**CIS 263 – Assignment 6**

By: Jack Lukomski

**Part 1:** Given input {4371, 1323, 6173, 4199, 4344, 9679, 1989} and a hash function:

Show the resulting Hash Table. Note: the hash table will start at 0 and end at 9 for a size of 10.

1. Separate chaining hash table:

A screen shot of a computer

Description automatically generated

1. Hash table using linear probing:

A screen shot of a computer

Description automatically generated

1. Hash table using quadratic probing:

A screen shot of a computer

Description automatically generated

1. Hash table with second hash function h2(𝑥) = 7 − (𝑥 𝑚𝑜𝑑 7)

A screen shot of a computer

Description automatically generated

As you can see, the 1989 is not in the table. This is because it is impossible in this case for the equation to find an unused index in the table for 1989. I put the equation in desmos and all the outputs of this equation, with , result in an output where that index is already occupied in the table. (You can see this simulation in the submission).

**Part 2:** Write a program to compute the number of collisions required in a long random sequence of insertions using linear probing, quadratic probing and double hashing. For simplicity, use only integers for your long random sequence. The simulation should continue until the first 500 numbers from the random sequence have been inserted. Hash function h(x) = x % D where D is the size of the table (fixed size of 1001). Second hash function: h2(𝑥) = 7 − (𝑥 𝑚𝑜𝑑 7)

This output shows the number of collisions for inserting 500 random numbers into a hash table that is size 1001. The collision methods used are linear probing, quadratic probing, and double hashing. This is performed n times, and the number of collisions is averaged together (n is 10000 in this case).

A screen shot of a computer

Description automatically generated