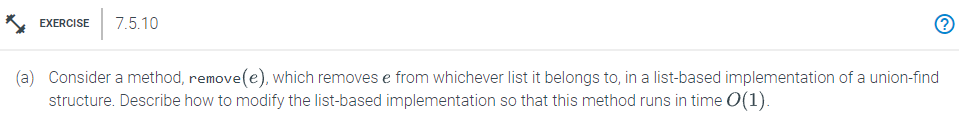
# Michael Chillemi

# 06/28/2023

# CS 590 - Algorithms

# M6.B2: Module 6 Union-Find Structures Creativity Exercises

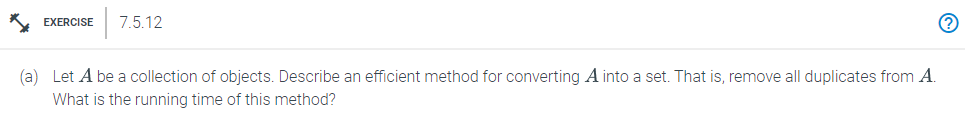
Problem 7.5.10



Answer:

This approach entails keeping track of the elements in the list that occur before and after the one that is now being referenced. The components before and after it can be connected without needing to go through e when e is removed from the equation. Then we will set e pointers to NULL this will result in the creation of a delete operation, which will finish in O(1) time.

Problem 7.5.12



Answer:

First, we sort the items of A using a quick sorting technique, such as Mergesort, with a time complexity of O(n log n). The duplicates can then all be eliminated by iteratively going over the sorted sequence. The duplicates will be next to one another. Sorting takes O(n lg n) time, and removing duplicates takes O(n) time. This algorithm is O(n lg n) in size overall. Once the list is sorted and the duplicates are removed we can now iterate through the list. At each iteration we will take the current element call the makeSet() function to turn it into a singleton set. Last since there is no duplicates in the list we can call the union() function to make a union between the singleton set and the final set that is being created.