Lab 5:

Points: 20

Goals: The purpose of this lab is

1. to practice use of indefinite loops (WHILE loops)
2. to gain confidence using string processing methods
3. To increase familiarity with using Scanner object for input from console

Directions:

Write a modular menu driven program that process a given line of text. The user is given the choice of counting the number of words in the line, replacing acronyms for words, searching for a word in the line or exiting the program.

The program should work regardless of leading and trailing spaces in the line and multiple spaces between words. It should also work for an empty line or line with only spaces.

The set of acronyms available for substitution are “TTYL”, “IDK” and “IMO”. They are to be replaced with “talk to you later”, “I don’t know” and “in my opinion”. If an acronym other than the above are included in the line, then no replacement is to be done.

The word search may return the first occurrence of the word if found in the line. . If word is not found in the search, the message “word not found” should be displayed.

If an invalid option is entered, the menu should be repeated with any additional message. If an acronym other than the above are included in the line, then no replacement is to be done.

Program Output:

You program must display the following sequence of sample interactions. Test your program thoroughly for various cases. For example, no words, a single word, empty line, leading and trailing spaces, mutliple spaces between words. I will grade your program according to the tests that it can pass.

```
Enter a sample text:
You entered:
MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:r
Edited text:

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:f
Enter a word or phrase to be found:
dd
"dd"not found

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:w
Number of words: 0

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:
```

```
Enter a sample text:Hello, IMO Its a great day
You entered: Hello, IMO Its a great day
MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:w
Number of words: 6

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:f
Enter a word or phrase to be found:
great
"great"found at position 17

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:r
Edited text: Hello, in my opinion Its a great day

MENU
w - Number of words
f - Find text
r - Replace acronyms
q - Quit

Choose an option:q
```

Program Design:

Your program should be designed in a modular way

Your program should consist of two files: *Main.java* and *LineProcessor.java*

The methods file *LineProcessor.java* contains the class called *LineProcessor* which defines the following methods in the flow of execution given:

```
public static void run()  // declares Scanner and inputs line
   public static void printMenu()
   public static void process(String text, Scanner scnr, char x)
       public static int getNumOfWords(String text)
       public static String replaceAcronyms(String text)
       public static int findText(String s, String text)
```

The file *Main.java* contains the driver class called *Main* and calls the method *run().*This method inputs the line from the user and calls *printMenu()* to print the four menu options and then calls the *process* function to process the line as per the user selection. The user is repeatedly offered menu options until he/she quits.

Documentation: Make sure your source file has a header comment box and line comments for each method call and method definition. Use meaningful names for variables and methods where applicable and indent your program using the recommended style.