Program #2

CISC 3130

Spring 2016

Description

Implement and compare a singly linked list, a doubly linked list with a header node, and a vector. You will be given an implementation of a singly linked list.

Requirements

- Copy the singly linked list implementation provided and modify it to make it a doubly linked list with a header node. Call the header file for the doubly linked list dllist.h.
- In main.cpp, implement the selection sort algorithm we covered in class.
- In main.cpp, implement the function insertNextToMatch, which takes a Container and an element, finds the element in the container, and inserts it next to its match.

  For example, if vector v has the elements `[18, 3, 19, 6, 5, 10, 20, 17]`, after calling `insertNextToMatch(v, 6)` it will look like this: `[18, 3, 19, 6, 6, 5, 10, 20, 17]`.

  After calling `insertNextToMatch(v, 5)`, it will look like this: `[18, 3, 19, 6, 14, 5, 5, 10, 20, 17]`.

  After calling `insertNextToMatch(v, 12)`, it will be unchanged, because 12 is not in the container and so will not be inserted.
- main.cpp will add `n` elements to the singly linked list, the doubly linked list, and a vector; it will sort each one using selectionSort, and add `n` more elements to each one using insertNextTo. It will time each task. Run main.cpp with the following values for `n`: 100, 1000, 10000, 50000. Record how long each task takes under each condition.
- Write a short discussion of how long each task took for each container and each data size and of what factors affect this. How many operations are performed for each method for each container?

Submission

In Blackboard, submit

- Your dllist.h
- Your main.cpp, which should have your implementations of selectionSort and insertNextToMatch
- The text file discussion.txt, filled out.