CS 303 Assignment 2 ReadMe

# Singly Linked Lists

The push\_front() function takes an input and inserts it at the front of the linked list. The function then sets that node as the head of the list. If the list is empty, the item is set as the head and the tail. The number of items in the list is then increased by 1.

A screen shot of a computer program

AI-generated content may be incorrect.

The push\_back() function takes an input and inserts it at the end of the linked list. The function then sets that node as the tail of the list and makes the previous tail point to it. If the list is empty, the item is set as the tail and the head. The number of items in the list is then increased by 1.

A screen shot of a computer program

AI-generated content may be incorrect.

The pop\_front() function removes the first item in the list by setting the head to the next item in the list. The number of items in the list is then decreased by 1. If the list is empty, an error is thrown.

A computer screen with text on it

AI-generated content may be incorrect.

The pop\_front() function removes the last item in the list by iterating through the list up to the item before the tail and setting its reference to null. The node before the just removed tail is then set to the tail. The number of items in the list is then decreased by 1. If the list has only 1 item, the head and tail are both set to null. If the list is empty, an error is thrown.

A computer screen shot of a program code

AI-generated content may be incorrect.

The front() function works by returning the data at the head of the list. The back() function works by returning the data at the tail of the list.

A screen shot of a computer program

AI-generated content may be incorrect.

The empty() function returns the statement “num\_items == 0.”

A computer screen with text and symbols

AI-generated content may be incorrect.

The insert() function takes an index and input and places the item in the desired location in the list by iterating through the list and creating a new node that points to the next item in the list. If the index is 0, the push\_front() function is called. If the index is the same as num\_items, the push\_back() function is called. The number of items in the list is then increased by 1. If the index is greater than the number of items in the list, an error is thrown.

A computer screen shot of a program code

AI-generated content may be incorrect.

The remove() function takes an index and removes the item at that index. The function returns true or false if the item was removed or not. The function works by iterating through the list to the item that points to the indexed item and is then changed to point to the item after the indexed item then the number of items in the list is decreased by 1. If the index points to the start or end of the list, the pop\_front() and pop\_back() functions are called, respectively; the function the returns the value of *true* in every instance mentioned. If the index is equal to or greater than the number of items in the list, an error is thrown and the function returns false.

A computer screen shot of a program code

AI-generated content may be incorrect.

The find() function takes an item and returns the index of the item. If the item is as the front or back of the list, confirmed with the front() and back() functions, the index 0 or num\_items – 1 is returned, respectively. If the item is not at the front or back of the list, the function iterates through the list looking for the item in a for loop then returns i when the item is found. If the item is not found then the number of items in the list is returned. If the list is empty, the number of items in the list is returned and an error is thrown.

A screen shot of a computer code

AI-generated content may be incorrect.

The class created in this code is a template class; therefore, the desired data type of the list must be explicitly instantiated at the end of the SinglyLinkedLists.cpp file.



# Stack

The push() function in the stack class takes an item and inserts it onto the stack. The function utilizes the push\_back() function from the vector class.

A computer screen shot of text

AI-generated content may be incorrect.

The pop() function removes the item at the top of the stack and returns the value to the user. The function saves the value to a variable, then uses the pop\_back() function from the vector class to remove the item, then returns the variable the item from the top of the stack. If the stack is empty, an error is thrown.

A computer screen with text

AI-generated content may be incorrect.