CSCI 1323 (Discrete Structures), Spring 2006

Review for Final Exam

1 Format of the exam

The exam will be at the scheduled exam period, May 6 at 8:30am. It will be about twice the length of the midterm (or a bit less) and so should take about two hours, but you will have the whole three-hour exam period if you need it. You may use your textbook and any notes or papers you care to bring (with the exception of solutions from previous years, as described in the syllabus), but you may not use other books, a calculator or computer, or (of course!) each other's papers.

The exam will be comprehensive but will focus on material since the midterm (more than half the questions/points will be about material from the second half of the course).

Most questions will be similar in form to those in the quizzes, minute essays, homework assignments, and midterm. Questions should be about as difficult as the ones on the midterm. So, it would probably be a good idea to review

- Quizzes and minutes essays (solutions online).
- Homeworks (solutions distributed, or to be distributed, in hardcopy form). If you didn't already do the "not to turn in" problems, consider doing so as a way to review.
- Midterm (solution to be distributed in hardcopy form).

2 Lecture topics to review

You are responsible for all material covered in class or in the assigned reading. (See Lecture Topics and Assignments[1] for a list of assigned reading.) You should review in particular the following topics. This list is *not necessarily exhaustive*, but should give you an idea of what topics I think are most significant.

- (*Review*) Propositional logic:
  - Translating English into propositional-logic wffs (emphasizing understanding of propositional logic connectives over ability to untangle complicated English).
  - Proving that a propositional-logic wff is a tautology using truth tables.
  - Proving that a propositional-logic wff is a tautology using proof rules.
- (*Review*) Predicate logic (propositional logic plus quantifiers):
  - Translating English into predicate-logic wffs (emphasizing understanding of quantifiers over ability to untangle complicated English).
  - Determining whether a predicate-logic wff is true in a given interpretation.
  - Proving that a predicate-logic wff is valid using proof rules.

[1] `http://www.cs.trinity.edu/~bmassing/Classes/CS1323_2006spring/HTML/schedule.html`

- *(Review)* Proof techniques:
  - Direct proofs, proof by cases, proof by contraposition, proof by contradiction.
  - Proofs by induction.
- *(Review)* Recursion and recurrence relations:
  - Recursive definitions of sequences, sets, operations, and algorithms.
  - Defining and solving recurrence relations.
- *(Review)* Analysis of algorithms:
  - Defining and solving recurrence relations to estimate the number of basic operations performed by a recursive algorithm.
- *(Review)*Proofs of program correctness:
  - Rules for assignment, conditional statements, loops.
  - Combining these rules to verify correctness of simple programs.
  - Meaning of Hoare triples.
  - Loop invariants.
- Sets:
  - Defining sets.
  - Operations on sets.
  - Countable versus uncountable sets.
- Counting:
  - Multiplication and addition principles.
  - Principle of inclusion and exclusion.
  - Pigeonhole principle.
  - Permutations and combinations.
  - Permutations and combinations with repetitions.
- Probability:
  - Basic definitions (finite and conditional probability, expected value).
- Relations:
  - Definition and properties (reflexivity, symmetry, transitivity, antisymmetry).
  - Partial orderings and topological sorting.
  - Equivalence relations and equivalence classes.
- Functions:
  - Definitions and properties (one-to-one, onto).
  - Composition and inverse functions.

  - Order of magnitude of functions.

- Graphs:
  - Definitions and terminology (just the things mentioned in class or homework problems).
  - Computer representation (adjacency matrices and adjacency lists).

- Trees:
  - Definitions and terminology (just the things mentioned in class or homework problems).
  - Recursive definition, recursive algorithms, inductive proofs.