Slide 1

Administrivia

- Homeworks 6 and 7 graded. I will mail averages so far and approximate letter grades later today.
- Reminder: Homework 8 due Monday. Solutions will be available Wednesday.
- Solutions to all quizzes on Web.
- Review sheet on Web.
- Review session next Thursday at 11;30am.
- Extra-credit problems in work. Will be on Web by Monday. Due following Monday. *Can only help your grade.* (Maximum of 30 points, which would increase your average by about 5 points.)

Slide 2

Recap — Course Goals

- Learn math needed for later courses. Something of a “grab bag” of topics, but you probably *will* see some of this material again.
- Increase “mathematical maturity” — in part, this is the ability to think logically, which can only help, both in programming and other CS courses.
- A recurring theme is to take something that might be difficult to think through from first principles and turn it into a symbol-manipulation problem.

Slide 3

Topics and Why We Include Them

- Formal logic:
  - Understanding connectives/tautologies related to simplifying boolean expressions, e.g., in programs.
  - Example of "formal system" — you will deal with others, e.g., formal grammars (basis for compilers, e.g.).
  - "Mathematical maturity".

  (Aside: Dr. Myers recommends the symbolic logic course as one of your math electives. Students who take it seem to find it worthwhile.)
- Proof techniques (direct proof, contraposition, proof by contradition, proof by induction):
  - Background for courses that involve proofs.
  - "Mathematical maturity".

Slide 4

Topics and Why We Covered Them, Continued

- Program correctness:
  - Another way to think about programs – even if not applied formally, E.g., "loop invariant" idea — recall problem with black/white marbles.

Slide 5

Topics and Why We Covered Them, Continued

- Recursion:
  - Recursive definitions used in theory courses.
  - Recursive algorithms sometimes easier to write than iterative equivalents (e.g., tree traversal).
- Analysis of algorithms:
  - Simplified version, but gives background for discussions in programming classes.
- Recurrence relations:
  - Useful in doing "analysis of algorithms" on recursive algorithms.

Slide 6

Topics and Why We Covered Them, Continued

- Sets, counting, and probability:
  - General background. (Stuff about infinite sets is a tangent, but an interesting one?)
  - Often useful to know how many cases must be considered.

Topics and Why We Covered Them, Continued

- Relations, functions, and matrices:
  - General background.
  - Background for formal study of relational databases. (Definitions of set operations needed here too.)

Slide 7

Topics and Why We Covered Them, Continued

- Graphs and trees:
  - Abstraction behind some key data structures.
  - Trees you've used already.
  - Many uses for graphs — serialization in Java, garbage collection, shortest path through a network, etc., etc.

Slide 8

Minute Essay

- None — sign in.

Slide 9