Homework 4

Assigned November 17, 2019
Due November 25, 2019 on Canvas

Design and Analysis of Algorithms
(CSCE 423/823, Fall 2019)

CSCE 823 students have to do all problems for full credit. CSCE 423 students need to do only the Core Problems for full credit, and may do the Advanced Problem for bonus points.

For this homework assignment, you are to work in the team that you established in Homework 0. This will be your collaborative team for the rest of the term. You may freely discuss solutions to exercises within your team, and you are to submit a single pdf file from your team. **The internet is not an allowed resource on this homework!**

Clarity of presentation is important. You should give a clear description of all your algorithms, each with a proof of correctness and analysis of time complexity. You must submit your solutions in a single pdf file via Canvas, and are encouraged to prepare your solutions in LaTeX. **Only** pdf will be accepted, and you should submit only one pdf file for Questions 2–4. When you submit for Question 1, submit a second pdf file to the *Questions* assignment in Canvas.

Core Problems

1. **(bonus points, but mandatory submission)** Present one question that you have on the lectures and/or textbook on either all-pairs shortest paths or lower bounds. This question should be thoughtful and nontrivial, and suggest depth of knowledge in the material. Also, present what you consider to be a reasonable (doesn't have to be completely correct) answer to this problem.

   Your question and answer should be submitted to the *Questions* assignment in Canvas in a pdf file separate from the rest of your homework submission.

2. **(20 + 5 possible bonus points)** Give an efficient algorithm based on SLOW-ALL-PAIRS-SHORTEST-PATHS to find the length (in terms of number of edges) of a minimum-length negative-weight cycle in a directed, weighted graph. Argue your algorithm's correctness and time complexity. (If your algorithm is based on FASTER-ALL-PAIRS-SHORTEST-PATHS, your algorithm will be more efficient and you will obtain 5 bonus points if your algorithm is correct.)

3. **(15 pts)** It is known that a max heap on n elements can be built in time $O(n)$, even though building a binary search tree on n elements requires time $\Omega(n \log n)$. Why does the $\Omega(n \log n)$ lower bound proof for binary search trees not apply to heaps?

Advanced Problem

4. **(40 points)** There is an alternative way to reconstruct shortest paths in Floyd-Warshall by replacing the matrix $\Pi^{(k)}$ with $\Phi^{(k)} = \left(\phi_{ij}^{(k)}\right)$, where $\phi_{ij}^{(k)}$ is the highest-numbered intermediate vertex of a shortest path from i to j in which all intermediate vertices are from the set $V_k = \{1, 2, \ldots, k\}$. Give a recursive formulation of $\phi_{ij}^{(k)}$, modify Floyd-Warshall to compute all values of $\phi_{ij}^{(k)}$, and rewrite Print-All-Pairs-Shortest-Path to print the shortest path from vertex i to vertex j using the matrix $\Phi = \Phi^{(n)}$ instead of $\Pi = \Pi^{(n)}$. Argue that your modified algorithms are correct and analyze their time complexities.