CS 200 Fall 2014
Midterm 2 Preparation Guide

Exam Date: **November 13, 2014**
Location & Time: **In-class, 9:30 ~ 10:45 AM (and in lab that week)**

In this exam, you will have a mix of multiple-choice questions and some short answer questions. To study, you should attempt the examples and exercises at the end of indicated textbook sections/ chapters. Also review the written assignments and quizzes. Your lab section during the prior week will include some practice questions from an earlier midterm.

The lab section during the week of the midterm will comprise a programming question. Note: the question could require you to write code, read code and/or debug code. The programming topics were binary trees, binary search trees, priority queues, heaps, and hash tables.

Key concepts

The problems in this exam will be about the concepts covered in the lectures (week 8 through week 12). Please review your lecture and lab/recitation notes (https://www.cs.colostate.edu/~cs200/Fall14/home_progress.php) The distribution of points in the midterm will reflect the amount of time spent on the topic either in lecture, in the readings, in the labs or in the assignments.

Binary Tree, Binary Search tree
Lecture Notes: L9
Text book: Prichard Ch. 11; Rosen Ch. 11

1. What are Binary Trees and Binary Search Trees?
2. Search and Traversal algorithms.
3. Insertion and Deletion algorithms.
4. The terminology and definitions from Rosen
5. Implementation of BSTs.
6. Efficiency of Binary Search Tree implementations.
7. Know how to sort a list using a BST (treesort)
8. What is the complexity of treesort?
9. What are iterators? How are iterators used for traversal?
10. Be able to program BST operations

Recurrence Relations, Divide-and-Conquer and Master Theorem
Lecture Notes: Recurrence Relations
Text book: Rosen Ch. 8.1-8.3

1. What is a recurrence relation?
2. What is the relationship between recurrence relations, sequences and closed forms?

3. What is a linear, homogeneous recurrence relation and how does it relate to the Master Theorem?
4. What is the Master Theorem and what is it for?
5. How can algorithms be characterized using the Master Theorem?

Tables, Priority Queues, Heaps and Heapsort

Lecture Notes: "Tables and Priority Queues"
Text book: Prichard Ch. 12

1. What is a table? Priority queue? Heap?
2. How can you implement a table or priority queue using arrays/references/BSTs?
3. What are the important characteristics of the heap?
4. How to insert and delete items?
5. Array representations of heap.
6. Heapsort algorithm and efficiency
7. Be able to program heap operations.

Hash Tables

Lecture Notes: "Hashing"
Textbook: Prichard Ch. 13.2

1. What are important characteristics of hash functions?
2. How are hash tables implemented?
3. What are generics? How can they facilitate tables?
4. What are collisions? What is clustering?
5. How are collisions dealt with?