Michael Kearns

Professor Carpenter

ELEC-3150-03

11/7/19

<div align="center">Lab05</div>

This program operates as a doubly linked list utilizing memory pointers and functions to organize and alter a list of Grammy winners. The algorithm used to achieve this is a doubly linked list at its core with added functionality to complete the set forth requirements.

The algorithm adds entries to the linked list from a text file at the start of the program. Each entry is sorted alphabetically. The positioning of an entry in the list is handled by memory pointers for both next and previous entries. There are three main operations to adding an entry to the list. These are adding to the front, the middle, and the end of the list. Each is unique in how they handle the pointers because the front and end entry need to be ground, either the previous pointer or next pointer respectively. All other main functions simply navigate the list to complete their operation. These operations include adding an entry, deleting an entry, printing the list, searching for an entry, and deleting the list.

A new entry is created using the "new" keyword to allocate memory for the entry. This new entry location is then filled with information and pointed to the appropriate next and previous entry. When an entry is added to the front of the list, its previous pointer is grounded to zero. When an entry is added to the end of the list, its next pointer is grounded to zero. The previous front and end entries are then pointed to the new entry, either the previous pointer to new entry or next pointer to new entry.

This program allows the user to obtain an alphabetical list of all Grammy winners since 2000, add new winners, delete winners, and search for a winner by either name, year, or number of wins. When a new entry is added it is put into alphabetical order with the current list. When an entry is deleted it no longer appears in the list, and the list maintains its order.