CSCE629 Analysis of Algorithms

Homework 6

Texas A&M U, Fall 2019 *10/07/19*

Lecturer: Fang Song *Due: 10am, 10/18/19*

Instructions.

- Typeset your submission by LaTeX, and submit in PDF format. Your solutions will be graded on *correctness* and *clarity*. You should only submit work that you believe to be correct, and you will get significantly more partial credit if you clearly identify the gap(s) in your solution. You may opt for the "I'll take 15%" option.
- You may collaborate with others on this problem set. However, you must **write up your own solutions** and **list your collaborators and any external sources** for each problem. Be ready to explain your solutions orally to a course staff if asked.
- For problems that require you to provide an algorithm, you must give a precise description of the algorithm, together with a proof of correctness and an analysis of its running time. You may use algorithms from class as subroutines. You may also use any facts that we proved in class or from the book.
- **If you describe a Greedy algorithm, you will get no credit without a formal proof of correctness, even if your algorithm is correct.**

This assignment contains 3 questions, 4 pages for the total of 60 points and 0 bonus points. A random subset of the problems will be graded.

1. (15 points) (Print neatly) Consider the problem of neatly printing a paragraph with a mono-spaced font (all characters having the same width) on a printer. The input text is a sequence of n words of length $\ell_1, \ell_2, \ldots, \ell_n$ measured in characters. We want to print this paragraph neatly on a number of lines that hold a maximum of M characters each. Our criterion of "neatness" is as follows.

   If a given line contains words i through j, where $i \leq j$, and we leave exactly one space between words, the number of extra space characters at the end of line is $M - j + i - \sum_{k=i}^{j} \ell_k$, which must be non-negative to fit the words on the line. We wish to *minimize* the sum, over all lines except the last. Describe and analyze an algorithm (both time and space) to print a paragraph of n words neatly.

2. (Shortest path with bounded negative edges) Suppose we are given a directed graph G with weighted edges and two vertices s and t.

   (a) (10 points) Describe and analyze an algorithm to find the shortest path from s to t when exactly one edge in G has negative weight.

   (b) (15 points) Describe and analyze an algorithm to find the shortest path from s to t when exactly k edges in G have negative weights. How does the running time of your algorithm depend on k?

3. (Arbitrage) Arbitrage is the use of discrepancies in currency exchange rates to transform one unit of a currency into more than one unit of the same currency. For example (exchange rates not up to date), suppose 1 US dollar buys 71 Indian rupees, 1 Indian rupee buys 1.6 Japanese yen, and 1 Japanese yen buys 0.0093 US dollars. Then by converting currencies, a trader can start with 1 US dollar and buy $71 \times 1.6 \times 0.0093 = 1.0565$ US dollars, thus making a profit of 5.65 percent.

   Suppose that you are given n currencies $c_1, c_2, \ldots, c_n$ and an $n \times n$ table R of exchange rates, such that one unit of currency c_i buys $R[i, j]$ units of currency j.

   (a) (10 points) Describe and analyze an algorithm to determine whether or not there exists a sequence of currencies $\langle c_{i_1}, \ldots, c_{i_k} \rangle$ such that

   $$R[i_1, i_2] \cdot R[i_2, i_3] \cdots R[i_{k-1}, i_k] \cdot R[i_k, i_1] > 1 \,.$$

   (b) (10 points) Describe and analyze an algorithm to print out such a sequence if one exists.