CS 3303-01 Data Structures

Instructor: Professor Vandelun Amokaha

Name: Ryohei Hayashi

Learning Journal 7

**Summary of the Week**

This week, I mainly focused on understanding the differences between primary storage and secondary storage, and further deepened my understanding of disk drive architecture and terminology. I was able to clearly comprehend the characteristics of RAM memory and disk drives, as well as the distinction between volatile and persistent memory. Additionally, I learned about the cost associated with disk access, particularly the head movement and seek time. I also worked on implementing buffer cache heuristics (such as FIFO, LFU, LRU) and explored external sorting algorithms.

**Personal Reflections**

Although I had previously studied the differences between storage types, applying these concepts to the implementation of algorithms was a new and insightful experience. In particular, understanding the different approaches to cache management (FIFO, LFU, LRU) and reflecting on their respective use cases helped me gain a deeper appreciation of their practicality and efficiency. I realized how critical buffer pool implementation is to the overall performance of a system. The knowledge gained through this week’s discussions was highly beneficial, and I felt that I could grasp the concepts more concretely, especially in practical programming.

**Topics Studied in Depth**

This week, I particularly focused on the implementation and management techniques of buffer caches. Buffer caches serve as a temporary storage space for data, significantly improving system performance. I explored different heuristics for cache management, such as FIFO (First In First Out), LFU (Least Frequently Used), and LRU (Least Recently Used), and learned how each of these approaches differs in practical scenarios.

FIFO, while straightforward, can lead to performance degradation under certain conditions. On the other hand, LFU is effective in keeping frequently used data in the cache by prioritizing the removal of less frequently accessed data. LRU, which removes the least recently used data, is particularly useful for managing cache based on recent access patterns and proves to be highly effective in real-world applications. By implementing and testing these techniques, I gained a deeper understanding of their strengths and weaknesses.

Moreover, in the area of external sorting, I worked on implementing both quicksort and mergesort, gaining insights into how sorting algorithms function efficiently in large-scale data processing. The replacement selection algorithm, in particular, stood out as a key technique for external sorting, demonstrating how it can work effectively even when the data exceeds the memory capacity.

**Future Challenges and Goals for Next Week**

Next week, I plan to focus on search and indexing algorithms, delving deeper into topics such as jump search, dictionary search, and secondary binary search. I also aim to understand the characteristics of self-organizing lists and study hash algorithms (open hashing, closed hashing, etc.), which play a crucial role in efficient data access in databases and file systems.

Additionally, to achieve satisfying results in the final test, I plan to dedicate significant time to the Review-Quiz. By reviewing everything I've learned so far and addressing any areas where my understanding is lacking, I hope to approach the final test with confidence.

Word Count: 510

References

1. Shaffer, C. (2011). *A Practical Introduction to Data Structures and Algorithm Analysis.* Blacksburg: Virginia. Tech.