Topic Ω: Course Overview

CPS 230, Fall 2001

Reminder of tools, topics we have covered:
• Asymptotic notation for representations, running times
• Analysis frameworks: Worst-case, best-case, average-case, probabilistic (randomized), amortized
• Divide-and-Conquer paradigm (Strassen’s algorithm, quicksort, merge sort)
• Solving recurrences (small cases, substitution (guess and prove by induction), iteration, Master method)
• Comparison-based sorting—insertion, merge, quick
• Lower bounds and adversaries
• Linear-time sorting—counting, bucket, radix
• Linear-time median finding and order statistics
• Dynamic data structures: binary search trees, red-black trees, splay trees, B-trees, heaps, union-find
• Hashing, universal hashing
• Dynamic programming
• Greedy algorithms: when locally optimal \(\implies\) globally optimal (e.g., Huffman codes, Prim’s algorithm for MST, Kruskal’s algorithm for MST)
• Amortization: cost of sequences of operations (e.g., splay trees)
• External memory algorithms—need to build locality into algorithms since data transfer is done via large contiguous blocks (e.g., distribution paradigm, merge paradigm, B-trees)
• Single-source shortest paths (Dijsktra, Bellman-Ford)
• All-pairs shortest paths as matrix multiplication or repeated squaring (Floyd-Warshall)
• P, NP, and NP-Completeness!
• Approximation algorithms (vertex cover, traveling salesperson)—WILL NOT BE ON FINAL
Themes

Techniques:
- Divide-and-Conquer, greedy, dynamic programming
- Lazy algorithms (for good amortized performance or for batched dynamic external memory algorithms)
- Adversary for lower bounds
- Invariants (correctness, efficiency!)
- Approximation algorithms

Transformation and reuse:
- Heaps/sorting; Prim/Dijkstra, all pairs shortest paths
- Data structure augmentation (R-B)
- Tabulating and reusing solutions (Dynamic Programming)
- Duality (merge/distribution)
- NP-completeness (reduction arguments)

$L' \leq_p L$ means that $L$ is at least as hard as $L'$, up to a polynomial factor in the running time.

Thank you, and good luck!