Project 1 – Parking Garage Double Linked List

Due: October 16, 2022 11:59 pm **Total point value: 100**

Goals

The goal of this project is to get some mild experience with Sets and Bags while using Double Linked Lists as the data structure. By the end of the project, students will have created a very simple Parking Garage System.

Implementation Requirements

You **must have** the CarDataNode, GarageSet, and GarageExitBag classes.

Following is a description of **some** specific implementation requirements:

1. CarDataNode
    a. CarDataNode is a doubly linked list node
    b. It must implement addNodeAfter and toString in a fashion similar to the Book's ADT/Class Demo Code
        i. removeNode removes the calling node from the list!
    c. checkIn is used to set the Date checkIn to the current time
    d. checkOut is used to set the Date checkOut to the current time
    e. equals is only concerned with checking if the license plates between two nodes are the same
    f. Two static methods are specified
        i. Finding a node with a given license plate – takes two arguments, a node to start searching at and the string
        ii. Getting a node with a given index – takes two arguments, a node to start searching at and the index of the node to return

2. GarageSet
    a. Must at least have head. You may choose to add a tail
    b. Should be able to add Nodes (checkIn) and remove nodes (checkOut, using both a String and an index)
        i. checkIn and checkOut are also responsible for ensuring the node's checkIn/checkOut field update appropriately!
        ii. Duplicates are not allowed in a set!
    c. GarageSet's toString is to be used when displaying the GarageSet on the garage display frame
    d. GarageSet is responsible for saving itself/loading itself upon program start using the two static methods loadGSData() and saveGSData()\

3. GarageExitBag
    a. Should only have a head, not a tail

   b. Nodes that are removed from the GarageSet should be added to the GarageExitBag
   c. The method, dumpOutputFile, should print the contents of the exit bag to a text file named using the current date.
4. General details
   a. When the application exits using the Exit option in the File menu, GarageSet should be saved and the GarageExitBag should be dumped

Reaching the rest of the functionality requirements is up to you.

Coding Style

You will be graded on following proper OOP principles and basic coding style. No bad variable names etc.

Comments

Each class and method should have a Javadoc (/** */) style comment, explaining what the class/method does. @param should be used to describe each argument, while @return should be used to describe return types.

Submit: Upload a zip of your .java to brightspace. YOUR CODE MUST COMPILE TO BE GRADED.

Evaluation:

Requirements	Pts	Comment
Correctness	**80/80**	
Code Compiles	25/25	
CarDataNode	20/20	
GarageSet	20/20	
GarageExitBag	15/15	
Design	**10/10**	
Documentation and Style	**10/10**	
Bonus: GUI is nice	**10/0**	**Grader discretion**