

18. Write a function that counts the nodes in a linked list.
 19. Build a complete `List` class, using a linked-list implementation as described in this section. For basic operations it should have a constructor, destructor, copy constructor, assignment, and the basic list operations: empty, traverse, insert, and delete. Also, include a linear search operation to search the linked list for a given item, returning a pointer to a node containing the item in its data part, or a null pointer if it is not found.
 20. For the `List` class in Exercise 19, add a member function to reverse the linked list; that is, the last node becomes the first node, and all links between nodes are reversed.
 21. For the `List` class in Exercise 19, add a boolean-valued function that determines whether the data items in the linked list are arranged in ascending order.
-

A skeleton of link list implementation is attached. You must also include the test/main function to test the class.