Assignment 4

COMP 250 Fall 2017

posted:	Wed., Nov. 22, 2017
due:	Tues., Dec. 5, 2017 at 23:59

The Teaching Assistants handling this assignment are Tabish Syed (email) and Jack Guo (email). Their office hours will be posted on mycourses Announcements.

The general instructions are the same as in previous assignments, in particular, 20 points late penalty per day up to 3 days late and we will deduct 20 points for any student who has to resubmit after the due date (i.e. late) because the wrong file or file format was submitted. This policy will hold regardless of whether the student can provide proof that the assignment was indeed done on time. See the submission instructions at the end of this PDF.

Hash Table

In this Assignment you will implement a hash table as an arraylist of buckets. Each bucket of the hash table contains a singly linked list of nodes containing key-value pairs.

The starter code consists of five classes:

- `MyHashTable` which is a simplified and reduced version of `HashMap`
- `HashNode` which stores a key-value pair (i.e. hash table entry); you may add helper methods to this class if you wish. For simplicity of editing, this is a separate java file rather than an inner class of `MyHashTable`.
- `HashLinkedList` which is a linked list of `HashNode` objects with code stubs.
- `Song` that describes song objects which will be stored in the hash table
- `HashTableTester` class which uses a `MyHashTable` to store a list of songs. The key is the song title and the `Song` object is value. Thus, the key happens also to be one of the fields of value object.

You may add additional tests in the HashTableTester class, but you must **not** modify the `Song` class.

Your Task

(30 points)

Implement HashLinkedList, which is a simple version of the singly linked list class for the buckets of the hash table. You can start from the linked list implementation provided on the course web site (Exercises 3) as a guide. In particular, implement the methods

- add(K,V), remove(K), removeFirst(), getListNode(K).

Note that you will need to modify the given singly linked list code to use it in your hash table. In particular, the HashLinkedList<K,V> class has two generic types K,V instead of one generic type E in SLinkedList<E> class.

(70 points)

Implement the following methods in the MyHashTable class:

- the constructor MyHashTable()
- put(K,V), get(K), remove(K), rehash(), keys(), values()
- the constructor for the inner class HashIterator.

For the specification of what these methods must do, see the comments in the code.

You may wish to add helper methods to the HashNode class.

Submission instructions

Submit a single zipped folder A4.zip which contains your java files MyHashTable.java, HashLinkedList.java and HashNode.java.

Good luck and have fun!