**Coding Assignment 6 Results**

CSE 3318

**Test 1**

**A.** How many rows are in your file/how many cells are in your hash table array?

30 rows/30 cells

**B.** How many of those cells contained the head of a linked list?

20 cells

**C.** What percentage of the array is being used?

66%

**D.** What is the length of the longest linked list?

Longest linked list has 3, and there are two that have 3

**Test 2**

**A.** Did increasing the size of the hash table array give you different results than Test Question 1?

Yes.

**B.** Explain why or why not.

40% of the array was used and the longest linked list contained two elements and there were 6 of them

**Test 3**

**A.** How many rows are in your file/how many cells are in your hash table array?

30 rows/15 cells

**B.** How many of those cells contained the head of a linked list?

14 cells

**C.** What percentage of the array is being used?

93%

**D.** How did decreasing the size of the hash table array affect the percentage of the array that filled?

Decreasing the size made the percentage go up, in this case it was nearly filled all the way with only one cell open.

**E.** Did your hash table get any linked lists that were longer than in Test 1? Why or why not?

Yes, there is one linked list that contains 4 elements. This is because the hash table is smaller. When testing with 10 cells, there was a linked list with 5 elements.

**Test 4**

**A.** What was your average search time when your HASHTABLESIZE matched the number of records in the file?

14.7 tics

**Test 5**

**A.** What was your average search time when your HASHTABLESIZE was set to 1?

14.7 tics again

**Test 6**

**A.** What was your average search time when your HASHTABLESIZE was set to 1 and you only searched for the last record of your input file?

15.3 tics

**B.** Was this average different from your answer to Test 5. If yes, why?

The answer was different but not that different. Even if we are searching for the last record, hashing would make it so it doesn’t matter if it’s the last record, it matters where it is on the hash table.

**Bonus Question**

If your program was using Open Addressing rather than Separate Chaining, then how many cells of the hash table array would be used when HASHTABLESIZE is set to the number of lines in the file? How did you calculate this number? Show/explain your answer.

If we used Open Addressing, we would use 30 cells. This is because open addressing will put the element in next available cell if there is already an element in the cell the hash value says its supposed to go. Since I have 30 lines in my input file, I would need at least 30 lines to hold my entire file.