Data Structures and Algorithms Artifact

Daniel Little

CS-499 Computer Science Capstone

Professor Brooke

04/24/2024

Briefly describe the artifact. What is it? When was it created?

The artifact I chose for this category is part of my CS-260 Data Structures and Algorithms course. I created this artifact a year and a half ago. This artifact uploads bids from a CSV file and sorts them using sorting algorithms upon user request.

Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in software development? How was the artifact improved?

The implementation of algorithms in this code before enhancements is why I believe it would be a great inclusion for this category. I was able to build new functionality where the user could sort by date and bid ID instead of solely using selectionSort and quickSort functions.

By adding new functionality to this code I showed my ability to work with data structures and algorithms to create new sort functions. I also showcased my ability to take user input more effectively by re-using getBid to allow the user to add bids by inputting the proper attributes.

Did you meet the course objectives you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?

I met these course outcomes:

Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts.

By creating add a bid and creating sort by date and bidID I was able to design, develop, and deliver quality communications. I was able to do code review during the process of adding this functionality to ensure that it was all technically sound.

Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals (software engineering/design/database).

Adding the sort functionality to allow the user to have more sorting options meets industry-specific goals because it allows for statistical research within companies and businesses.

Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision making in the field of computer science.

With each update I was able to push the newest change to a git repository which is one of the greatest environments for team building. On top of this I made sure my code was always commented thoroughly.

Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while managing the trade-offs involved in design choices (data structures and algorithms).

By designing “add a bid”, “sort by closing date”, and the sort by bidID functions I was able to show my ability to design and evaluate computing solutions that solve a given problem using algorithmic principles. In each naming convention I was able to consider computer science practices and standards to make sure the design choices were appropriate.

Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?

One of the difficulties faced was when sorting by different parameters we had to change what was defined to be less than in the comparator. With the date time in particular I was getting a few errors when trying to sort, converting the strings to integers so I could get a numerical representation. But the solution ended up being using date-time objects. Another complication that I ran into was implementing sorting algorithms that were efficient in time and showed a good time complexity. It looked like “quicksort” had a good result.