Edward Garcia

CS499-17487-Computer Science Capstone

4-2 Milestone Three: Enhancement 2: Algorithms & Data Structures Narrative

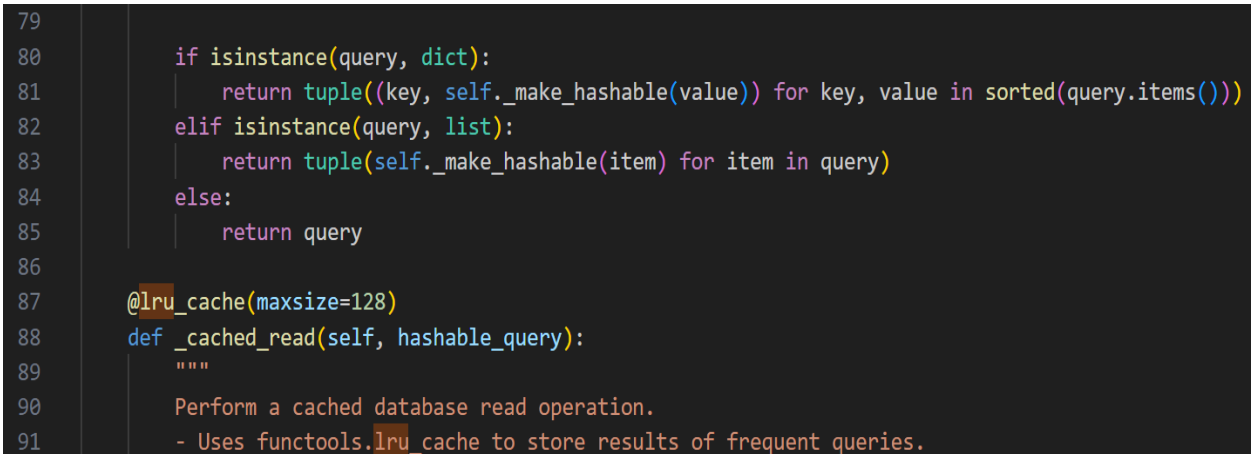
Professor Krupa

November 21, 2024

Enhancement 2: Algorithms & Data Structures Narrative

The artifact I selected for Enhancement 2 is an optimized version of my CS340 Client/Server Development *Animal Shelter Project*, which is a MongoDB-based tool designed to manage animal shelter data. This project demonstrates my ability to address challenges in database management through efficient algorithm design and implementation. For enhancement two, I integrated advanced data structures and algorithms to improve system performance and scalability. My specific improvements included adding an LRU caching mechanism for faster repeated queries, implementing a hash map for breed-based lookups to achieve O(1) time complexity, and utilizing binary search for sorted attributes like age and animal type to achieve O(log n) search efficiency. I also used testing techniques within the Jupyter Notebook to validate these enhancements, and show the results of their impact.

I selected this artifact for my ePortfolio because it highlights my technical skills in optimizing database interactions and implementing advanced algorithms. By implementing LRU caching, it reduces response time for frequently accessed data. The hash map and binary search improve the efficiency of data retrieval operations. Below is a screenshot of how the LRU cache works in my code:



```
79
80          if isinstance(query, dict):
81              return tuple((key, self._make_hashable(value)) for key, value in sorted(query.items()))
82          elif isinstance(query, list):
83              return tuple(self._make_hashable(item) for item in query)
84          else:
85              return query
86
87      @lru_cache(maxsize=128)
88      def _cached_read(self, hashable_query):
89          """
90          Perform a cached database read operation.
91          - Uses functools.lru_cache to store results of frequent queries.
```

These enhancements are supported by testing methods I designed within the notebook, such as a Hash Map Lookup Time Complexity Test, which compares hash map performance to direct database queries, and a Cache Performance Test, which benchmarks query speeds with and without caching. The test results indicate that using the LRU cache significantly reduces query time, as shown in the screenshot below by the faster response time (0.004052 seconds) when using the cache compared to querying the database directly (0.011664 seconds).

```
# Cache Performance Test

# This test compares the query performance with and without the LRU cache implementation.

import time

# Query for performance testing
query = {"breed": "American Bulldog"}

# Without cache
shelter.clear_cache()  # Clear cache
start_time = time.time()
result_no_cache = shelter.read(query, bypass_cache=True)
end_time = time.time()
print(f"Query without cache: {end_time - start_time:.6f} seconds, Results: {len(result_no_cache)}")

# With cache
start_time = time.time()
result_with_cache = shelter.read(query)
end_time = time.time()
print(f"Query with cache: {end_time - start_time:.6f} seconds, Results: {len(result_with_cache)}")
```

```
Query without cache: 0.011664 seconds, Results: 4
Query with cache: 0.004052 seconds, Results: 4
```

These enhancements align strongly with several of the course outcomes. They associate with Outcome 3, as they demonstrate my ability to design and evaluate computing solutions that address scalability and efficiency challenges using algorithmic principles. By implementing Python's functools.lru_cache, hash maps, and binary search, I highlighted my proficiency in creating effective solutions while managing the trade-offs between memory usage and speed. The structured and professional documentation of my enhancements and testing results aligns with Outcome 2, as it demonstrates my ability to deliver clear, and technically sound written communications tailored to diverse audiences. My use of caching and data structures aligns with

Outcome 4, showcasing my ability to use well-founded tools and techniques to implement scalable and efficient computing solutions that deliver value. These enhancements also align with Outcome 5, as they highlight a security mindset by implementing data validation and error-handling mechanisms to prevent vulnerabilities and solidify the integrity of my application.

The process of enhancing and modifying this artifact for me was both challenging and rewarding. Designing the LRU caching mechanism required some understanding of how caching impacts memory usage, to make sure it was compatible with real-time database queries. Implementing the hash map for breed-based lookups involved me balancing memory overhead with the need for efficient data access. The binary search required maintaining a sorted structure for relevant attributes.

Some challenges I faced included monitoring synchronization between the hash map and live database operations and validating the accuracy of binary search results for sorted attributes like animal types. Another challenge I faced was creating tests to benchmark these enhancements without altering the database. In one of my test cases for verifying the LRU cache's performance, I created a script with a test breed class so that it does not interfere with the breeds already in the animal database. Due to an error in my code, my entire animal database was overwritten by the three test breeds I created. I learned a valuable lesson, that even though testing is beneficial it can also be a disaster if tests are not implemented correctly.

My chosen artifact and these enhancements represent my ability to design scalable and efficient applications by leveraging advanced algorithms, optimized data structures, and modern development practices. Through this work, I have demonstrated not only my technical proficiency but also my ability to critically analyze and address performance bottlenecks in real-world systems. Including these enhancements in my ePortfolio will help highlight my growth as

a software engineer, my problem-solving abilities, and my commitment to delivering high-quality solutions that are both effective and adaptable.