

## **My Journey**

During this homework, I learned how different string-matching algorithms behave under many scenarios, and how difficult it is to design a strategy that consistently picks the fastest one. I also realized that even if you test many different patterns and texts, there is always some error margin because small changes in structure can completely change which algorithm performs best. Writing Boyer-Moore and the algorithm-selection logic was actually quite easy, but I couldn't achieve the level of accuracy I expected. Some of the provided test cases also behaved differently from theory. For example, in the "KMP Advantage" and "Best Case for Boyer-Moore" tests, the Naive algorithm ended up being the fastest, which was surprising and made it harder to build a reliable strategy. Through analyzing my results and reviewing general algorithm principles, I learned when each algorithm usually works well for example, Naive for very short patterns, KMP for repetitive prefixes, Boyer-Moore for long texts with larger alphabets, and Rabin-Karp for long or highly repetitive patterns. Overall, the homework was interesting, but I feel it could be improved with more consistent and realistic test cases. Even though I didn't reach perfect accuracy, I still learned a lot about practical algorithm behavior and the challenges of designing heuristic-based selection rules.

23050111078

Ali Burak Karaat