ENSF 338 - L02

Final Project - Module 1 and 2

2023-04-14

Findlay Brown & Grace Jang

Distribution of Work

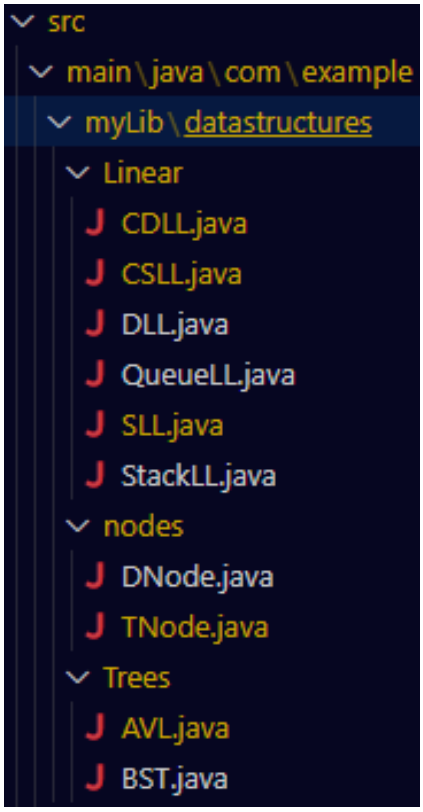
Name	UCID	Work
Findlay Brown	30145677	- AVL - BST - Testing file - Reviewed Module 1
Grace Jang	30142405	- SLL - DLL - CSLL - CDLL - Stack - QueueLL - Reviewed Module 2

Github link:

https://github.com/graceejangg/ENSF338-Final-Project-Group-35

Library Design Overview:

This is a general view of our library



In the library, it is broken down to Linear, Nodes, and Trees. In Linear, there are a total of 6 files. The 6 files are Single Linked List (SLL), Double Linked List (DLL), Singly Circular Linked List (CSLL), Double Circular Linked List (DCLL), Stack (StackLL), Queue (QueueLL). In each file there is a common theme of public functions. These include: InsertHead(node), InsertTail(node), Insert(node,position), SortedInsert(node), isSorted(), Search(node), DeleteHead(), DeleteTail(), Delete(node), Sort(), Clear(), Print(). All nodes in the linear are using DNode from the node section of the library. The linear sections are all somewhat interconnected, where it extends from one another to make a more concise file. For example, DLL will extend from SLL and etc.

In nodes, the file DNode the functions: getNext(), setNext(next), getPrevious(),setPrevious(previous), getData(), setData(int data), toString() are set up and specified.

Both TNode and DNode are listed in the nodes section of the library.

Both AVL and BST are listed in the Trees section of the library.

The file BST.java includes 3 constructors, BST(), BST(int data), and BST(TNode root).

BST.java also includes the following public methods: getRoot(), setRoot(TNode root), insert(int data), insert(TNode node), delete(int val), printInOrder(), printInOrder(TNode current), printBF(), search(int val).
BST also includes the following private methods: minimumValue(TNode node) and delete(TNode node, int val)

The file AVL.java includes 3 constructors, AVL(), AVL(int data), and AVL(TNode newNode).

AVL.java also includes the following public methods: insert(int data), insert(TNode node), and delete(int val)

AVL also includes the following private/helper methods: getBalance(TNode node), rightRotate(TNode node), leftRotate(TNode node), insert(TNode node, int data), height(TNode node), getBalance(TNode node), insert(TNode root, TNode node), rebalance(TNode node)

Bonus Element Attempted:

- Maven component attempted
- AVL Balancing constructor
- AVL delete with balancing updates
- Packaging assignment

Instructions/Requirements to Run the Code:
There's no specific requirement to run the code, but a note we had to note when building the code was to remind ourselves to import the right files so the code could run. However, to actually run the code, there is no specific instruction or requirements to run the code.

To include the correct libraries and modules: use
Include com.example