Project #5: String Matching Algorithms

This project consists of the following three parts.

**Part 1: Comparing the Runtimes**

For this part of the project you are required to implement the following string matching algorithms and compare their running time with different inputs.

1. The Knuth-Morris-Pratt Algorithm
2. The Boyer-Moore algorithm
3. Rabin-Karp Algorithm

For this part of the project you are going to be given two files: one that includes the patterns to be searched and another that includes the text. You will then search each pattern and take the exact running time for each search and plot your results for all three algorithms.

**Part 2: Bit-Oriented Approach vs AhoChorasick Algorithm**

For this part of the project you are required to implement the bit vector string matching algorithm and the AhoChorasick Algorithm. For this part you are going to use the same test files and execute a similar analysis to part one. To test your algorithm, you are asked to find all occurrences of each pattern from the patterns.txt file in the given text.

**Part 3: Suffix Tree**

For this part, you are going to implement the Suffix Tree data structure and search for the patterns. You are also required to take the running time for your string match and compare with the algorithms in part 1. Remember that you need to preprocess the text to construct the suffix tree, which is why we do not prefer including this in part 1.

Test files will be posted on blackboard.

This project is due Tuesday May 2, which is the last day of classes.

Late submissions will not be accepted for this project.