

CS 140A - Homework 2

Due Wednesday, October 22

1. Write a recursive function to find a particular element in a singly linked list. What is the running time of the algorithm?
2. Write an efficient (in both time and space) algorithm to reverse the items in a doubly linked list. Do not consider what type of item may be stored in the list. Your algorithm should have the same time complexity regardless of the item type. Give the time complexity of your algorithm.
3. Draw a singly linked list which contains three items, A, B, and C. The head pointer points to the A item. Now, explicitly list the pointer operations required to:
 - (a) insert D at the head of the list (assume you have a pointer to node D)
 - (b) append D to the end of the list (assume you have a pointer to node D)
 - (c) insert D before B in the list (assume you have a pointer to node D)
 - (d) delete A
 - (e) delete B

All of these operate on the original list. Draw a picture for each update and show the necessary pointer code.

4. A self-adjusting list is like a regular list, except that all insertions are performed at the head, and when an element is accessed through *Search*, it is moved to the head of the list without changing the relative order of the other items. The *Search* function returns either a TRUE or FALSE based on whether the item was found. Write the *Insert* and *Search* functions for a self-adjusting list implemented with an array. What are the time complexities of the functions? (Your algorithm should not take into consideration the type of item in the list).
5. Write the *Insert* and *Search* functions for a self-adjusting list implemented with a linked list. What are the time complexities of the functions?
6. Show how to implement a queue using two stacks. Analyze the running time of the queue operations. You do not have to actually write the code, you may just describe the method and use pictures, but be sure you are clear.
7. Show how to implement a stack using two queues. Analyze the running time of the stack operations. You do not have to actually write the code, you may just describe the method and use pictures, but be sure you are clear.