Documentation

The implementation of this linked list uses two classes, the first being the node class, which is responsible for managing the nodes themselves, and the second, the Linked List class, which manages the organization of the list. The linked list class has the same functions as the Nodes class. When a method of the linked list class is called the linked list class first checks the list to see if it exists and if it does, it calls the same method on the head node object which has access to the data members and methods of the Node class and completes that instruction. The implementation of the Iterative reverse creates a node to track the previous node and a node to track the current node. Then it disconnects the previous node from the current node, and reassigns if to the previous node, reconnects the list and continually does that until the end of the list. The Recursive method exists in the node class and creates a temporary node that points at the next node of the object, it then reassigns that objects next to the previous node then recursively calls passing the next node and once the base case at the end of the list, it swaps the last node and returns the current object in reverse order.