Khoa Phan
CS261
Assignment 5 – Answers

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

An example would be the words 'tea' and 'eat' as they both end up with the same values when using hashFunction1 due to it only taking the char's values but not manipulating it. hashFunction2 does, which creates a different value in since it does account for the order in which the letters are.

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

Since it does account for the ordering of the letters, it creates a unique value in the hash table within the same bucket, thus creating an efficient O(1) complexity within the structure.

**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?**

No, it is not possible because if the input files are identical, then they will contain the same words and ordering, which would result in the same number of hash links created.

**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 hashMapTableLoad function will not return different values. If the same input file is used, the function finds the same ratio between the numbers of links and total number of buckets. It does not track any of the empty buckets and tables will resize exactly the same.

**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?**

hashMapEmptyBuckets can return different values because different hashFunctions will contain in a different number of collisions.

**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?**

There is a different in the number of empty buckets because the intial hash table size is set to 1000 opposed to the hash table size that is set to a prime like 997. The capacity is used as the modulo to determine the placement of the bucket, so the prime number capacity decreases the chances of collision only because it has two common factors while composite numbers can have more.