Algorithms and Data Structures

- Plumbing and wiring of programs.
- Design is a very creative activity.
  - There are no formulas
  - There are tried and true methods
- Think first - program later
  - Design, design, design
  - Code
Applied Algorithm Scenario

1. Real world problem
2. Abstractly model the problem
3. Find abstract algorithm
4. Adapt to original problem

- Wrong problem
- Wrong model
- Incorrect algorithm, poor performance

Evaluate
Evaluation Step Expanded

1. Algorithm Correct?
   - yes
   - no
     - New algorithm
     - New model
     - New problem
2. Choose Data Structures
3. Performance?
   - satisfactory
   - unsatisfactory
     - New data structure
     - New algorithm
     - New model
4. Implement
Tool Kit

- Data Organization
  - Lists, trees, arrays
  - Priority queues
  - Up-trees
  - Connections between them

- Algorithmic approaches
  - Recursion
  - Divide and Conquer
  - Depth-first search
  - Greedy

- Abstractions
  - Graphs

- Algorithms
  - Sorting
  - Shortest paths
  - Huffman codes

- Analysis
  - Recurrences
  - Worst case
  - Amortized
What’s Next

• Specific Applications (CSE 417, 410, 415)
  › Graphics, simulation, games, networks, systems
• Programming Languages and Compilers (CSE 413)
  › Languages and their implementations
• Software Engineering
  › Engineering large programs
• Algorithms (CSE 417)
  › Deeper study of algorithms
  › Mathematical
• Computational Complexity (CSE 417)
  › Study of the inherent time and storage needed to solve problems.