*HW\_3*

*Write your code in the file Compress.java. Your code should go into a method with the following signature. You may write your own main method to test your code. The graders will ignore your main method:  
  
public static String compress (String original){}*

*Run-length encoding (RLE) is a simple "compression algorithm" (an algorithm which takes a block of data and reduces its size, producing a block that contains the same information in less space). It works by replacing repetitive sequences of identical data items with short "tokens" that represent entire sequences. Applying RLE to a string involves finding sequences in the string where the same character repeats. Each such sequence should be replaced by a "token" consisting of:*

1. *the number of characters in the sequence*
2. *the repeating character*

*If a character does not repeat, it should be left alone.*

*For example, consider the following string:*

*qwwwwwwwwweeeeerrtyyyyyqqqqwEErTTT*

*After applying the RLE algorithm, this string is converted into:*

*q9w5e2rt5y4qw2Er3T*

*In the compressed string, "9w" represents a sequence of 9 consecutive lowercase "w" characters. "5e" represents 5 consecutive lowercase "e" characters, etc.*

*Write a method called compress that takes a string as input, compresses it using RLE, and returns the compressed string. Case matters - uppercase and lowercase characters should be considered distinct.****You may assume that there are no digit characters in the input string. There are no other restrictions on the input - it may contain spaces or punctuation.****There is no need to treat non-letter characters any differently from letters.*

*Write your code in the file PigLatin.java. Your code should go into a method with the following signature. You may write your own main method to test your code. The graders will ignore your main method:  
  
public static String translate (String original){}*

*"Pig Latin" is a fake language used as a children's game. A word in English is "translated" into Pig Latin using the following rules:*

* *If the English word begins with a consonant, move the consonant to the end of the word and add "ay". The letter Y should be considered a consonant.*
* *If the English word begins with a vowel (A, E, I, O, or U), simply add "way" to the end of the word.*

*(This is a simplified dialect of Pig Latin, of course.)*

*Write your method so that it returns the pig latin translated original string.****You may assume that the input does not contain digits, punctuation, or spaces.****The input may be in any combination of uppercase or lowercase. The case of your output does not matter.*

*Write your code in the file WordCount.java. Your code should go into a method with the following signature. You may write your own main method to test your code. The graders will ignore your main method:  
  
public static int countWords(String original, int minLength){}*

*Your method should count the number of words in the sentence that****meet or exceed****minLength (in letters). For example, if the minimum length given is 4, your program should only count words that are at least 4 letters long.*

*Words will be separated by one or more spaces. Non-letter characters (spaces, punctuation, digits, etc.) may be present, but should not count towards the length of words.*

*Hint: write a method that counts the number of letters (and ignores punctuation) in a string that holds a single word without spaces. In your countWords method, break the input string up into words and send each one to your method.*