Gentian Hoxha

[Gentian.hoxha@snhu.edu](mailto:Gentian.hoxha@snhu.edu)

3/28/2024

The code as the assignment was implemented to use algorithms in a selection sort and quicksort of vector bids loaded from the CSV file. Even though I had some problems when running the towards the end with the Unhandled exception error even though If exceptions were used in csv;Parser. The assignment also provides a simple text-based menu to interact with the user, demonstrating basic input/output (I/O) and program flow control. We also initialized clock\_t to measure the execution time of operations, providing an insight into performance analysis.

A simplified pseudocode:

Start Of Program

Initialize csvPath with a default CSV file path

If command-line argument is provided:

Update csvPath with the argument value

Initialize an empty vector of Bids named bids

Initialize choice variable

Display Menu:

1. Load Bids

2. Display All Bids

3. Selection Sort All Bids

4. Quick Sort All Bids

9. Exit

While choice is not 9:

Get choice from user

If choice is 1:

Start timer

Call loadBids function with csvPath

Store returned bids in bids vector

Stop timer and display time taken

If choice is 2:

Iterate over bids vector

Call displayBid function for each bid

If choice is 3:

Start timer

Call selectionSort function with bids vector

Stop timer and display time taken

If choice is 4:

Start timer

Call quickSort function with bids vector, start index 0, and end index size-1

Stop timer and display time taken

If choice is unrecognized:

Display error message

End While

Program End

Function loadBids(csvPath):

Display loading message

Initialize an empty vector of Bids named bids

Try

Initialize csv::Parser with csvPath

For each row in csv file:

Initialize a Bid object

Assign values from CSV to Bid fields

Add Bid to bids vector

Catch any csv::Error exception

Display error message

Return bids vector

Function displayBid(bid):

Display bidId, title, amount, and fund of the bid