**Compiler Design (Lab 3)**

**Problem:**

1. Write a Lex program that copies a file, replacing each non-empty sequence of white space by a single blank.

2. Write a Lex program that copies a C program, replacing each instance of the keyword float by double.

3. Assume a file is a sequence of words (groups of letters) separated by whitespace. Write a Lex program that converts this file in the following manner. Every time a word is encountered:

a) If the first letter is a consonant, move it to the end of the word and then add ay.

b) If the first letter is a vowel, just add ay to the end of the word.

4. Write a Lex program that counts characters, words, and lines in a text file and prints the counts. Assume a word to be any sequence of letters and/or digits, without punctuation or spaces. Punctuation and white space are not counted as words.

**Input-Output:**

1. The program requires a read.txt file as an input and provides write.txt as an output with the desired changes.

2. The program requires a read.c file as an input and provides write.c as an output with the desired changes.

3. The program requires a read.txt file as an input and provides write.txt as an output with the desired changes.

4. The program requires a read.txt file as an input and displays the character count, word count and the line count on the console.