn in

The midterm will be held in class on November 10. The exam will be **closed book, closed notebook**. However, students will be allowed to have one (1) 3×5 card with notes on *one side*. As these cards will be collected, put your name on the other side. Here are some recommended study problems on the newer material.

1. Exercise 7.4-1 on page 149. (heap-sort example)
2. Exercise 7.5-1, 7.5-2 on page page 150. (heap insertion/deletion).
3. Exercise 7.5-3 on page page 151. (simulating fifo and stack with priority queue)
4. Exercise 12.2-2 on page 226. (Chaining example)
5. Exercise 12.4-1 on page 240. (hash example, rehashing)
6. Exercise 12.4-2 on page 240. (hash example, rehashing)
7. Suppose we hash 10 keys into a hash table of size 20. What is the expected number of collisions? (Assume that each key is equally likely to be mapped into each bin of the 20 bins.) If two keys hash into the same bin that counts as 1 collision and if three keys, a, b, and c all hash into the same bin, that counts as three collisions: $\{a, b\}$, $\{a, c\}$, and $\{b, c\}$. In general if n keys hash into m bins, what is the expected number of collisions? (Hint: consider how many new collisions are expected when each element is inserted into the table.)
8. Exercise 5.5-5 on page 96. (induction) path compression)
9. Exercise 13.1-4 on page 246. (Pre-order, post-order walks)
10. Exercise 12.2-2 on page 250. (path, left, right not ordered in BST)
11. Make up a list of insertion/deletion operations, and show the binary search tree after each operation.
12. Exercise 14.1-1 on page 265 (coloring red-black trees)
13. Exercise 14.2-3 on page 267 (rotations preserve inorder key order)
14. Exercise 14.3-3 on page 272 (insertion into red-black trees)

Some sample exam questions will be available through the class web page.

Here are the main topics that we have (or will have) covered before the midterm this quarter.

- Growth rates of functions and asymptotic notation using the book's n and c_o style definitions, for O, o, Ω, and Θ. You should know how to verify when one function is related (or not related) to another, and the relationships between the different asymptotic notations.
- ADTs, ADT interfaces, information hiding, ADT handles and header records.
- Formal proofs by induction, including induction over structures (trees)
- Recurrences and the Master Theorem
- Average case analysis, computing averages
- Priority Queues, Heaps, and Heapsort, including the packed array representation of heaps
- Dictionaries,
- Hashing – both chaining and "open-addressing" (or re-hashing)
- Basic tree definitions, tree traversals, recursive definition of binary trees
- Binary Search Trees, including insertion/deletion algorithms
- Red-Black trees: properties, insertion algorithms (you will not be tested on deletion in Red-Black trees)

Although I expect you to know how to implement and use ADTs in C, the midterm will not stress C coding (although questions on the basics of pointers/handles and high level ADT/module questions are possible).