Data Structures Assignment 3 Report

# DLinkedList Implementation

The doubly linked list is template and uses a template node. As I described in a previous assignment, there is no reason to create a cpp implementation file for template classes since they will not be portable in a library if this is done.

The head of the list has garbage data and points to the first real data node. The head always exists and is never NULL.

Sorting is performed recursively. As such, safety mechanisms should be in place to prevent a stack overflow. The swap function correctly swaps nodes A and B. It accounts for the case when nodes A and B are adjacent.

insert performs insertion after the parameter node.

pushFront and pushBack do exactly what they would be expected to do.

min finds the minimum value in a list where the fist valid node is head.

removeAfter removes the node after the parameter one and returns the parameter node.

The functions follow the guidelines set out in the assignment instructions.

All the functions have been tested and work as expected. A check for memory leaks returned no leaks.

This is a project and should be compiled as a project.

# Running the Application

Running it is simple. Basic validation has been added to the main to prevent a crash or a stack overflow.