Union and Intersection

The two main functions for this task were union, combining the list together, and intersection, getting the identical nodes in the linked list. For union, it traverses through both linked lists and adds them both to a new linked list. Linked lists have a space complexity of O(n) and traversal of it is also linear. For intersection, I put more focus on speed by using a dictionary to keep track of the node values and the amount of time it occurs in the list. The main intersection function goes through a linked list and then checks with the dictionary to add to a new list the intersected values. The alternative would be to use lists because of the ease of access and traversal, it comes at a time efficiency cost however.