CS 272 Homework 7 - Basic Data Structures

In Google Drive, go to **File->Make a Copy** and save your own copy of this document as: **h05_LastName1_FirstName1.** You will work on your own copy, since you will not be able to edit this shared master copy. Complete each of the homework exercises below using Eclipse.

You should begin with the starter code. In has all the files you need to get started and complete this assignment.

H07-A—Extending the Singly-Linked List Class

Find the completed implementation of singly-linked lists in the starter code. We will add a few methods to the class LinkedList.

Add methods the following methods to the LinkedList class:

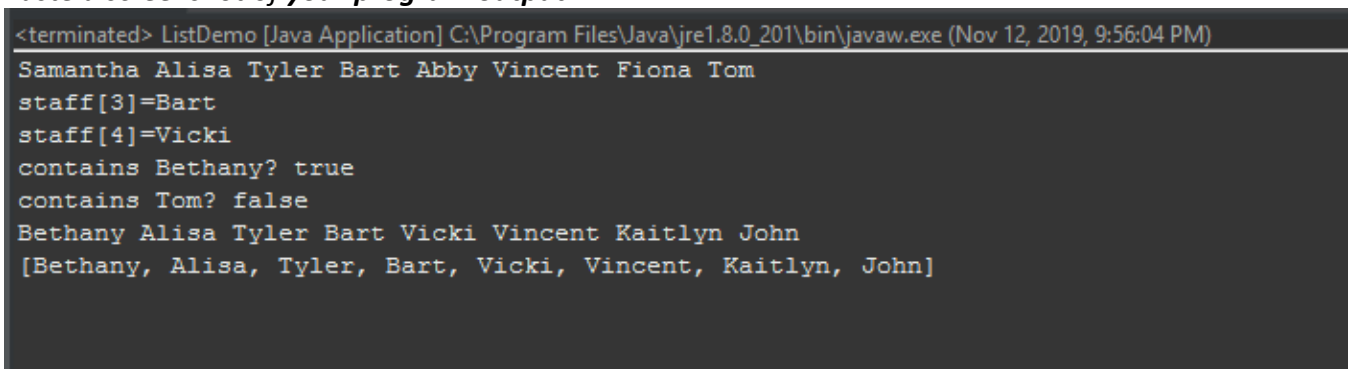
- `Object get(int n)`
  - Returns the object data stored at the node reached by following n links.
- `void set(int n, Object newElement)`
  - Sets the data to `newElement` at the node reached by following n links.
- `public String toString()`
  - Overrides the inherited `Object.toString` method for the LinkedList.
  - You will need to use the `toString` method of the node data.
- `public boolean contains(Object obj)`
  - Checks whether the list contains a given object.
  - Use the `equals` method to determine whether `obj` equals `node.data` for a given node.

Hint: For `get` and `set`, use a helper method that starts at first and follows n links:

- `private Node getNode(int n)`

Another hint: Implemented these methods by directly traversing the links, not by using an iterator. Simply grab the reference to first (Node current = first;) and use current.next() to walk to the end (null).

*Paste a screenshot of **your program output**.*



H07-B—Hash Collisions Stats

In this assignment we experiment with the `hashCode` method that is inherited from class `Object` each time an object is created. The method returns an `int` that is used by some Java data structures such as `HashSet` for determining the position of an object in the structure. Every object has a unique hash code that can take on the full range of integer values. Imagine determining the position of an object in an array by transforming the object's hash code into an index that is within the bounds of the array. Suppose we have an array with room for 5749 (a nice prime number) elements. Here is one way to convert the `int`-valued hash code of an object to an integer in the range 0 to 5748:

1) If the number is negative, negate it, so that it is positive.

2) Mod the number by 5749. The remainder fits the bill for an index into the array.

The Stanford Graph Base contains a list of the 5757 five-letter words in English. Here is a link: http://www-cs-faculty.stanford.edu/~uno/sgb-words.txt. Download the list and save it as a file on your computer. (This is already included in the starter code Eclipse project.) Write a `main` method that creates a `Scanner` that reads each word in the file. Because each word is read as a `String` object, every word has a hash code that can be retrieved by calling the `hashCode` method. The goal in this assignment is to compute where each word in the list would be stored in an array of length 5749. We will also compute a few statistics, so you will need to build an integer array of length 5749, named `x`, to keep up with the positions where words are hashed. Complete the following steps for each word in the file:

1) Retrieve the hash code of the word.
2) Compute the index, `i`, where the word will be placed in the array.
3) Add 1 to `x[i]` to keep up with the number of times a word hashes to location `i`.

When you have read the file completely, compute and print the following statistics:

1) The number of empty positions (buckets) in the array `x`.
2) The maximum number stored in a single array location (the longest bucket chain).
3) The average of all non-zero array entries (the average bucket chain length).

*Paste a screenshot of **your program output** with those three statistics here.*

```
The number of empty buckets is 2075
The longest chain in a bucket is 6
The average length of a chain is 1.5669569951007076
```

After you have completed the assignment, there are two kinds of deliverables:

1. Make a **zip** file of the entire eclipse project and upload the zip to Canvas.
2. Save this document as a **pdf** and upload the pdf to Canvas.