ALGONQUIN COLLEGE

Computer Engineering Technology – Computing Science

Course: Numerical Computing – CST8233
Term: Fall 2021

Assignment #1

Due Date: Monday, October 4th, 2021 before 11:59 pm.
Earning: This assignment worth 5% of your final course mark.

1. Part I (15 marks)

The objectives of this part is to learn how to: 1) read in ".csv" files using **read.csv()** function, 2) use different arguments of this function, and 3) extract some data and write them to a new ".csv" file.

Task:

You are given a file called "**assignment1.csv**" that includes some information about different cars models, such as MPG, Cylinders, Horsepower, etc. Write a script using R language that does the following:

A. (2 marks) Read "**assignment1.csv**" file and save it as a dataframe called "**myCarsList**". Show the first six rows of **myCarsList**.

   Hint: search for **read.csv()** function in R Documentation to learn how to use the arguments of this function. Specifically, the "sep =" argument.

B. (2 marks) Examine the file and notice that the second line represents the data type of each column. You need to delete this row entirely from the list. Then, print the number of columns and rows.

C. (3 marks) Examine the class of each column and change it as required. For example, MPG column is of type *char*, you need to change it as *numeric*. The Origin column must be of type *factor*.

D. Find the car model that has:

   a. (1 mark) Maximum MPG
   b. (2 marks) Horsepower greater than 100
   c. (2 marks) Acceleration less than 15

For items b and c above, extract the results to two files and name them as “**Horsepower.csv**” and “**Acceleration.csv**”, respectively.

E. (3 marks) Using the main dataframe, plot the histogram of the MPG using 10 bins. Note that there are some zero values in this column. You need to ignore these values when creating the histogram plot. Set the title of the graph as: “MPG Histogram” and the color of the graph as Red, and the x-axis as MPG.

   Hint: search for **hist()** function in R Documentation to see how to set the arguments.

2. Part II (10 marks)

The objectives of this part is to get familiar on how to deal with vectors and calculate functions.

Task:

Write a script using R language that does the following:

A. (3 marks) Create two vectors and name them as **aVec** and **bVec**, respectively. The length of each vector is 50. The elements of these two vectors should be random integers sampled from the range 0 to 500. In order to get the same rando integers every time you generate the vectors, set the **seed** to a constant value of 75 before you create these vectors.

B. (2 marks) Create a vector that shows the result of subtracting aVec[1:n-1] from bVec[2:n] where n is the length of each vector and put the result in a 7x7 matrix filled by column.

C. (2 marks) Create a vector that shows the result of:

$$\left(\frac{\cos b_1}{\sin a_2}, \frac{\cos b_2}{\sin a_3}, \frac{\cos b_3}{\sin a_4}, \dots, \frac{\cos b_{n-1}}{\sin a_n}\right)$$

   where $b_n = \mathrm{bVec[n]}$ and $a_n = \mathrm{aVec[n]}$

D. (3 marks) Calculate the result of the following series:

$$\sum_{i=1}^{n-1} \frac{e^{-a_{i+1}}}{a_i + 5}$$

3. Part III (Bonus 10%)

Write an R function that converts any decimal number to its IEEE-754 32-bit floating point binary representation. Your function should accept a decimal number as input and should output the sign bit, the normalized mantissa and the biased exponent as binary strings of 1s and 0s without spaces in between. An example output would look as follows:

```
> Input: -103.5 IEEE-754, 32-bit floating point number representation:
> Sign = 1
> Mantissa (23 bits) = 10011110000000000000000
> Exponent (8 bits) = 10000101
```

> *“Any fool can write code that a computer can understand. Good programmers write code that humans can understand.” – Martin Fowler*