**Pseudocode Week 4**

**Tammy Hartline**

**03/26/2023**

**CS-300**

DEFAULT constructor

INITIALIZE node that holds bid

RESIZE HashTable

DESTRUCTOR

IMPROVE storage

IF class is destroyed

REMOVE beginning nodes

CALCULATE hash value for each key

DIVIDE key by the table size

RETURN remainder of key value divided by the table size

FINISH INSERT bid

CREATE key with hash(stoi) to FIND node

IF key IS NOT found

THIS node is the index of key

ELSE IF key is unused

FIND last index

WHILE this node is pointing to next, AND next is NOT nullptr

SET node EQUAL to node pointed to next

PRINT ALL BIDS

SET Bid AS bid

FOR node at start

ITERATE until FINAL bid

IF key is NOT EQUAL to UINT\_MAX

WHILE node is NOT EQUAL to null pointer

PRINT key, bid id, title, amount and fund

SET node EQUAL to node points to next

REMOVE bid

SET unsigned key EQUAL to hash(atoi) with bid id

DELETE beginning node and its key

SEARCH bid

CREATE key for the users entry

SET node EQUAL to new node bid and key

IF key IS found

RETURN bid and bid found message

IF node OR key is NOT found

RETURN node points to bid NONE and bid not found message

WHILE node does NOT point to null pointer

IF this node matches compared node

RETURN node bid

END IF

SET node EQUAL to node pointed to next

END WHILE

RETURN bid

**Week 4 Code Reflection**

I found this week’s assignment to be quite a bit more challenging than the previous weeks. The purpose of this week’s code was to create a program that can access, read, and write to a file, containing bid information. The end user should be able to go into the program and immediately be given a menu display option, that then prompts them to choose one of the options. If they select 1, it should load the bid file and output time taken to load. If 2 is selected, it should print a list of all the bids inside that file, with the bids corresponding information. If 3 is selected, the user should be prompted to search for a specific bid id, and then notified whether the bid was located or does not exist yet. If 4 is selected, the 1st bid in the file should be deleted from the file. If 9 is entered, the user sees a “Good-bye” message, and the program is exited.

The main challenge faced this week was with the PrintAll() section of the code. While the instructions were clear, or seemed to be clear, I was unable to print all of them to the screen. I was able to get it to print however many file lines are in the file, but all data showed as blank or a 0. I tried several approaches to correct this, including using (it) iterator, the displayBids() option, and cout. However, I was met with either returning to the menu prompt, or all empty and zero values returning. I am not certain how to fix this, even after researching for several hours. A quick thank you for all your help this week, with this and last week’s assignment, as well as the assignment in my other course.