When given the linked list for this assignment it felt a bit easier than last weeks assignment and I felt that I understood this assignment better and was able to understand more of how we would use this type of storage for variables and how it is able to make it a better experience of working with the given variables. I feel that the linked lists offer a good storage for all manner of variables and allow a faster way of storing or removing those variables.

**Pseudocode for the linked list Assignment**

\_Make a default constructor and make the head and tail equal to nullptr  
\_in the next deconstructor create a new node and start at the head of the list  
While current is not Null  
\_check the nodes and update the current node to the next node  
\_Remove the unneeded Nodes  
  
**The Append Function**  
\_Create a new node  
If the head is equal to Null  
\_make head and tail the new Node  
else  
\_make the current tail the new node and increase the size

**The prepend function with parameters bid**  
\_create a new node  
checks if the head doesn’t equal null  
then set the head to the next node so it will become the new node  
then increase the size amount  
**Create the PrintList Function**  
\_create a node that will start at the head  
While the current doesn’t equal null  
\_Prints the current bidID, title, amount, and fund  
\_set the current variable to next  
  
Create the remove function with the parameters of string bidId  
If head bidId is equal to bidId  
\_make the variable head point to the next node in the list  
\_decrease the size  
then return  
\_start at the head again

While current doesn’t equal the null  
if the next node bidID is equal to the current bidID  
\_create a temp node  
\_create a current node that points to the next node  
\_delete temp node  
\_decrease the size  
\_return  
set the current node to equal the next node

**Create the Search Function**\_Create a variable that starts at the head  
While current -> next  
if current matches   
return  
Else  
set the current bid to the next node  
return  
  
**Display Menu for the user**start the timer  
LinkedList bidList;  
Bid bid;  
int choice = 0;  
While loop that only ends if the user chooses option 9  
switch function for the menu options  
-if user input is 1 :Enter a Bid  
 \_Get the bid from the input   
 \_Append and display the Bid  
-if user input is 2: Load Bids  
 \_Print the clock, opens and reads the document  
-If user input is 3: Display All Bids  
 \_Access print list  
-If user enters 4: Find Bids  
 \_display clock  
 \_If bid is empty   
 display the message “bid ID: bidkeyvariable Not found”  
-If user input is 5: Remove Bids  
 \_Remove the bidKey  
-If user input is 9: exit  
display the message “Good Bye”

End the program