Timothy Hoang

2/24/19

CMPT435-111

Assignment 4

**Date Assigned: 02/18/2019**

**Due: Midnight 02/24/2019 on iLearn**

**Please read turn-in checklist at the end of this document before you start doing exercises.**

**Section 1: Pen-and-paper Exercises**

1. In this course, we have studied the following data structures: an unsorted array, linked list, stack, queue, and hashtable. For each of the following applications, indicate which of these data structures would be most suitable and give a brief justification for your choice.
2. Customers at the airport check-in counter

Note:

(1 point) data structure.

Queue

(1 point) justification for your choice.

FIFO when you are in an airport line you expect the person at the front of the line to get serviced first or else it would not be fair.

1. A list of items, random access.

Unsorted array because you get random array items form the list.

1. A list of items, output items in opposite order.

Stack because LIFO, the items come out opposite of the way they came in.

1. A list of items, constant time search and update.

Hash table list because key-pairs allow for everything to have a constant time.

**Section 2: Java Implementation**

1. Create a linked list and remove the minimum element in the list in Java.

Note:

Find a file called LinkedList.java in assignment 1 folder.

Complete the method of Removemin().

Test your method in the main method provided following the comments.

**Important: In all of the assignments of this course, when you are asked to implement an algorithm for a problem, your code will be evaluated based on:**

**5 points - Execution**

**Each file must run without error or warning on valid input described in the main method provided.**

**5 points - Within Code Documentation**

**Is the code documented for obvious understanding of the use, preconditions, and postconditions of each function?**

**20 points - Correctness**

**Is the algorithm implemented correctly? Does your method pass the test?**

1. Given an access log, create a hashtable using the given SeparateChainingHashTable class to count the number of times an ip address occurs in the access log.

**Note:**

**Find the file named AccessLog.txt, item.java, and SeparateChainingHashTable.java in assignment 4 folder.**

**Complete the main method.**

**Test the main method following the comments.**

**Full credit (30 points) will be awarded for an algorithm that uses SeparateChainingHashTable. Algorithms not using hashtable will be scored out of 10 points.**

**TURN-IN CHECKLIST:**

1. **Answers to Section 1 (.doc/.txt/.pdf), and to Section 2 (all your source Code (.java files)). Remember to include your name, the date, and the course number in comments near the beginning of your code/report.**
2. **Create a folder and name it 'FirstName\_LastName\_assignment\_4'. In the newly created folder copy and paste your files (.doc/.txt/.java files). Then compress the folder, and push it to iLearn.**