Willem Kroeger

CS 300: Data Structures and Algorithms: Analysis and Design

Southern New Hampshire University

January 28, 2023

Writing this code for the hashtable came easier to me than the linked list project last week. For me, visualizing a hashtable and how it all comes together is easier than a linked list, especially when using pointers to jump around a linked list. The hashtable works on all functions, but I had to lookup several different operations to understand what was happening, especially with the character into integer transformations that are part of the hash value.

Using the UINT\_MAX as a “blank” value, I was able to tell if a key value had been placed there or not. If not, a new key value was created. If a key already existed, a new item was appended to the list. From there, the list is searched, items removed, or the list traversed to print all items.

**Structure to hold bids** **& initialize structure**resize tableSize

**Free Storage**erase node in beginning

**Generate hash value**  
key = key % tableSize

**Insert a bid**  
develop key with bidID

Address node that matches key value

If node key != UINT\_MAX (default value)  
 create new node  
 insert new node to beginning of list

Else

If node key == UINT\_MAX (default value, list is empty)

Set key, bid, and next values

Else

While node->next != nullptr  
 increment node->next->next

**Print Bids**

While traversing node lists

If node key != UINT\_MAX (default value)  
 Print node data  
 while next node != nullptr  
 print node data  
 increment node to next node

**Search Bids**

Discover key value

Set node address to key value

If node != nullptr AND key != UINT\_MAX (default values)  
 return node bid

If node == nullptr AND key == UINT\_MAX (default values)  
 return blank bid  
  
while search node != nullptr  
 if node bidID = search bidID  
 return search bid  
   
 increment search node to next value

Return blank bid

**Remove bids**

Discover key value

Erase beginning of node and key that matches key value