Review for Midterm 2.

Lecture 25
CS 312

The Schedule

- Project 3 due 1 week from today.
- Project 4 due December 3.
- Project 5 cancelled.
- Grading: each project counts 12.5%.
- Schedule for the rest of the semester available at class web page.

Objectives

- Compare and contrast Greedy, Dynamic Programming, Divide and Conquer and Backtracking algorithms.
- Compare and contrast top down (memory functions) and bottom up dynamic programming solutions.

Comparing Paradigms

- Greedy: simple, but need a selection fn.
- Divide and conquer: recursively break the problem into pieces. Top down.
- Dynamic programming: Solve subproblems, store the solutions and use them to build a solution. Reuse results. Bottom-up. (Except memory functions which are top-down).
- Backtracking: Search a graph, but back-up when a partial solution can not yield a complete solution.

Comparing Dyn. Programming

- Bottom-up: store solutions to all subproblems in a table, extract solutions from the table. Might compute unneeded values.
- Top-down: recursively sub-divide the problem into subproblems. Store the subproblem results as you go. Lots of recursive calls.

Objectives

- Apply dynamic programming to the coins, knapsack, chained matrix multiplication and travelling salesperson problems. Apply Floyd's algorithm.
Applying algorithms.

- Examples completed on request.

Objectives

- Apply backtracking to the knapsack, eight queens, graph coloring
- Modify a backtracking algorithm to return the composition, rather than just the value, of an optimal solution.

Applying algorithms

- Examples completed on request.
- Modifying backtracking algorithms
  – did it Monday. Can do it again if needed.

Objectives

- Design backtracking or dynamic programming algorithms.

Designing algorithms

- This will take the form “here’s a problem we haven’t seen, its solvable using an algorithm we have seen. Think. Pick the right algorithm.”
- Won’t have to design an algorithm from scratch.

Won’t be on the test...

- Term rewriting (theorem proving)