Linked List Pseudocode

Forward declarations (string str, char ch);

Define a structure to hold bid info;

Struct Bid {

string bidId;

string title;

string fund;

double amount;

Set amount = 0.0;

Define a class for data members and methods to implement linked-list;

Internal structure for list entries and housekeeping variables;

Default constructor Node () {

Next = null ptr;

}

Initialize with a bid;

Initialize housekeeping variables;

Set head and tail to null

while (current != nullptr) {

temp = current;

current = current->next;

delete temp;

Append new bid to end of list;

void LinkedList::Prepend(Bid bid) {

Node\* node = new Node(bid);

if (head != nullptr) {

node->next = head;

}

head = node;

Implement prepend login & create new node;

Output bids in the list to console;

while (current != nullptr) {

Print current->bid.bidId ":" current->bid.title "|" current->bid.amount "|"

current->bid.fund;

current = current->next;

bidId to remove from list;

void LinkedList::Remove(string bidId) {

if (head != nullptr) {

if (head->bid.bidId.compare(bidId) == 0) {

Node\* tempNode = head->next;

delete head;

head = tempNode;

}

Implement bidId to search;

Bid LinkedList::Search(string bidId) {

Node\* current = head;

while (current != nullptr) {

if (current->bid.bidId.compare(bidId) == 0) {

return current->bid;

}

current = current->next;

Display bid info;

displayBid(Bid bid) {

Print (bid.bidId ": " bid.title " | " bid.amount; " | "bid.fund;

return;

Prompt user for bid information;

Return bid struct containing bid info;

Load CSV file containing bids into linked list;

Read rows of CSV file;

for (int i = 0; i < file.rowCount(); i++);

int main(int argc, char\* argv[]) {

string csvPath, bidKey;

int choice = 0;

DISPLAY Menu to console

while (choice != 9) {

DISPLAY << "Menu:"

DISPLAY << " 1. Enter a Bid";

DISPLAY << " 2. Load Bids";

DISPLAY << " 3. Display All Bids" ;

DISPLAY << " 4. Find Bid";

DISPLAY << " 5. Remove Bid ;

DISPLAY << " 9. Exit";

DISPLAY << "Enter choice: ";

cin >> choice;

Reflection

We had to extract bid information from a CSV file for this program. We practiced using singly linked lists and used methods like Remove, Append, Prepend, and searching through a linked list. I struggled to get the CSV file to load correctly. I didn’t have any issues with the menu and console, but when trying to Load Bids (option 4) and Search Bids, I was getting an error. I was dereferencing a null pointer, and it was throwing an exception error. That was my biggest issue with this assignment. I utilized a few different resources when trying to solve this issue. Stack Overflow was a huge help, as well as Reddit. I finally reached out to my professor, asked for additional support, and correctly finished the program to meet the program's requirements and functionality.