* Briefly describe the artifact. What is it? When was it created?
  + This artifact is a final project I created for my CS300 (Data Structures) course at SNHU. It was published to GitHub on December 16, 2023, and demonstrates the insertion and retrieval of data from a HashTable. The project showcases foundational skills in data structures, such as efficient data storage and lookup, which are essential for software development.
* 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?
  + I included this artifact in my ePortfolio because it effectively demonstrates my skills in Software Design and Engineering, Algorithms and Data Structures, and Databases. The artifact allowed me to showcase my ability to implement and test a Red-Black Tree data structure, focusing on compliance with its core properties. For example, I designed a deletion function to remove a node from the Red-Black Tree, as well as test cases to ensure its functionality.
* Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?
  + I have successfully met these two outcomes:
    - 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.
    - 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.
  + At this time, I have no updates to my outcome-coverage plans, as the current enhancement aligns well with the intended learning goals.
* 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?
  + This time, because I was working on existing code, I took notes of the enhancements and struggles I worked with. This documentation will help if I encounter the same problems in the future. The documentation is also great for self-esteem.
  + Improved RedBlackTree constructors using delegation and variable initialization for cleaner, more modular code
  + I was having a problem with the RedBlackTree module destructor being called twice on deletion. However, this only occurred in the test cases, and not in the Main() function. The problem was that I had a function that would create and return a RedBlackTree. To fix this, I converted the function to return std::unique\_ptr<RedBlackTree>.
  + Implemented DeleteNode, which deletes a node from the Red-Black Tree
  + Moved StringToKey to its own .hpp and .cpp files
  + New Tests:
    - TestDeletion: test DeleteNode() successfully deletes the node
    - TestDeletionBlackHeight: Ensures a consistent number of black nodes, from root to leaves, after deletion
    - TestDeletionConsecutiveRedNodes: Ensures no consecutive nodes are red after deleting a node.