Assignment 3 report

Kevin Chou

October 13, 2015

Program Description

This program demonstrates doubly linked lists. Part 1 creates a list of elements/nodes, inserts elements in the front and back, and deletes elements from the list. Part 2 applies the concept of lists and uses it to create a phonebook.

Purpose of the assignment

The purpose of the assignment is to understand how a doubly linked lists functions and how to apply a doubly linked list for real-world programs.

Data structures and algorithms description

The data structure learned in this assignment are doubly linked lists. This is implented to make a phonebook program that stores a persons name, UIN, and phone number and is able to search for a person by the last name, first name, and UIN.

Program organization and description

Three .h files were made for this assignment. One header was for the doubly linked list program, one was a template version for the doubly linked list, and the last was the a header for the record phonebook program. The record program has a class which contains variables storing people's names, UIN, and phone number. The doubly linked list program has a class that contains nodes and functions needed to work with the doubly linked list. The functions included were insert and remove front and back functions.

Instructions to compile and run your program

The files for each part of the program are separated into different folders. There is a folder for doubly linked list, template doubly linked list, simple doubly linked

list, and the phonebook program. The file to run is DoublyLinkedList.cpp, TemplateMain.cpp, SimpleDoublyLinkedList.cpp, and Main.cpp respectively.

Input and output specifications

The output for the linked list programs just list out the elements in the list. The phonebook program sorts the people by their last names and has the search function at the bottom. The phonebook program gets a segmentation fault when the name of the person is not capitalized.

C++ object oriented or generic programming features

Object oriented features include classes, inheritance, polymorphism, and templates. Classes were used for all of the programs and the doubly linked list header file was templated. The templated doubly linked list header was used for the phonebook program.

Tests

The doubly linked list programs work correctly and the phonebook program works mostly. It gets errors when the input isn't exactly the program wants.