**Project Report 6**

**Problem Statement:**

The primary goal of this homework assignment is to gain familiarity with a hash table collision management technique called separate chaining. The secondary goals are to create classes for a student node, the student list, and the hash table. Utilizing these classes, several linked lists will be generated inside of each index of the hash table. The tertiary goal is to remember how to implement linked lists. The program inputs are a data set of five attributes and various integers that command the program to accomplish some tasks like update, remove, and search. The outputs are arrays of nodes at certain indexes.

**Design:**

Not many aspects of the design process were left up to creator’s discretion. Three classes needed to be made, two of which were required for a linked list to function properly and the other to manage the hash table. Data structures used include linked lists and arrays. Arrays were used for the hash table indexes and the linked lists were used by the separate chaining technique to manage collisions. The hash key function was the only algorithm used in the program. That function converted the key (full name) to a number which could be placed at a certain index among the hash table. As these data structures and algorithms were required for this assignment to function accordingly, there are no disadvantages to implementing the way I did.

**Implementation:**

First, I decided to start programming each class. I started with the Student Node class which encompassed many parallels with how the Restaurant Node class was constructed. The main difference was the Student Node class was singly linked whereas the Restaurant Node class was doubly linked. Then came the Student List class which was not dissimilar to the Restaurant List class. After that was coding the Hash Table class. Finishing this class came with ease as it utilizes the methods from the Student List class. My development timeline was three days amounting to about six hours of work.

**Testing:**

The main objective of this assignment centered around testing different hash functions and analyzing their efficacy in relation to number of collisions. Normal inputs include the integers that correspond to the menu navigation and appropriate data type input for each attribute of each student. Special cases tested include inputting none integers in the menu. My hash function was num + = 19 \* key[x] + 7. This resulted in 14 collisions in my typescript. The hash example given in the homework document num += num \* 17 + key[i] & size resulted in 13 collisions. This means my hash functions was marginally worse than that provided in the project document. Most things worked as expected.

**Conclusions:**

The assignment resulted in a fully functioning hash table organized by separate chaining. The project was a success as most ever aspect of the program worked in accordance with the Homework 6 rubric. I would understand more about how hash tables work and try to internalize the syntax and functions of linked lists. This would have made the programming process much smoother and quicker. The project took six hours to complete.