cs473: Algorithms	Assigned: *Thu., Sep. 10, 2020* Updated: *Tue., Sep. 15, 2020*

Problem Set #2 (v2)

Prof. Michael A. Forbes	Due: *Thu., Sep. 17, 2020 (5:00pm)*

Some reminders about logistics.

- **Submission Policy:** See the course webpage for how to submit your pset via gradescope.
- **Collaboration Policy:** For this problem set you are allowed to work in groups of up to three. Only one copy should be submitted per group on gradescope. See the course webpage for more details.
- **Late Policy:** Late psets are not accepted. Instead, we will drop several of your lowest pset problem scores; see the course webpage for more details.

All problems are of equal value.

1. Consider as input n functions on non-negative integers $T_1, \ldots, T_n : \mathbb{N} \to \mathbb{N}$, where lookup in these tables can be done in constant time. Given a non-negative integer $k \geq 0$, find non-negative integers $k_1, \ldots, k_n \geq 0$ to minimize the sum $\sum_{i=1}^{n} T_i(k_i)$, subject to the constraint that $\sum_{i=1}^{n} k_i = k$.
   (a) Describe an algorithm to find an optimal solution $k_1, \ldots, k_n$ that runs in time polynomial in k and n.
   (b) Describe an algorithm to find the optimal *value* $\sum_{i=1}^{n} k_i$ that runs in time polynomial in k and n, *and* uses $O(k)$ space.
2. We have seen an algorithm to solve the minimum weight dominating set problem in a given node-weighted tree. Consider the following generalization. The input now consists of a tree $T = (V, E)$ with non-negative integer weights $w : V \to \mathbb{N}$ and also an integer k. Describe an efficient algorithm that computes the weight of the minimum weight dominating set with $\geq k$ nodes.

   Hint: Consider using the algorithm from problem (1).
3. Erickson Chapter 9, #6 (`http://jeffe.cs.illinois.edu/teaching/algorithms/book/09-apsp.pdf`). Assume that all exchange rates are positive.