

Project 1: Commentary

- **What is the computational complexity of the methods in the implementation?**
 - `splitSpring()`
 - Helper method used to split a given string into a string vector. The extraction is utilized to extract the first input as well as (if needed) index.
 - Takes in a string, and runs `std::stringstream ss(input)`, which is a computational complexity of $O(n)$, n being amount of words within the input sentence separated by whitespace.
 - `cleanUpString()`
 - Helped method used to essentially reverse `splitSpring()` method, while removing the quotes ("").
 - Runs through the vector of strings (the entire sentence input separeated by whitespaces) twice, thus $O(2n)$ or properly written $O(n)$.
 - `insertEnd()`
 - Inserts new node at the end of LinkedList.
 - Runs for $O(n)$.
 - `insertIndex()`
 - Inserts new node at the given index.
 - Worst case for $O(n)$, best case $O(1)$, realistically $O(\text{inputted index})$.
 - `deleteIndex()`
 - Deletes at the given index.
 - Same as above; worst case for $O(n)$, best case $O(1)$, realistically $O(\text{inputted index})$.
 - `edit()`
 - Edit a given index.
 - Same as above; worst case for $O(n)$, best case $O(1)$, realistically $O(\text{inputted index})$.
 - `print()`
 - Print the values of all nodes of the LinkedList.
 - Exactly $O(n)$, as it has to traverse through the LinkedList to print.
 - `search()`
 - Searches within the LinkedList.
 - Traverses through the entire list to search, thus $O(n)$.
- **Your thoughts on the use of linked lists for implementing a line editor. What are the advantages and disadvantages?**
 - More bothersome than it needs to be. Utilizing `vector<vector<string>>` would've been much easier, and allowed much more versatility. LinkedList is disadvantageous due to their lack of native support by C++, like vectors.

- Advantages include memory size (small), being a dynamic data structure, and finally relatively easy to implement.
- **What did you learn from this assignment and what would you do differently if you had to start over?**
 - If I was to start over, I would probably have implemented `splitSpring()` and `cleanUpSpring()` differently.
 - Didn't learn much from the assignment -- we programmed a `LinkedList` that was honestly 5x harder during Programming 2 (double linked list).