Homework Assignment #7
Due: March 26, 2019 at 11:30 a.m.

1. Let U be an ordered set. Suppose you are given two linked lists A and B, each containing n distinct elements of U. You wish to check whether the two lists contain the same set of elements (i.e., whether the set of elements that appear in A is the same as the set of elements that appear in B).

Assume that elements of U can fit into a single word of memory and that operations on words of memory (e.g., comparisons, arithmetic) can be done in constant time.

(a) Give a simple and efficient comparison-based algorithm that solves the problem. The worst-case time and space used by your algorithm should depend on n, and not on the size of U. State the worst-case time and space used by your algorithm as a function of n.

(b) For the remaining parts of the question, assume $U = \{0, 1, 2, \ldots, N-1\}$ where N is a very large number. Describe how you can use a hash table with chaining (i.e., representing each bucket with a linked list) to solve the problem.

(c) Give a good upper bound on the expected running time of your algorithm in part (b) in terms of n if your table has n buckets and you choose a hash function randomly and uniformly from a universal class of hash functions. Justify your answer. (The expectation should just be over random choices made by the algorithm; do not assume any probability distribution on the inputs.)

(d) Explain how to solve the problem so that the expected running time is the same as the algorithm in (b), but the worst-case time is no worse than the algorithm in part (a).