COMPSCI 311: Introduction to Algorithms **Fall 2019**

Homework 2

Released 9/23/2019 Due 10/7/2019 11:59pm in Gradescope

Instructions. You may work in groups, but you must write solutions yourself. List collaborators on your submission. Also list any sources of help (including online sources) other than the textbook and course staff.

If you are asked to design an algorithm, please provide: (a) the pseudocode or precise description in words of the algorithm, (b) an explanation of the intuition for the algorithm, (c) a proof of correctness, (d) the running time of your algorithm and (e) justification for your running time analysis.

Submissions. Please submit a PDF file. You may submit a scanned handwritten document, but a typed submission is preferred. Please assign pages to questions in Gradescope.

1. **(10 points) One-Way Path** In a game, a set of nodes is connected via some set of one-way edges. At every node there is an object to pick up. Design an algorithm to find a path that you can follow to collect all objects, if this is possible. To make your task easier, you know that starting from any node, no matter what path you follow, you will never get back to the same node.

2. **(10 points) Marching Penguins** It is well known that penguins march. In one particular colony, any two penguins have grown to establish a marching order between them: one of them is willing to follow the other but not the other way around. Given a group of penguins, can you design an algorithm that lines them up such that every penguin is willing to follow the one immediately in front of it?

3. **(20 points) Diameter and Average** For an undirected graph G its *diameter* $dia(G)$ is defined as the maximum distance between any pair of nodes (the distance is the number of edges on the shortest path). We can also define the average distance over all $n(n-1)/2$ distinct node pairs: $avgd(G) = 2\frac{\sum_{\{u,v\}\subseteq V} dist(u,v)}{n(n-1)}$. Clearly, as the maximum, the diameter is at least the average. But does there exist a constant c such that for any connected graph G, $\frac{dia(G)}{avgd(G)} \leq c$? Prove or give a counterexample.

4. **(20 points) Message Transmission** A network of n observation posts p_i transmit optical messages between them. Transmissions are scheduled at given timepoints: in total, there are m transmissions (p_i, p_j, t_k) (post p_i transmits to p_j at time t_k). A post always transmits all messages ever received. If transmissions in a chain $p_{i_1} \rightarrow p_{i_2} \rightarrow \ldots \rightarrow p_{i_k}$ occur at the same time, messages from p_{i_1} will get to p_{i_k}.

   Design an algorithm that given such a network can answer in time $O(m+n)$ whether it is possible for an observation that is collected by post p_i at time t_s to reach post p_j by time t_f.

5. **(20 points) Round the Clock Scheduling** Consider a set of jobs, each with a regular schedule (starting time and finishing time) every day. There may be jobs that straddle midnight (e.g., 10 p.m. to 1 a.m.). Only one job can be processed at a time. Design a polynomial-time algorithm that chooses a schedule of jobs to be repeated every day, maximizing the number of jobs processed.

6. **(20 points) Optimal Snow Plowing** A particular state has a set of E roads among its set of V cities, where the time to traverse the road from city u to a neighbor city v is given by c_{uv}. (Note that c_{uv} need not equal c_{vu} – in fact there may be no road from v to u.) After a snowstorm, some of the roads are impasable, but the governor needs to drive from city s to city t very soon, so soon that there is time to clear only one of the impassable roads. Give an $O(E \log V)$ algorithm that determines which road to plow (one only) in order to achieve the minimum possible time path from a city s to city t. The input is a list of all the roads, the values c_{uv} for each road, s, t, and the set of impassable roads. If road clearing is of no help, the algorithm should say so.

7. **(0 points).** How long did it take you to complete this assignment?