Homework Assignment #5
Due: July 10, 2020 at 12:00 noon

1. Recall the fundamental theorem of arithmetic: every integer greater than 1 can be represented as a product of prime numbers and this representation is unique (except for the order of the factors). In other words, any integer n greater than 1 has a unique representation of the form

$$n = p_1^{e_1} \cdot p_2^{e_2} \cdot \dots \cdot p_k^{e_k}$$

where $p_1 < p_2 < \cdots < p_k$ are all prime numbers and $e_1, \dots e_k$ are positive integers. For example, $2020 = 2^2 \cdot 5 \cdot 101$. Note that 1 is not considered a prime number.

In this assignment we shall use this theorem to come up with a new data structure for representing integers. The idea is to represent an integer n as some kind of binary search tree whose nodes contain its distinct prime factors $p_1, p_2, \dots, p_k$. The node containing p_i will have another field that stores e_i too. In the special case where $n = 1$, we use an empty tree.

The operations in this question will have to do comparisons and additions on the prime factors p_i and the exponents e_i. Throughout this question, assume that if two such numbers have d_1 bits and d_2 bits when represented in binary, then they can be compared in $O(\min(d_1, d_2))$ time and added in $O(\max(d_1, d_2))$ time. (This is the time needed if comparisons and additions are done bit-by-bit, which is necessary if the numbers are too big to fit into a single word of memory.)

(a) Give a simple proof that k, the number of distinct prime factors needed to represent n, is $O(\log n)$.

Remark: In fact, it is also known that $k \in O(\frac{\log n}{\log \log n})$ but this is harder to prove.

(b) Consider the operation DIVISIBLE-BY-PRIME(n, p) which returns true if integer n is divisible by the given prime number p and false otherwise. Assume n is given as a BST data structure. Describe how to implement this operation efficiently and give an upper bound on its worst-case running time in terms of n and p.

(c) Consider the operation DIVISIBLE-BY(n, m) which returns true if integer n is divisible by integer m. You can assume $m \leq n$. Assume both n and m are given as BST data structures. Describe how to implement this operation efficiently and give an upper bound on its worst-case running time in terms of n and m.

(d) Consider the operation MULTIPLY(n, m) which returns the product of the integers n and m. Assume both n and m are given as BST data structures. The output should also be a BST data structure. Describe how to implement this operation efficiently and give an upper bound on its worst-case running time in terms of n and m.

(e) Describe one disadvantage of this data structure for representing integers.

(f) **For EECS5101 students only**: Consider the operation DIVISORS(n) that returns the number of positive integer divisors of n. For example, DIVISORS(2020) would return 12 because 2020 has 12 divisors: 1, 2, 4, 5, 10, 20, 101, 202, 404, 505, 1010 and 2020. Assume n is given as a BST data structure. Describe how to implement this operation efficiently and give an upper bound on its worst-case running time in terms of n.