

For Monday

- Read chapter 10, section 1
- No homework

Exam 2

- Friday
- Covers material through minimum spanning trees
- Take home handed out at the exam due the following Wednesday

Research Paper

- Any questions?

Program 4

- Any questions?

Review questions?

Examples

- Traveling Salesman
- Hamiltonian Cycle
- Satisfiability (technically, 3Sat)
- Graph coloring
- Knapsack
- Bin packing

Algorithm Design

- How do you design an algorithm?
- We're going to talk in these last weeks about some different strategies for designing algorithms.
 - Greedy algorithms
 - Divide and Conquer algorithms
 - Dynamic Programming
 - Randomized
 - Backtracking

Algorithm Design Methods

- For each design method, we'll ask:
 - What is the strategy?
 - When is the strategy applicable?
 - What is the general complexity of algorithms designed using this strategy?
- We'll also look at several algorithms that fall into the design category.

Optimization Problems

- In many problems, there are two types of solutions
 - **feasible** solutions
 - **optimal** solutions
- These types of problems are called **optimization** problems

Applying Algorithms to Optimization Problems

- When applying an algorithm design method to a problem, we may get
 - an unacceptable solution
 - a feasible solution that is clearly not optimal
 - a feasible solution that is close to optimal
 - an optimal solution

Optimization Problem Examples

- Job Scheduling
- Loading
- Minimum Spanning Tree
- Traveling Salesman

Greedy Algorithms

- With the greedy method, at each step of the algorithm we make the decision that appears best at that time.
- We must always define a **greedy criterion** that determines the best decision.
- Decisions are never changed.
- Each decision must take feasibility of the final solution into account.

Making Change

Examples

- Making Change
- Shortest Path
- Topological Sort
- Minimum Spanning Tree
- Job Scheduling
- Huffman Code
- Bin Packing