CMPSC 465 — Data Structures & Algorithms
Spring 2021 — Paul Medvedev and Chunhao Wang

# HW 6

## Due March 11, 10:00 pm

**Instructions:** You may work in groups of up to three people to solve the homework. You must write your own solutions and explicitly acknowledge up everyone whom you have worked with or who has given you any significant ideas about the HW solutions. You may also use books or online resources to help solve homework problems. All consulted references must be acknowledged.

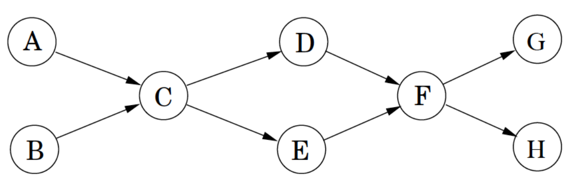
You are encouraged to solve the problem sets on your own using only the textbook and lecture notes as a reference. This will give you the best chance of doing well on the exams. Relying too much on the help of group members or on online resources will hinder your performance on the exams.

Late HWs will be accepted until 11:59pm with a 20% penalty. HWs not submitted by 11:59pm will receive 0. There will be no exceptions to this policy, as we post the solutions soon after the deadline. However, you will be able to drop the three lowest HW grades.

For the full policy on HW assignments, please consult the syllabus.

1. **(0 pts.) Acknowledgements.** The assignment will receive a 0 if this question is not answered.
   (a) If you worked in a group, list the members of the group. Otherwise, write "I did not work in a group."
   (b) If you received significant ideas about the HW solutions from anyone not in your group, list their names here. Otherwise, write "I did not consult without anyone my group members".
   (c) List any resources besides the course material that you consulted in order to solve the material. If you did not consult anything, write "I did not consult any non-class materials."

2. **(16 pts.) Linearization** Run the DFS-based topological ordering algorithm on the following graph. Whenever you have a choice of vertices to explore, always pick the one that is alphabetically first.




   (a) Indicate the *pre* and *post* numbers of the nodes.
   (b) What are the sources and sinks of the graphs?
   (c) What linearization is found by the algorithm?
   (d) How many linearizations does this graph have?

3. **(16 pts.) DFS variations** You are given a set of cities, along with the pattern of highways between them, in the form of an undirected graph $G = (V, E)$. Each stretch of highway $e \in E$ connects two of the cities, and you know its length in miles, $\ell_e$. You want to get from city $s$ to city $t$. There's one problem: your car can

only hold enough gas to cover $L$ miles. There are gas stations in each city, but not between cities. Therefore, you can only take a route if every one of its edges has length $\ell_e \leq L$. (Note: Only use the algorithms that have been covered in class so far to solve the below problems.)

(a) Given the limitation on your car's fuel tank capacity, show how to determine in linear time $O(|V|+|E|)$ whether there is a feasible route from $s$ to $t$. State your algorithm clearly, prove that it is correct and analyze its running time

(b) You are now planning to buy a new car, and you want to know the minimum fuel tank capacity that is needed to travel from $s$ to $t$. Give an $O((|V|+|E|)\log|V|)$ algorithm to determine this. State your algorithm clearly, prove that it is correct and analyze its running time. (Hint: Use the algorithm in (a) as a subroutine. Consider all the possible values for the minimum tank capacity. How many are there? What would be the running time of the algorithm that tries them all? Then, try to improve this algorithm.)

**4. (10 pts.) One-Way Streets** The police department in State College has made all streets one-way. The mayor contends that there is still a way to drive legally from any intersection in the city to any other intersection, but the opposition is not convinced. A computer program is needed to determine whether the mayor is right. However, the city elections are coming up soon, and there is just enough time to run a *linear-time* algorithm.

(a) Formulate this problem graph-theoretically, and explain why it can indeed be solved in linear time.

(b) Suppose it now turns out that the mayor's original claim is false. She next claims something weaker: if you start driving from town hall, navigating one-way streets, then no matter where you reach, there is always a way to drive legally back to the town hall. Formulate this weaker property as a graph-theoretic problem, and carefully show how it too can be checked in linear time. (Hint: in the meta-graph, what kind of component should the town hall be in?)