Ethan Dunham
Assignment 6
CS 261

1. Give an example of two words that would hash to the same value using hashFunction1 but would not using hashFunction2.

"eat" and "ate" would hash to the same value using hashFunction1, but not hasfunction2. This is due to the fact that they contain the same letters, but function 1 does not shift the value based on it's location.

2. Why does the above observation make hashFunction2 superior to hashFunction1?

It cuts down on the number of same hashed words. In function1, all words containing the same letters would be hashed together, but with function 2, they would be spread out. This allows faster execution time because the item will be located at the index instead of a link farther into the location.

3. When you run your program on the same input file once with hashFunction1 and once with hashFunction2, is it possible for your hashMapSize function to return different values?

The size will remain the same since it will just add them to another bucket. There will still be 2 links/ size of 2.

4. When you run your program on the same input file once with hashFunction1 and once with hashFunction2, is it possible for your hashMapTableLoad function to return different values?

The tableload would remain the same. The size and capacity will both be the same for each function, so the load would not change.

5. When you run your program on the same input file once with hashFunction1 and once with hashFunction2, is it possible for your hashMapEmptyBuckets function to return different values?

It is possible to have a different number of empty buckets for each function. Since anagrams will be placed in separate buckets, the number of empty buckets could be different, depending on other keys.

6. Is there any difference in the number of empty buckets when you change the table size from an even number like 1000 to a prime like 997?

Prime numbers work the best for hash tables because they cause less collision, due to the rules of math.