**Module 3 – Linked List Reflection Assignment**

This assignment focused on implementing a linked list in C++ to manage auction bids. The goal was to load bids from a CSV file, store them in a linked list, and allow users to search, remove, and display bids. I used a singly linked list structure with Node and LinkedList classes, along with methods like Append, Prepend, PrintList, Remove, Search, and Size. One challenge I encountered was to handle the edge cases like empty list or null pointer. I overcame these by carefully reading the pseudocode instructions and reading text book to make sure the logic is correct.

**Pseudocode for append:**

//Create new node

//if there is nothing at the head...

// new node becomes the head and the tail

//else

// make current tail node point to the new node

// and tail becomes the new node

//increase size count

**Pseudocode for prepend:**

// Create new node

// if there is already something at the head...

// new node points to current head as its next node

// head now becomes the new node

//increase size count

**Pseudocode for PrintList:**

// start at the head

// while loop over each node looking for a match

//output current bidID, title, amount and fund

//set current equal to next

**Pseudocode for Remove:**

// special case if matching node is the head

// make head point to the next node in the list

//decrease size count

//return

// start at the head

// while loop over each node looking for a match

// if the next node bidID is equal to the current bidID

// hold onto the next node temporarily

// make current node point beyond the next node

// now free up memory held by temp

// decrease size count

//return

// current node is equal to next node

**Pseudocode for Search:**

// special case if matching bid is the head

// start at the head of the list

// keep searching until end reached with while loop (current != nullptr)

// if the current node matches, return current bid

// else current node is equal to next node

//(the next two statements will only execute if search item is not found)

//create new empty bid

//return empty bid

**Pseudocode for Size:**

// Returns the current size (number of elements) in the list