Question 1 Write a single C program with the following functions.

- Write a function to take two files and write the differences between the two files to another file. The difference should only account for the words and order of the words. It should not consider punctuation or spacing.
- Implement two functions for the Columnar Transposition Cipher, one to encrypt and one to decipher. This should be all done on files.
- Write a function in C to remove a file from the disk. I expect this function to be verbose in regard to telling the user if the removal was successful and if not, why.
- Write a function that will take a .c file and verify that any and all pairs of brackets, quotes, parentheses, *etc.* are matched and nested correctly.
- Write a function that translates a file from English to Pig Latin or visa versa.

  Pig Latin is a made-up children's language that's intended to be confusing. It obeys a few simple rules (below), but when it's spoken quickly it's really difficult for non-children (and non-native speakers) to understand.

  Rule 1: If a word begins with a vowel sound, add an "ay" sound to the end of the word. Please note that "xr" and "yt" at the beginning of a word make vowel sounds (e.g. "xray" -> "xrayay", "yttria" -> "yttriaay").

  Rule 2: If a word begins with a consonant sound, move it to the end of the word and then add an "ay" sound to the end of the word. Consonant sounds can be made up of multiple consonants, a.k.a. a consonant cluster (e.g. "chair" -> "airchay").

  Rule 3: If a word starts with a consonant sound followed by "qu", move it to the end of the word, and then add an "ay" sound to the end of the word (e.g. "square" -> "aresquay").

  Rule 4: If a word contains a "y" after a consonant cluster or as the second letter in a two letter word it makes a vowel sound (e.g. "rhythm" -> "ythmrhay", "my" -> "ymay").