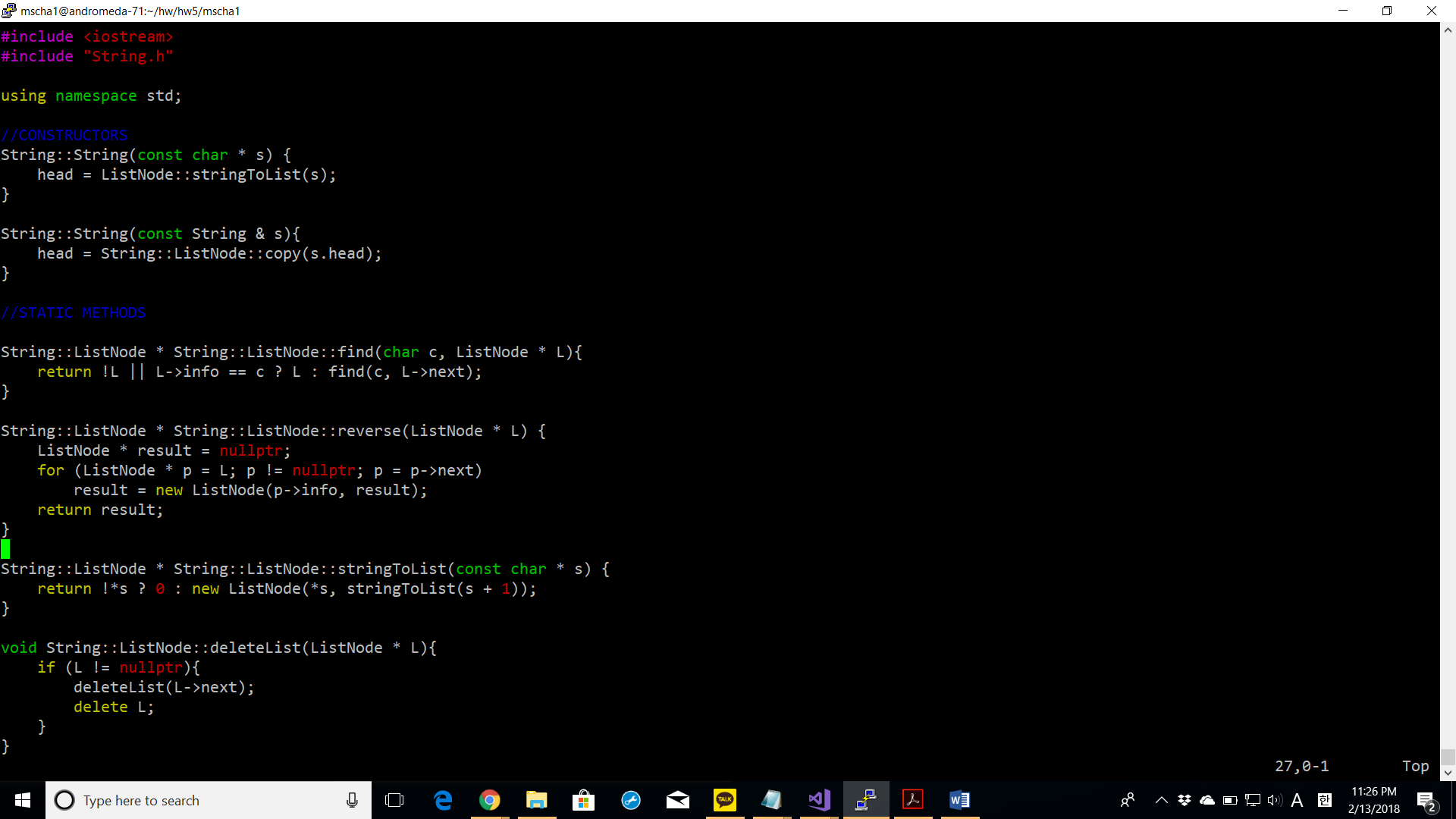
Name: Min Sung Cha Student ID: 85408485

Analysis of a few functions and any embedded functions

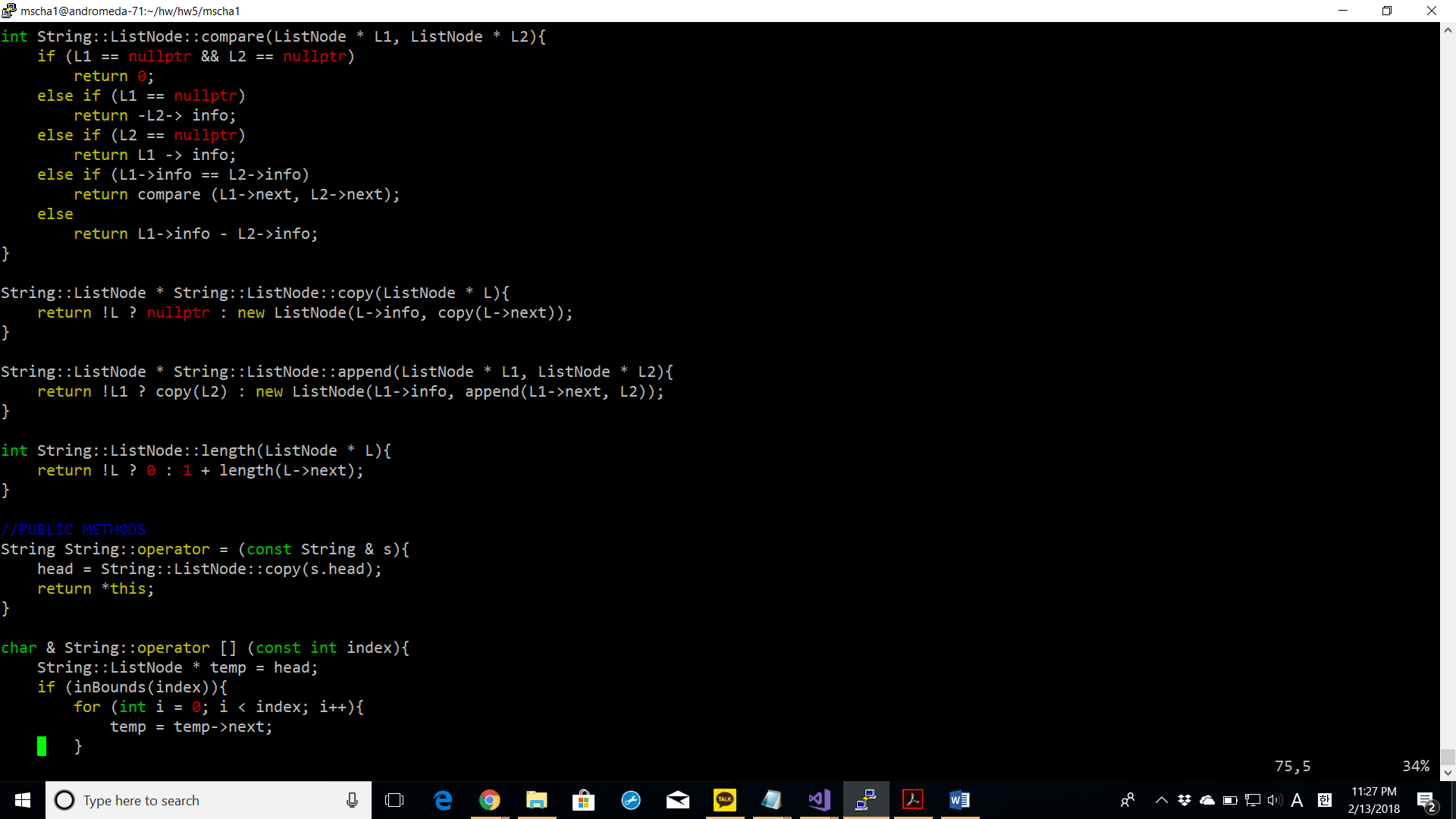
**String.cpp**



stringToList: this function recursively converts a c-string into a linksed list by first checking in its base case if the character is null. If not, it creates a new ListNode by recursively calling itself on the remaining characters.

reverse: this function reverses the linked list by creating a new linked list in the reversed order. It loops over the current linked list and creates the new linked list by creating new nodes with the next being preceding values.

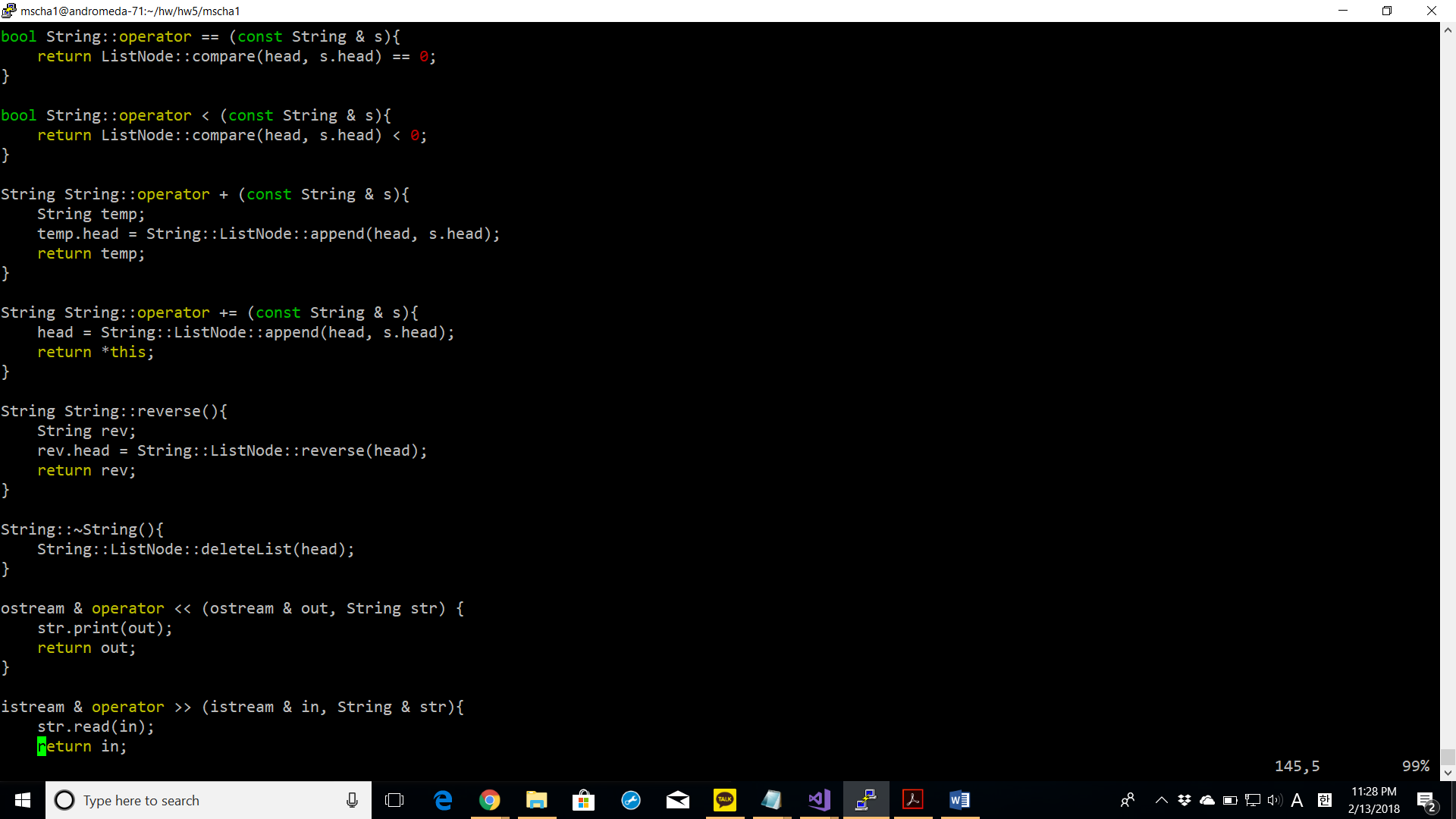
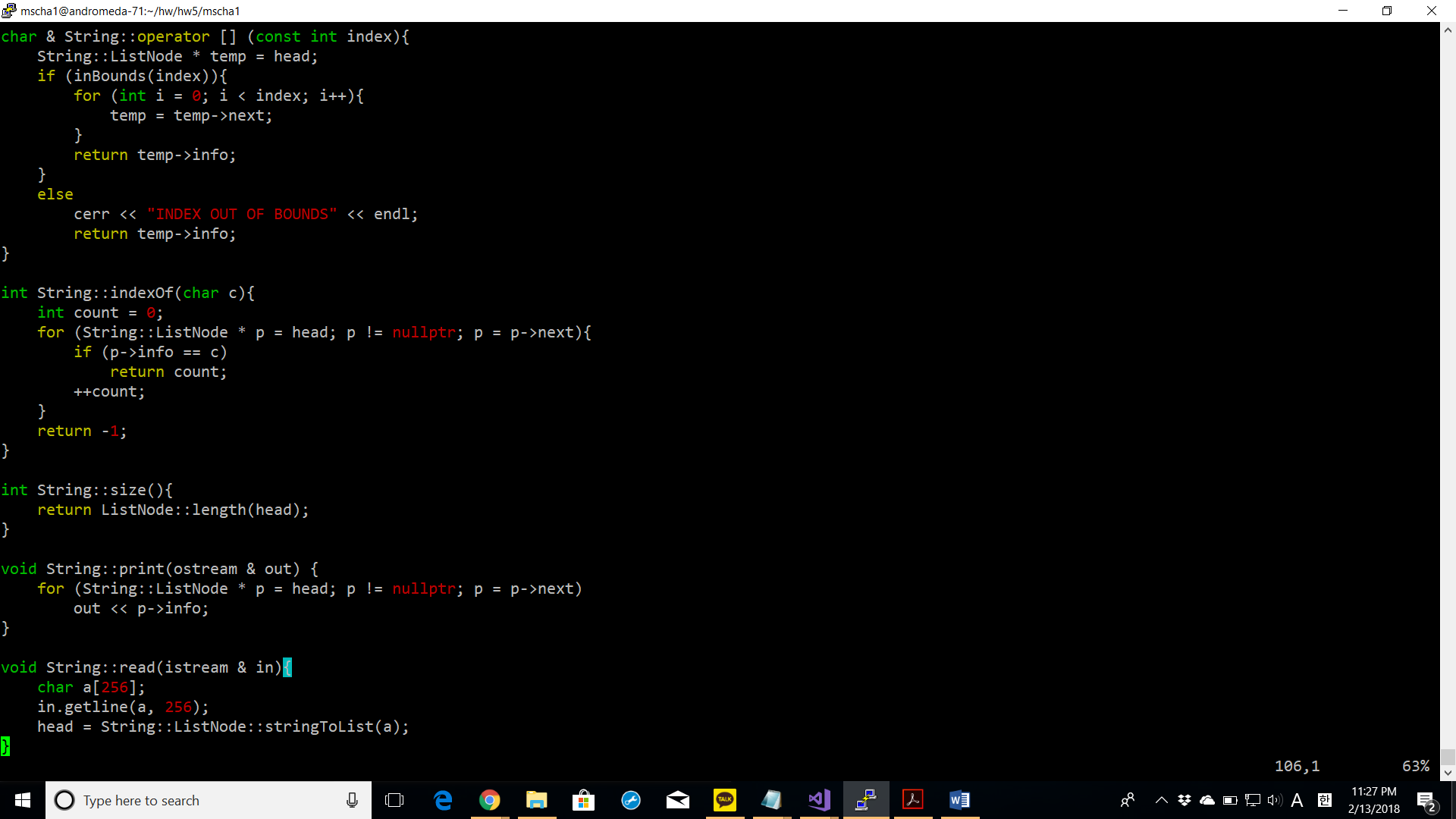
deletelist: it deletes all the ListNodes in the linked list by first checking if the node is nullptr. If not it recursively calls itself until reaching the nullptr and then deletes the all the nodes in the linked list.



copy: this function recursively copies a linked list into another linked list. In the base case, it checks whether the ListNode is a nullptr. If not, it makes a new list node and recursively calls itself on further list nodes.

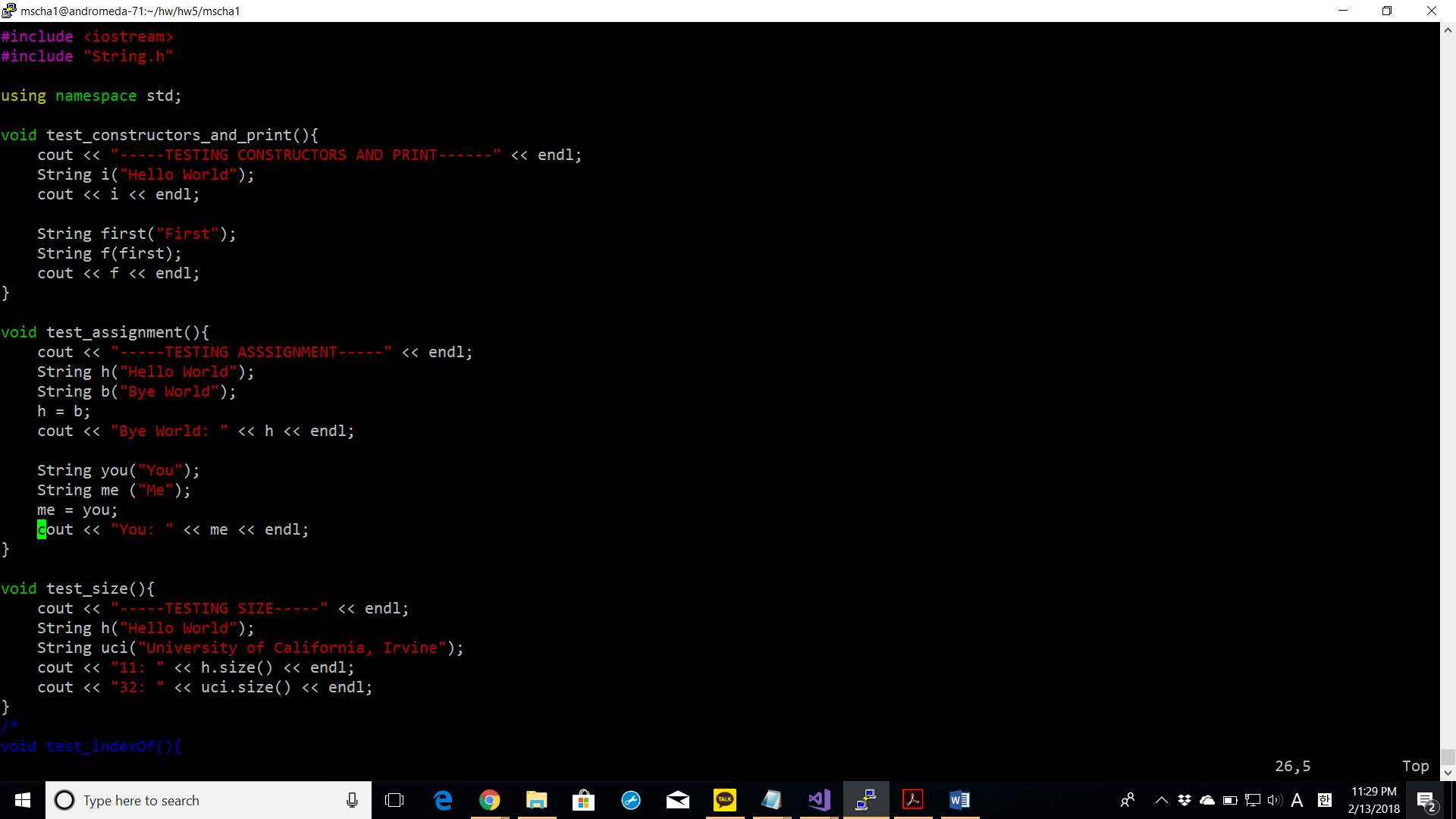
Append: it creates a new linked lists with both linked lists passed in as parameters added together. In the base cases it checks whether it has reached the nullptr in the first linked list and copies the second linked list into the new linked list. If not, it recursively calls itself to store all the nodes of the first linked list into the new linked list.

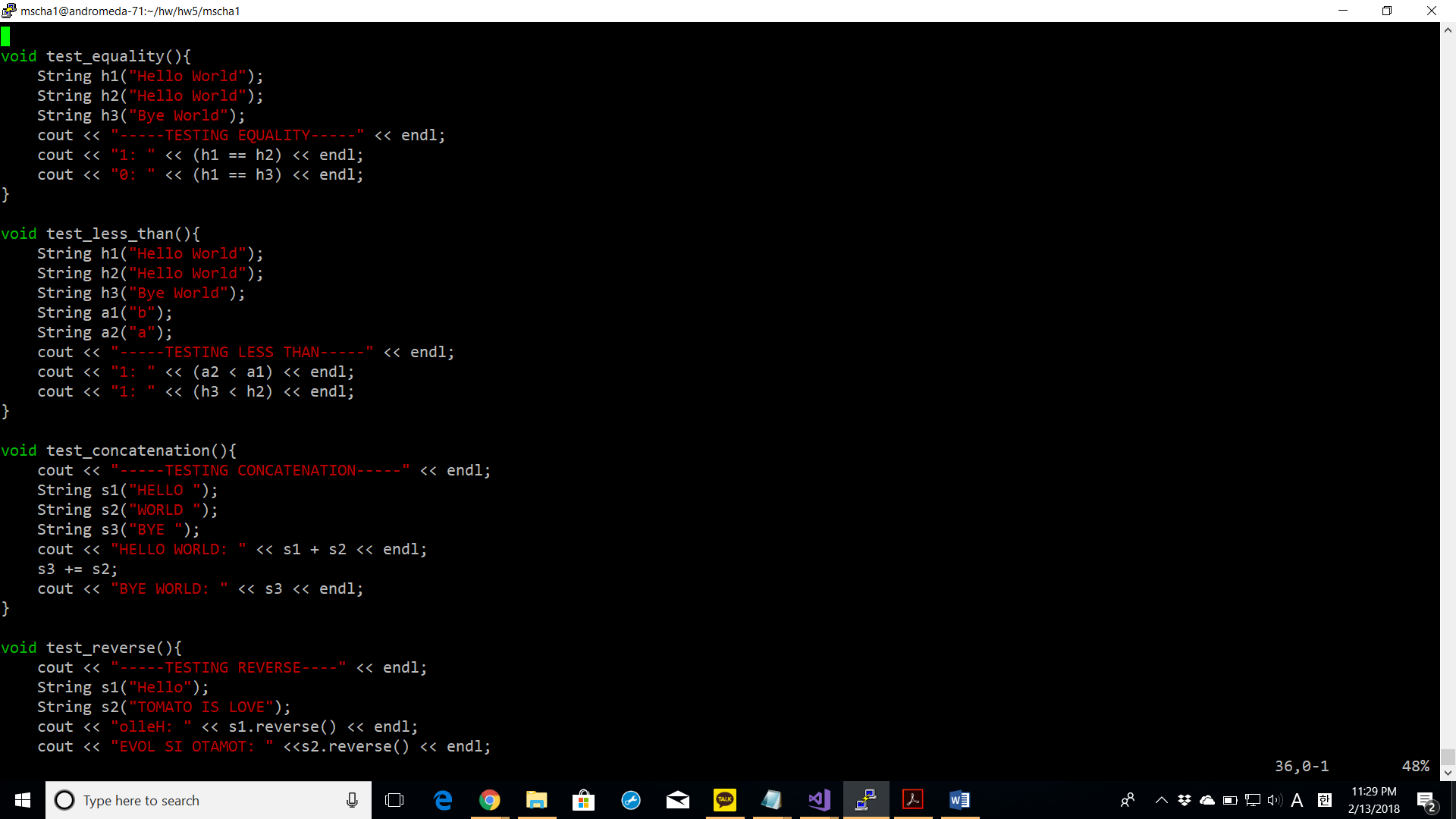
Length: computes the length of the linked list by recursively calling itself until reaching the base case (0) and adding 1 to itself.

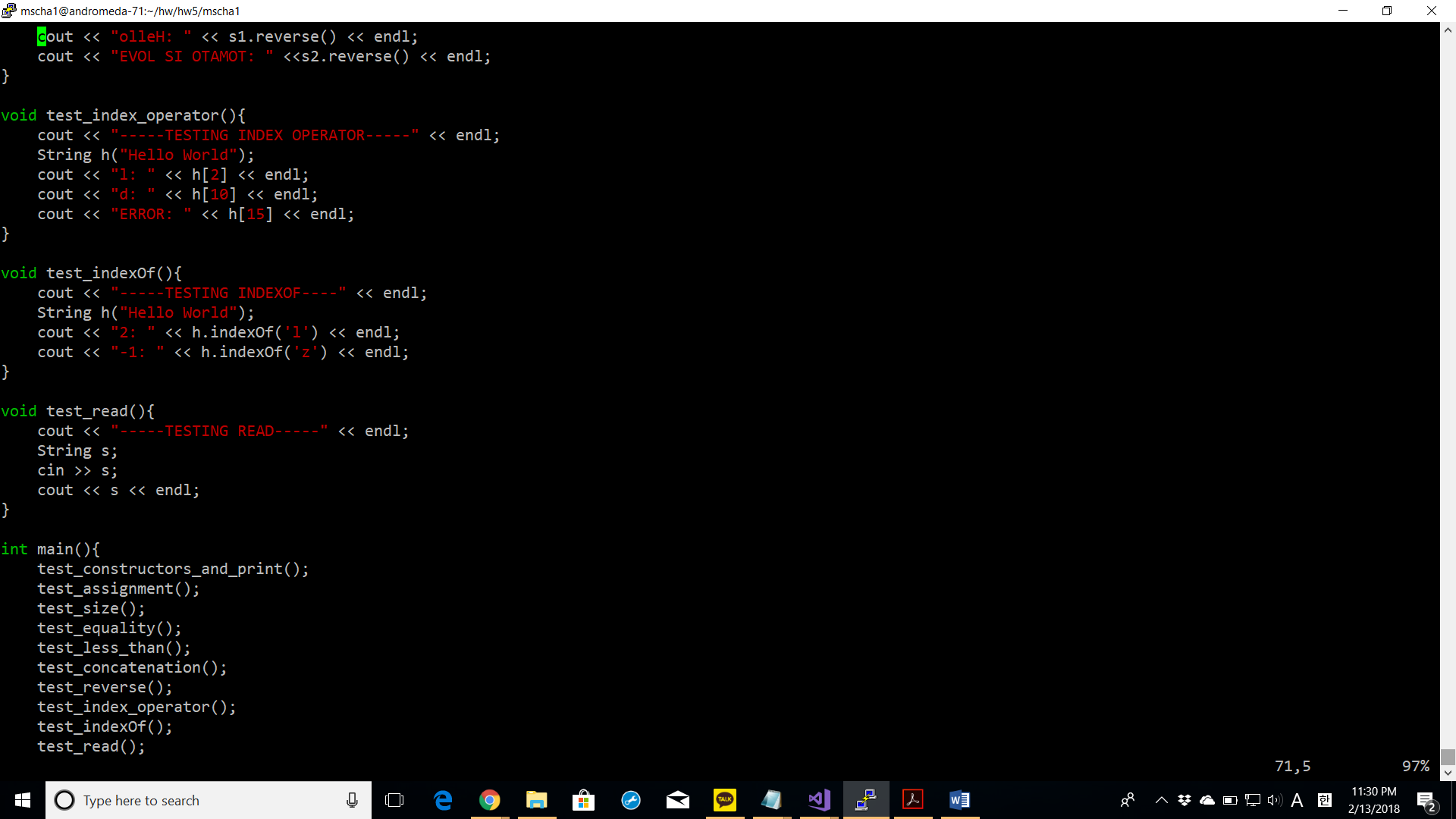


Operator += : This function utilizes the static compare function defined inside the struct ListNode to create an entirely new ListNode object with both the String objects in s and head added together. Then, the head of this string is assigned to the new ListNode object.

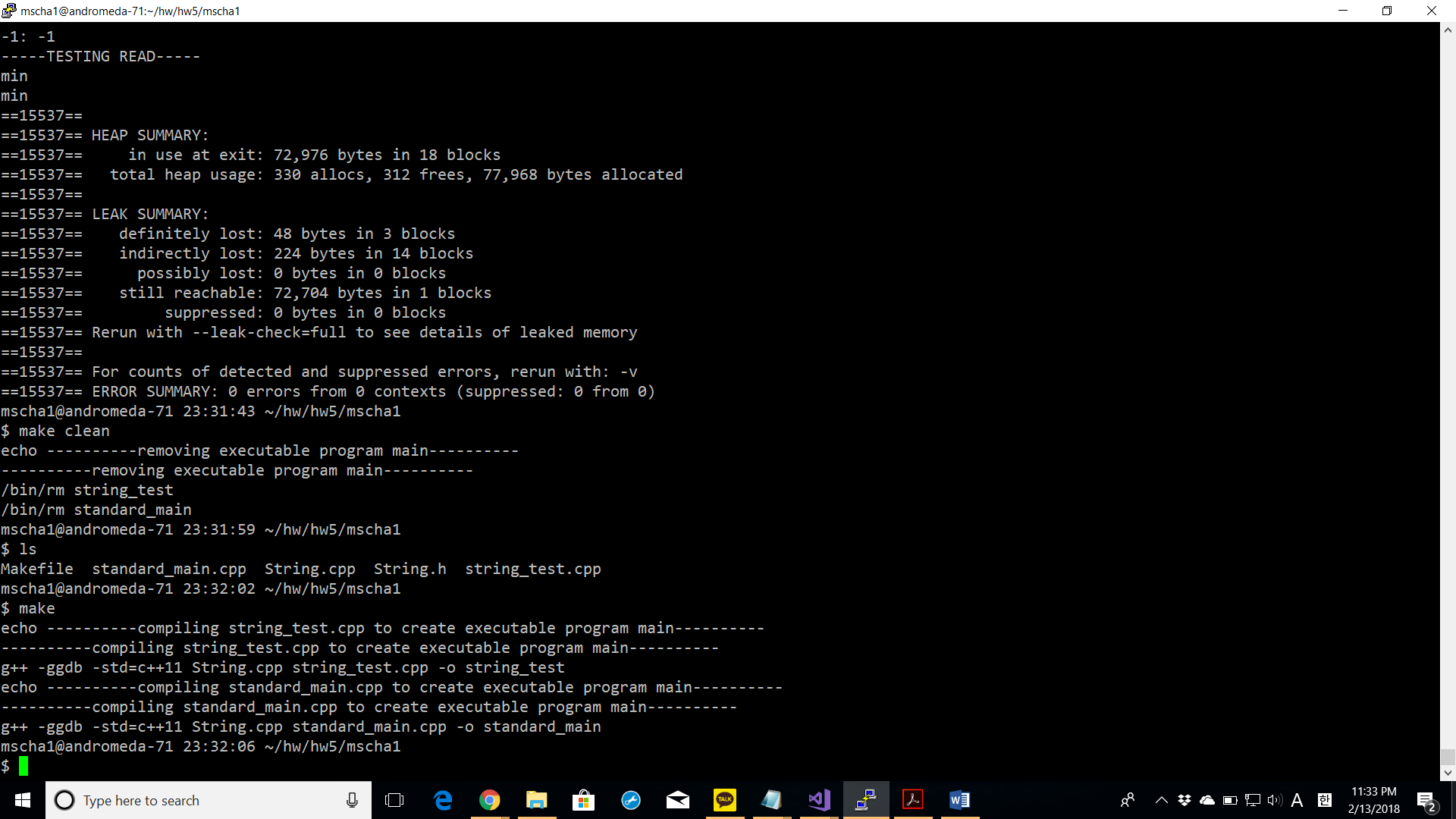
**string\_test.cpp**







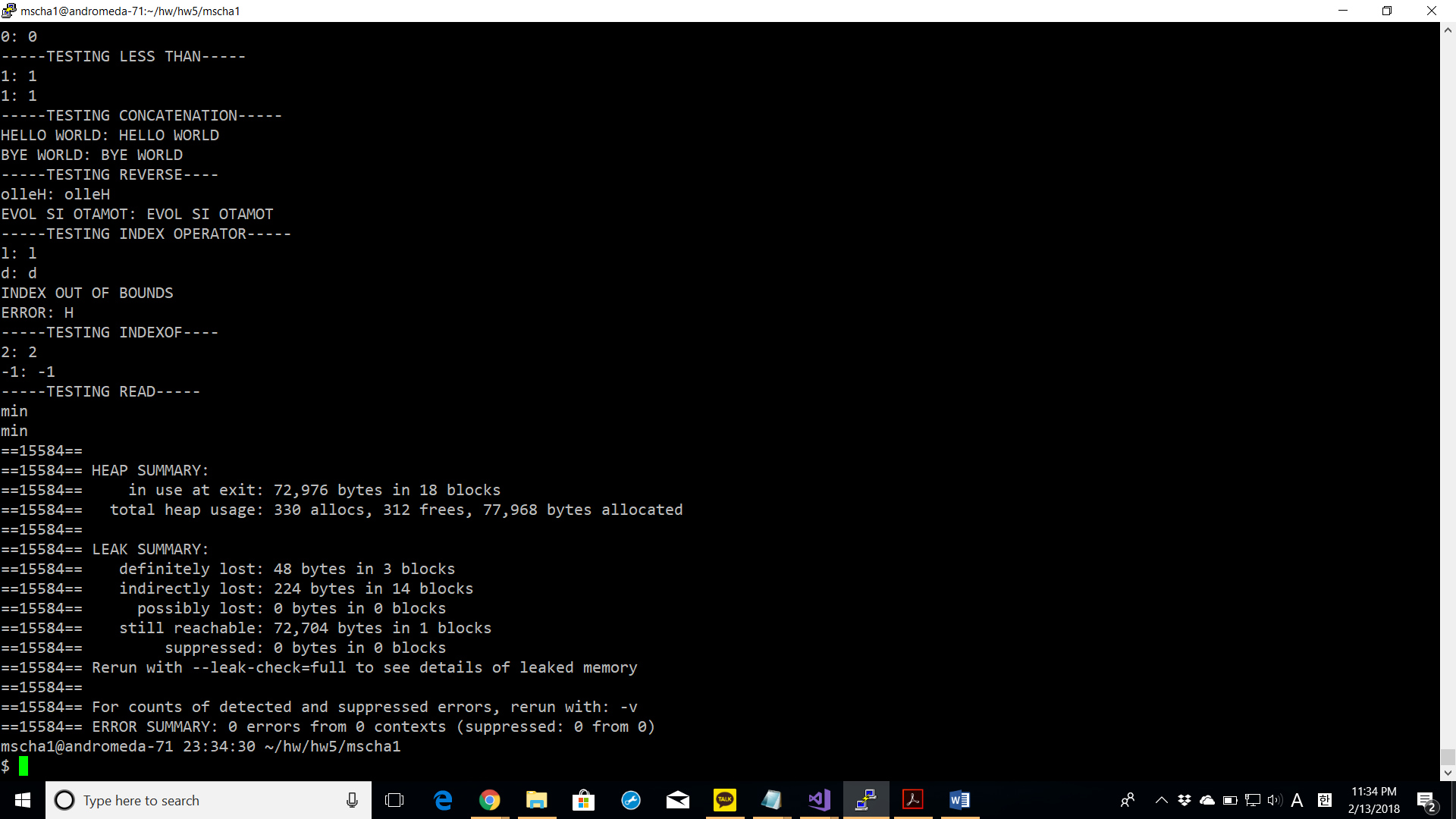
Screenshot of successful compiling with make command



My make command compiles both of the main functions at the same time.

Screenshots of successful execution with valgrind command

**valgrind string\_test.cpp**



**valgrind standard\_main.cpp**

