**Computer Science III HashTable: WordHash Lab Value - 100**

**Lab Goal :** This lab was designed to give you experience managing a hash table.

**Files Needed ::**

**UnorderedWords.txt**

**WordHashTester.java**

**Lab Description :** The file UnorderedWords.txt contains almost 10,000 words that are not in any particular order. The words will be stored in an array based hash table, using a linked list of WordNodes to contain any Words that hash to the same index

Use the Word class that you wrote for the Heap lab. Add an equals method. Also, you will override the hashCode method that is in the Object class. It returns an int. The trick of writing a hash table is the hashCode method. Your method needs to return an index into an appropriately sized array to store the desired data, minimizing collisions (hashing to the same index) as much as possible.

Write a WordNode class so that your hash table can store an array of WordNodes. Each array element in WordHash will be a linked list of WordNodes, so a WordNode needs to have a Word and a reference to the next Word with appropriate constructor, setter an getters.

Write a WordHash class that has an array of WordNodes as a field. The constructor should receive the size of the array. Additional methods are:

**add:** adds a Word using its hashCode() to get its index. If the array already has a WordNode there, it should add to the front of the list (fastest place to add to a linked list). It is void.

**remove:** removes the Word from the hash table and returns a boolean. Use the hashCode() to find it’s index, then determine if it is in the chain and remove it.

**find:** returns the index at which the specified Word was found, or -1. Again, do NOT search the entire array. Use the index returned by hashCode().

toString: We need to be able to tell if the indexes that are occupied are spread out evenly enough to be efficient. The toString should return a list of each word in the table labeled by index. We really don’t care how many vowels each Word has, I just didn’t change that part. You can change the Word toString if you like.

Example:

1 rustless: 2

1 automatically: 6

3 postseason: 4

3 fortissimo: 4

4 spoliators: 4

4 housewifely: 5

8 destroyers: 3

9 prettification: 6

10 misstatement: 4

10 haika: 3

14 jiujutsus: 4