Dylan Dunagan

1 June 2025

CS-300 DSA: Analysis and Design

Module 4 Assignment Hash Table Reflection

The purpose of this code is to create a hash table to hold bids from a given CSV file. First the code creates the framework of the hash table and defines parts of it, such as bids, nodes, and keys. Then different functions are created to allow for resizing (nodes.resize) and erasing nodes (nodes.erase). The “Insert” function is also made to allow for inserting bids into the middle of the hash table utilizing if/else statements and loops to ensure the program searches for the specific spot for the bid placement. The next functions allow for outputting (PrintAll) the data within the hash table, looping through each node to provide a full output. The “Remove” function allows the user to remove specific bids from specific node by searching through the hash table for the key that is associated to a given node. The “Search” function is how the user would search for a specific bid within the hash table. Next is the “displayBid” function that allows for the output of a specific bid and the “loadBids” function loads the CSV file into the program. Lastly the code then moves into the “main” function where it brings all the functions together into a working program. The biggest problem I had was overthinking. The actual code needed for each “FIXME” section was a lot simpler than I was expecting. So, after realizing that I could use much less code to get the same result, I was able to make my way through the code. I realized that in “FIXME (4)”, line 160, it asks me to “assign old node key to UNIT\_MAX” but throughout the rest of the code, the correct spelling is “UINT\_MAX”. Because “UNIT\_MAX” is no where else in the code, I ensured that line 162 of the code that I wrote said “UINT\_MAX” to avoid any undefined objects.

The pseudocode for the “main” function is:  
Main()

Open CSV file

While user input does not equal “9”

Output menu

Case 1

Start the timer

Load bids from CSV file into hash table

Output time elapsed

Case 2

Call PrintAll function to print the entire hash table

Case 3

Start the time

Use “search” function on a bid key

If bidID is not empty

Output the bid

Else

Output that the bid ID was not found

Output time elapsed

Case 4

Use “remove” function to remove a bid

Output “Good Bye”

End program