**CS 499 Milestone Three**

**Brian Chmura**

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

The artifact I’ve chosen to enhance is a *Hash Table Algorithm* from my Data Structures and Algorithms course (CS-260), which I completed more than two years ago. A hash table is a data structure that stores information using a key-value system, where each data value is assigned a unique index within an array, making data retrieval more efficient.

**2. 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 algorithms and data structure? How was the artifact improved?**

I selected this artifact for my ePortfolio because it aligns with key concepts from CS-260, including algorithmic design and the evaluation of complex data structures. This artifact showcases my ability to work with advanced data structures and demonstrates my understanding of algorithm efficiency. I improved the artifact by rewriting the original C++ code into Python, which involved adjusting syntax and resolving errors, thereby enhancing both the functionality and readability of the code.

**3. 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?**

Yes, I believe I met the course objectives, particularly CS-499-03 and CS-499-04. During the enhancement process, I employed development best practices, such as writing pseudocode to map out complex logic and performing incremental debugging to isolate and resolve errors. I plan to update my outcome-coverage plan to include testing and debugging strategies, as these were critical in delivering a working Python implementation of the hash table.

**4. 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?**

Throughout the process of enhancing the hash table, I revisited many foundational concepts in data structures and algorithms, which reinforced the importance of logical design and problem-solving. Converting the code from C++ to Python presented challenges, particularly with syntax differences and debugging, but it also allowed me to refine my understanding of both languages. Debugging was the most time-consuming aspect, but it ultimately strengthened the result.