Comp 333 Project 3 (35 pts)

**Due April 10 by 9am. Upload to Canvas.**

1. (25 pts) In this problem you will write a wordcount Scheme function, called wc, that inputs a string of characters and outputs the number of characters, words and lines in the string. The format of the function is ( wc str). For example ,

>(wc "A quick cat jumped \n over the fence \n and left." )

'(47 9 3)

Note that the \n is a newline and counts as one character. Each space counts as one character. Words are separated by whitespace ( newline, space or tab) and do not contain any whitespace characters. A word may include punctuation. For example “left.” and “one\_two” are single words.

To solve this problem, write functions a-f below. *You must write your functions from scratch, using only Scheme functions discussed in class or text.* You may assume each of lst arguments below are lists of characters.

* 1. (countChars lst) - returns number of characters in lst
  2. (countLines lst) - returns number of lines in lst ( 1 + number newline chars)
  3. (skipWS lst) - removes all white space from beginning of list. That is it returns a new list that looks like lst, with all white space remove before the beginning of the first word.
  4. (removeNextWord lst) - removes the first word in the lst. That is it returns a new list that looks like lst without the beginning white space and without the first word.
  5. (countWords lst) - returns the number of words in lst
  6. (wc str) - converts the string str to a list of characters and uses a,b, e to return a list of the counts of characters, words and lines in str.

1. (10 pts) Solve the following problems without recursion. Use map , filter, foldl and anonymous functions instead of recursion. *You must write your functions from scratch, using only Scheme functions discussed in class or text.*
   1. Write a Scheme function, ( stringhashcode str), that returns the ASCII sum of the characters in a string. For example, (stringhashcode “Algorithms”) returns 1050.

Use the built-in *char->integer* function to convert a character to its ASCII number.

* 1. Write a Scheme function ( allStrings? lst) that checks if all elements of lst are strings. You may assume that lst is a list.
  2. Write a Scheme function ( hashlist lst) that returns the list of the hashcodes of each of the strings in lst. If lst is not a list of strings return “ERROR”. For example,

>(hashlist '("Algorithms" "functional programming+" "A History of Ballet and Dance" ))

'(1050 2337 2570)

1. Upload one file called Project3.rkt to Canvas. This file must include in this order:
   1. Your full name, course and project number at the top
   2. All required Scheme functions (and any extra helper functions you created) for Problems 1 and 2. If any one of your Scheme functions do not work, have it just return an error message “NonWorking Function”. Organize and label your work in the file.
   3. Include my InstructorsTestCase function in your file. InstructorsTestCase.rkt is posted on Canvas.
   4. Include your run results of the IntructorsTestCase function. Comment out the results with semicolons.