**SUMMER 2012 CS 401 (hoffman) LAB #6**

**Handin 1 file named Lab6.java**

**STARTER CODE:**[**Lab6.java**](http://www.cs.pitt.edu/~hoffmant/401/lab-06/Lab6.java)**INPUT FILE:**[**input.txt**](http://www.cs.pitt.edu/~hoffmant/401/lab-06/input.txt)

**how**

**much**

**wood**

**would**

**a**

**wood**

**chuck**

**chuck**

**if**

**a**

**wood**

**chuck**

**would**

**chuck**

**wood**

**EXPECTED OUTPUT**

Your solution SORTS the entries by the word. Can be accomplished by pulling the keys (words) out into an array, sort the array then loop thru the sorted keys and use each word in a .get(word) to produce the freq

**C:\> java Lab6 input.txt**

**a 2**

**chuck 4**

**how 1**

**if 1**

**much 1**

**wood 4**

**would 2**

**Use this code as your starting point. It is from the Java tutorial online. We looked at it this week when we introduced HashMap**

**import java.util.\*;**

**public class Lab6**

**{**

**public static void main(String[] args)**

**{**

**HashMap<String, Integer> histogram = new HashMap<String, Integer>();**

**// Initialize frequency table from command line - YOU MODIFY TO READ FROM A FILE**

**for (String w : args)**

**{**

**Integer freq = histogram.get(w);**

**histogram.put(w, (freq == null) ? 1 : freq + 1);**

**}**

**System.out.println(histogram); // CHANGE THIS TO EXTRACT THE WORDS INTO AN ARRAY AND SORT THEN PRINT THEM**

**}**

**}**