Algorithm Design and Analysis

Assignment 2

Deadline: Apr 6, 2022

1. (20 points) **Amortized Cost of ADD.** Let us consider the following situation. An integer (initially zero) is stored in binary. We have an operation called ADD that adds one to the integer. The cost of ADD depends on how many bit operations we need to do. (one bit operation can flip 0 to 1 or flip 1 to 0.) The cost can be high when the integer becomes large. Use amortized analysis to show the amortized cost of ADD is $O(1)$. You should define the potential function in your analysis.

2. (25 points) We have learned from the lecture that Dijkstra meet some problems when we have negative weights. The following algorithm attempts to find the shortest path from node s to node t in a directed graph with some negative edges:

   - Add a large enough number to each edge weight so that all the weights become positive, then run Dijkstra's algorithm.

   Intuitively, the algorithm is not correct if there exists a negative cycle in the graph. However, does this approach work for the directed acyclic graphs?

   (a) (10 points) Given a DAG, either prove this algorithm correct on DAG, or give a counter-example.

   (b) (15 points) Consider a special directed layered graph (DLG) $G = (V, E)$. Each vertex has a layer v_ℓ chosen from $\{1, 2, 3, 4, ..., L\}$, and the layer of s is 0. We only have directed edges between adjacent layers, i.e., if $(u, v) \in E$, then $\ell_v = \ell_u + 1$. Either prove this algorithm correct on DLG, or give a counter-example.

3. (25 points) Given a directed graph $G(V, E)$ where each vertex can be viewed as a port. Consider that you are a salesman, and you plan to travel the graph. Whenever you reach a port v, it earns you a profit of p_v dollars, and it cost you c_{uv} if you travel from u to v. For any directed cycle in the graph, we can define a profit-to-cost ratio to be

$$r(C) = \frac{\sum_{(u,v)\in C} p_v}{\sum_{(u,v)\in C} c_{uv}}.$$

As a salesman, you want to design an algorithm to find the best cycle to travel with the largest profit-to-cost ratio. Let r^* be the maximum profit-to-cost ratio in the graph.

   (a) (15 points) If we guess a ratio r, can we determine whether $r^* > r$ or $r^* < r$ efficiently?

   (b) (10 points) Based on the guessing approach, given a desired accuracy $\epsilon > 0$, design an efficient algorithm to output a good-enough cycle, where $r(C) \geq r^* - \epsilon$. Justify the correctness and analyze the running time in terms of $|V|$, ϵ, and $R = \max_{(u,v)\in E}(p_u/c_{uv})$.

4. (30 points) Let $G = (V, E)$ be an undirected connected graph. Let T be a depth-first search tree of G. Suppose that we orient the edges of G as follows: For each tree edge, the direction is from the parent to the child; for every non-tree (back) edge, the direction is from the descendant to the ancestor. Let G' denote the resulting directed graph.

   (a) (5 points) Give an example to show that G' is not strongly connected.

   (b) (5 points) Prove that if G' is strongly connected, then G satisfies the property that removing any single edge from G will still give a connected graph.

   (c) (5 points) Prove that if G satisfies the property that removing any single edge from G will still give a connected graph, then G' must be strongly connected

   (d) (15 points) Give an efficient algorithm to find all edges in a given undirected graph such that removing any one of them will make the graph no longer connected.

5. (5 points) **Bonus Problem.** Let G be a directed graph. Suppose that G is not strongly connected, and you are allowed to add extra edges into G. What is the smallest number of extra edges to make G strongly connected? Write down your idea and intuition, try to prove it if you believe you are right.

   (a) (5 points) Write down your ideas and guesses to earn the 5 points!

   (b) (0 points) Write down the complete proof for happiness.

6. How long does it take you to finish the assignment (include thinking and discussing)? Give a score (1,2,3,4,5) to the difficulty. Do you have any collaborators? Write down their names here.