NO TUTORIAL TODAY
Greedy Algorithms

• Local, myopic choice
• Proof of correctness: promising partial solution
• Interval Scheduling Problem [4.1]
• Generic algorithm for MST
  – Blue/Red Rules
  – Termination and Correctness
  – Kruskal’s algorithm implements the Generic algorithm.
• Schedule to Minimize Lateness [4.2]
• Tutorial: Shortest Path Problem – Dijkstra’s algorithm [4.4]
• Tutorial: Making Change.
Dynamic Programming

Reusing solutions to smaller subproblems.

- Define the array
- Give initialization and recurrence for computing values in the array.
- Program to compute the elements of the array.
- Compute solution from the array.
- Longest Common Subsequence
- Interval Scheduling with Profits
- Subset Sum
- Knapsack
- Scheduling Jobs with Deadlines, Durations and Profits
• Matrix-Chain Multiplication
• Tutorial: All Pair Shortest Paths Problem
• Tutorial: Longest Increasing Subsequence
Divide-and-Conquer

- The Master Theorem
- Mergesort [5.1]
- Strassen’s algorithm for Matrix Multiplication
- Integer Multiplication [5.5]
- Finding closest pair [5.4]
- Tutorial: Counting Inversions [5.3]
Network Flow

- Residual network
- Max-Flow Min-Cut Theorem
- Ford-Fulkerson algorithm and properties
- Edmons-Karp algorithm

4 steps in solving a problem using Network Flow

- Construct the flow network.
- Specify the algorithm to find the maximum flow in the network Describe the output.
- Argue that the max flow returned by the algorithm can be used to construct the (optimal) solution to the original problem.
- Construct the solution to the original problem from
the maximum flow.

- Bipartite Matching [7.5]
- Edge-disjoint Paths in Directed Graph [7.6]
- Baseball Elimination Problem [7.12]
Linear Programming

- A Political Problem item Formulate Max-Flow
- Set Cover
- CNF-SAT
- Tutorial: Formulate Shortest Path Problem
- Tutorial: Simple Scheduling with Prerequisites
Approximation Algorithms

- Load Balancing [11.1]
- Bin Packing
  - First-Fit
  - Best-Fit
  - Next-Fit
- Vertex Cover [11.4]