

For this project, I used the Boyer-Moore algorithm's bad-character rule. First, I created a lookup table (bmBc) that tracks where each character appears last in the pattern. Then I compared the pattern to the text moving backwards (right to left). When I found a complete match, I saved the location. If the characters don't match, I used my table to calculate how far to slide the pattern along. For calculation I used ascii table but there was an unsolved case so I changed it to unicode. I repeated this until I run out of text, and finally, I returned a list of all the match locations as a single string.

For the preanalysis I used your hasRepeatingPrefix method then I implemented some basic if-else cases. If the pattern is short, I just used the Naive algorithm because it's quick. If the pattern repeats itself at the start, I used KMP, which handles that structure really well. For very long texts and patterns, I switched to Rabin-Karp to take advantage of hashing. For everything else it runs with Boyer-Moore since it's usually the fastest in practice.

My Boyer-Moore implementation first solved 29 out of 30 questions. After I changed the bad character analysis it completed all the questions. Including the preanalysis table, Boyer-Moore is slower than other algorithms for most of the cases. Boyer-Moore needs Bad character table because of this it runs slower than others. Naive algorithm is the chosen one for the most of the cases. Naive algorithm directly starts to research when the other algorithm make tables and calculations so they come with the delays. For repeating case KMP shows better performance and RobinKarp never been first choice.

Firstly string matching is hard to do for me. I saw this in our lab exam and midterm.
Secondly boyer-moore algorithm is harder than other matching algorithm(in my opinion).
I try to understand algorithm and try to code but I couldn't. After that I want to pseudocode in Gemini.

I code using with this code. I learned we use ASCII or Unicode for badcharacter method and it changes our efficiency.

Finally this is my Gemini chat for this HW: <https://gemini.google.com/share/f66a6aaa6a2f>

Özgür Atlı
23050151029