Algorithms Recap

COMP 215 Lecture 23

Basics

- What is an algorithm?
- What does it mean to analyze an algorithm?
- Worst case, best case, average case?
- Asymptotic notation
  - $O, o, \Omega, \omega, \Theta$

Basics

- Recurrence Relations
  - Mechanism for analyzing recursive algorithms.
  - Solution techniques:
    - Guess a solution and prove it using induction.
  - Some special forms have cookbook solutions:
    - linear recurrences.
    - divide and conquer recurrences.

Divide and Conquer Algorithms

- Mergesort.
- Quicksort.
- Convex Hull.
- When to use D&C and when not to.

Dynamic Programming

- Binomial coefficient .
- Floyd's Algorithm for all pairs shortest paths.
- 0-1 Knapsack.
- Principle of Optimality.

Greedy Algorithms

- Minimum Spanning Tree
  - Prim's Algorithm
  - Kruskal's Algorithm
- Single Source Shortest Path
  - Dijkstra's Algorithm

Backtracking/Branch and Bound

- Backtracking
  - N-queens.
  - Sum of subsets.
  - Graph Coloring.
- Branch and Bound (Best First Search)
  - 0-1 Knapsack.
  - Traveling Salesperson.

Heapsort

Complexity of The Sorting Problem

- Lower bound when only one inversion is removed per comparison.
- Lower bound for any comparison based sort.
  - Minimum height of a decision tree that has n! leaves.

Linear Time Sorts

- Radix Sort
- Bucket Sort

Complexity of Search

- Search:
  - Minimum height of a decision tree that has n nodes.

Hashing

- What are the odds that hashing will be slower than binary search?

Selection Algorithms/Adversary Arguments

- min, max, max/min, kth element, kth element in guaranteed linear time.

Complexity Theory

- P, NP
- Cook-Levin Theorem.
- polynomial many one reductions. NP-Completeness.
- Turing reductions. NP-Hardness.
- Other classes:
  - coNP, PSPACE, EXP

Approximation Algorithms for NP-Hard Problems

- Approximation algorithm for a special version of TSP
- Demonstrated that the general TSP problem can only be approximated if P=NP.