Homework #3 - Linked Lists Intro

Goal: To understand how to work with basic methods of a singly linked list.

Assignment: Complete the .cpp file so that all tests succeed. Please read the instructions for these methods carefully. I will be grading based on your ability to meet these requirements.

In the following file, complete the following methods for both the singly linked list linked list.

- `T getFifthElement() const`. This method returns the data at the fifth node of a linked list (the count starts at 1, not at 0). It should throw int error (i.e. throw 1;) if there is no fifth element.
- `void insertNewFifthElement(const T& value)`. This method inserts a node containing `value` between the existing 4th and 5th nodes, so that the original 5th node becomes a 6th node. If the collection has only 4 values, then insert it as a new last value. If the collection has only 3 or fewer values, don't insert.
- `void deleteFifthElement()`. This method deletes the 5th node. If there was a 6th node, the 4th node now points to the 6th node. If there was no 6th node, the 4th node becomes the new back node.
- `void swapFourthAndFifthElement()`. This method rearranges the 4th and 5th nodes. You are ***not*** allowed to swap the data in the nodes, you may instead only rearrange pointers. If there were only 5 items, then the 4th item becomes a new fifth item.

For each of these methods, remember to draw out the algorithm on paper, and trace the process through an exact sequence of steps. Also, it's highly effective to organize methods into sections of scenarios, going from the most specific to the most general. The lecture videos give many hints and strategies.

Note that because this class inherits from a base class, to access the data members, you need to always use this->. Do not create those three data members again in the derived class. (The detailed reason why is given here: https://isocpp.org/wiki/faq/templates#nondependent-name-lookup-members).