**Data Structures and Algorithms II**

**Fall 2020**

Student name: Colin Hwang

Student e-mail: colin.hwang@cooper.edu

Assignment: Program #1

Grade: 95

Comments:

The program works for all my tests.

Although it did not affect these test cases, I see a bug in the insert member function of the hash table. Your check for rehashing is in between the linear probing and the actual insertion. In the case where you rehash, you'll be putting the new item in the wrong place. You need to move that check either to the very start (before the call to the hash function) of very end (after the insertion) of the inert member function. -5 points

Also, no points off (because these things weren't required for the first program), but your findPos and insert routines are not handling lazily deleted items correctly. Insert should not return 1 if it detects a lazily deleted version of the key, and it is typical to probe past such an item too (stopping is OK, but you need to be careful). The main loop in findPos typically only checks for occupied slots (deleted slots should still be considered occupied), and only return if a non-deleted key matches. Also, rehash will need to reinsert void pointers along with keys.