4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure

Tyler Cornell

Southern New Hampshire University

4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure

## Category Two Narrative – Algorithms and Data Structure

The artifact selected is an invoicing system for Davy’s auto shop services that was originally created in March of 2019 in the CS-200 Computer Science Role in Industry course at SNHU. As previously stated, the artifact was selected because it was in the beginning of my computer science related courses and was one of the first experiences I had with coding. It is a great candidate for selection because it allows me to implement improvements from multiple course outcomes, satisfying multiple project requirements. Many improvements were made to the artifact during the category one narrative period that touches upon not only software design and engineering but algorithms and data structure as well. Those previous changes include naming conventions for dictionary keys and values and taking user input and converting it to lowercase when searching the dictionary to prevent case sensitivity problems. The addition of the ASCII art banner included data manipulation to accept the code as raw so that it would not be interpreted as other character functions commonly used in Python. During this category’s enhancement I decided to utilize another ASCII art banner for when printing the total cost to the user. This is another case of forcing the interpreter to use the code as raw to interpret the ASCII art correctly. The if else statements were combined with a try except clause to further establish a functioning solution to the user that interprets the input correctly, no matter the input. It can determine if the user has selected a valid service with a cost, no service with zero cost, or an invalid service and ask them to reselect. This handling enhancement has allowed for a seamless user experience with a higher level of data interpretation. The if statement that checks the service selected to see if it is greater than zero and prints the service and its respective price has been changed for readability. The clause now utilizes an f-string for manipulation that makes the print statement much easier to read for anyone reviewing the code. The artifact was overall improved by implementing new features that are interpreted correctly, changing a print statement into an f-string formatted print, changing naming conventions to explain data better, manipulating input for usability, exception handling for key errors, and verification of order to if else statements.

During the enhancement process, I learned some best practices and other forms of data manipulation that increase readability and change the interpreter’s interpretation of the code. An example of a best practice learned is as having if else statements search for the most common result first in order to improve efficiency. I learned how to use f-strings to improve the readability of rather busy statements to clean up the code and manipulate the data in a different way than I originally had known. Learning on the job and always advancing one’s knowledge is a key concept that employers seek. “Adaptable learners are in high demand. The world’s leading CEOs are publicly asserting the importance of learning and skills in a constantly evolving global economy” (Zao-Sanders & Schveninger, 2020). The course outcome of designing and evaluating 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 has been satisfied with the enhancement plan.

References

Zao-Sanders, M., &amp; Schveninger, C. (2020, March 27). *The simple joy of learning on the job*. Harvard Business Review. Retrieved March 21, 2023, from https://hbr.org/2020/03/the-simple-joy-of-learning-on-the-job