Mid-term Review

Lecturer: Xiaodi Wu   *Scribe: Xiaodi Wu*

1   Exam Policy

- In-class: 10/26/16. Total length: 50 mins, start promptly at 10:00am. Please be on time.

- Format: Total Score 50 (mid-term takes 25% of your final score). 3 problems with full score 9 each (easy). 1 problem with full score 10 (medium). 1 problem with full score 13 (medium and hard).

- Note: One page of handwritten[1] notes (8.5 by 11 inches, front and back) allowed.

2   General Comments

I will try to make the test short enough that well-prepared students can complete in the time allowed. Do your best to get as many points as you can: answer the questions you know first, and briefly show your thinking or write down a partial answer even when you're not complete sure.

Most of the questions will be of the form "prove X" or "show Y," with some guidance as to what methods you should use and how formal you should be. A few questions could be creative or qualitative, e.g., "Describe how to modify data structure X in order to efficiently implement operation Y" or "Which data structure would be most appropriate for problem Z?"

3   What to Study

- Solutions to assignment 1 and 2.

- Slides and notes posted online.

- Textbook Chapter 1, Chapter 2.1-2.3, 3.1.

- Your own notes from the lecture.

- Discussions on piazza. Ask questions and make good use of office hours.

4   Topics & Sample Problems

- Big-O, big-Omega, and big-Theta. For each of these, you should know the formal definition and be able to use it to prove or disprove statements.

  Sample Problems: R-1.15 to R-1.20

- Loop invariants. **Sample Problems**: lecture notes and assignment 1.

- Stacks, queues, vectors, and lists: Basic definitions and common usage.

  Sample Problems: R-2.1, slides and assignment 2.

---

[1]If extensive handwriting is a problem for you, please let me know as soon as possible and accommodations can be made.

- Amortized analysis: accounting and potential method. You will be free to use either method in your answer, but make sure you are comfortable with at least one.

  Sample Problems: C-1.1, C-1.2, lecture notes and assignment 2.

- Trees: binary trees, non-binary trees, array-based implementation of binary trees. Questions may involve traversals, insert/delete, and recursive and iterative algorithms, properties about trees.

  Sample Problems: R-2.3, R-2.4, C-2.9, C-2.10, C-2.19, C-2.20, assignment 2.

- Binary Search Tree: definition and properties of of binary search trees, insertion and removal operations.

  Sample Problems: R-3.1, R-3.2, assignment 2.