**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?

*Number of rows in the file=30*

*Therefore, number of cells in the HASHTABLEARRAY=30*

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

*The cells that contained the head of a linked list is 18*

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

*(18/30)(100)=%60 A total of sixty percent.*

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

*The length of the longest linked list is 3.*

*There are a total of 3 linked lists that have this Favorite Band Database=[7][13][14]*

**Test 2**

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

*Note: Changed HASHTABLESIZE from 30 to 60*

*It did make a difference in the results. So yes increasing the size of the has table array gave me different results.*

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

I noticed that the hash table/index if we change the sum of the alphabet by the amount of the hash table array size. When the amount of hash table array size changes, the inputs are then stored in different areas of My Favorite Band Databases.

**Test 3**

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

*//Note: changed HASHTABLESIZE from 30 to 15*

*Original HASHTABLESIZE & the inputfile was 30. Then I changed the HASHTABLESIZE to 15.*

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

12

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

*(12/30)\*100=%40 A total of 40 percent.*

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

*Decreasing the* size of the hash table array affect the percentage of the array that filled. Because there was way less space it caused an increase in the number of values in the array or per array.

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

*My hash table was very short for test 3. So, yes because I decrease the HASHTABLESIZE making it have less space.*

**Test 4**

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

*//NOTE: changed HASHTABLESIZE back to 30*

*Tove Lo, Ashnikko, Call Me Karizma, Doja Cat, Fall Out Boy, Kerli, Linkin Park, MOTHICA, NF, Shakira*

*2+3+5+2+3+4+4+3+5+4= 32…35/10= 3.5*

**Test 5**

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

//NOTE: changed HASHTABLESIZE back to

Tove Lo, Ashnikko, Call Me Karizma, Doja Cat, Fall Out Boy, Kerli, Linkin Park, MOTHICA, NF, Shakira

*3+2+2+2+2+3+3+3+3+3=26…26/10=2.6*

**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?

*29/10=2.9*

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

*It’s different because the program has do traversing/traversal to the linked list at the end of the program.*

**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.

Open Addressing is a little faster than separate chaining. Open Addressing is faster because instead of putting a new element in the array it has to put it in the next spot that it’s open. Let’s just say that if you put HASHTABLESIZE as 1. I’m pretty sure that Open addressing would have to relocate/allocate the array to fit all the chars/elements. Whereas separate chaining takes longer but it doesn’t have allocate because it links all the chars together, and performs a linear search essentially.

|  |  |
| --- | --- |
| Open Addressing | |
| 1 | |
| What’s left: | Rammstein|Bad Bunny|Tove Lo|Dorothy|Allie X|Ashnikko|Atreyu|Bishop Briggs|Call Me Karizma|Charlie Puth|Christina Aguilera|Doja Cat|Dua Lipa|Eminem|Fall Out Boy|Iggy Azalea|Ivy Levan|Kerli|Korn|Lizzo|Linkin Park|Meghan Trainor|Melanie Martinez|MOTHICA|Nelly Furtado|Shinedown|NF|Qveen Herby|STARSET|Shakira| |
| 1 element, then needs to allocate. | |

|  |  |
| --- | --- |
| Separate Chaining | |
| 1 | |
| What’s left: | Nothing |
| 1 element/array: |Rammstein|->|Bad Bunny|->|Tove Lo|->|Dorothy|->|Allie X|->|Ashnikko|->|Atreyu|->|Bishop Briggs|->|Call Me Karizma|->|Charlie Puth|->|Christina Aguilera|->|Doja Cat|->|Dua Lipa|->|Eminem|->|Fall Out Boy|->|Iggy Azalea|->|Ivy Levan|->|Kerli|->|Korn|->|Lizzo|->|Linkin Park|->|Meghan Trainor|->|Melanie Martinez|->|MOTHICA|->|Nelly Furtado|->|Shinedown|->|NF|->|Qveen|-> |Herby|->|STARSET|->|Shakira| | |