Recommended Approach

1. Viewed lecture materials and read textbook (mainly sections 9.5 and 12.2). Don't worry about the "applications" they are trying to solve, focus on just what a linked list is and how it works.
2. Read the comments in List.h with a focus on abstract list operations that will be available to an "app" program.
3. Read the comments in DataItem.h and SingleLinkNode.h with a focus on how they support a linked list.
4. Read the general comments in LinkedList.c with a focus on how a linked list implements basic list operations. Worry about the detailed function comments as you are coding each function. Note that each function's "using" focus is repeated from the list header file but it is the implementation notes that will help you understand what that specific function needs to do to the linked list.
5. In main(), comment out the 10 lines that work with mylist2 and get mylist1 working first!
6. Code the following linked list functions in the order shown
   (bodies are currently empty in LinkedList.c):
   a. newList()
   b. listLength()
   c. isListEmpty()
   d. isListItem()
   e. getListItem()
   f. setListCursorFirst
   g. setListCursorLast()
   h. setListCursorNext()
   i. appendListItem()
   j. setListCursorBack(), setListCursorAt(), and hasNextListItem() might be useful at this point
7. For initial testing, comment out lines 88-106 and activate the code on lines 111-112 and use a breakpoint on line 79 to view result of previous cycle as list grows item-by-item.
8. Code the following functions:
   a. findListItem()
   b. deleteListItem()
9. Reactivate lines 88-106 to test above functions.
10. Code the following:
    a. insertListItem() – it will be very similar to appendListItem()
11. Reactivate the lines with mylist2 and lines 113-114 to test the above function.

12. Code the following:
    a. freeList() and clearList()
13. At this point the existing main program should produce the results provided
14. In main(), copy lines 55-131 and paste over the comment lines at 141-143. Then modify the code to create the first sorted list. You will need to code the following function:
    a. findListItemAscend()
15. Then, copy and paste your code for step 14 and modify it to create the second sorted list. You will need to code the following function:
    a. findListItemDescend()
16. Code the remaining linked list functions:
    a. isListFull()
    b. insertListItemAt()
    c. appendListItemAt()
    d. deleteListItemAt()
    e. replaceListItem() and replaceListItemAt()
    f. getListItemAt()
    g. Anything else I may have missed...
17. Add extra actions to main to test the above features by leaving either mylist1 or mylist2 intact and perform the latest functions.

Most of the functions are very short (<10 lines of code) and only 3 of the functions require more than 15 lines of code! **READ THE COMMENTS** and take it step-by-step and test along the way...