Assignment21
Assigned Wednesday Jan. 12, 2014
Due Friday Jan. 31, 2014
script with the functions created is necessary, along with another script that executes the functions asked for in this lab.

The objective of this lab is to familiarize yourself with the creation of functions of R, both functions that you have used before and a few that you have not yet encountered.

All functions should be contained in a script called "functions.r" and stored in your "lab2/" folder. You'll create a second script called "execute.r", whose first line will be "source('functions.r')" and the following lines will call the functions as demanded of you in the questions below. Use the commands learned in class as appropriate to solve the problems.

Look up any terms you do not understand (help in R using `?` or `help(topic)`, google, and TA, in that order.) You can also try the function `example.` Type `help(example)` to find out more about this function.

All plots should be saved to files in (png, jpeg or pdf format). TA can help if you cannot figure this out. We will learn (in the future) how to do this with R. For now, use copy/paster technique. Or use your computer to perform screen shots.

Zip/tar (remember the 1st lasb) all the files used in this project and send this archived file to the TA.

All functions should have "reasonable" names.

Scoring: each question is 20 points.

Task 1
Create a function (give it a reasonable name) that sums all the elements of a vector, and returns the sum to the user. The vector should be the sole argument to the function.

Task 2
Create a function (give it a reasonable name) that sums the cubes of all the elements of a vector, and returns the sum to the user. The vector should be the sole argument to the function.

The `plot` command with two arguments draws a scatter plot of x versus y. Create a function that first creates a vector of `n` random numbers, and then a second vector where each random number is cubed. Plot a scatter plot of the first vector against the second vector. Do "`plot`" to learn the basics of the plot command. Make sure you get the plot when you run the function. Look up any terms you do not understand (help in R, google, TA, in that order.). Figure out (with or without help) how to copy this plot and make a `jpeg` or `png` or `pdf` file for inclusion when you return your homework.

Task 3
Create a function that does the following: takes three vectors: x, y, and z.

Return a new vector that is defined by the following operation: add x and y, multiply the result by the standard deviation of z. Return the result to the user (i.e., the caller of this function.) Keep in mind that the three vectors should have the same length. Provide two examples where you call this function, once with vectors of length 20, and once with vectors of length 7. Use random numbers in one of your examples, and no random numbers in the other. Other than this restriction, use any example you want.

Task 4

Create a function that returns all the even numbers between m and n, assuming that m is even, and n is odd. So m and n are two arguments to the function. Make sure that n is always greater than m when calling the function. Later in the class, we will use "if statements" to do this check. For now, just call the function while paying attention to this yourself. The return value for the function is a vector that contains the even numbers. For example, if $n = 6$ and $m = 11$, your function should return $6, 8, 10$. If $n = 8$ and $m = 13$, the function returns $8, 10, 12$. Test this function with $n = 126$ and $m = 255$ and provide the results.

Task 5

Create a function that plots the results of the function written in task 4. So, given the two arguments m and n, generate a plot (using `plot()` command) generate a graph of the integers versus the index. Example: if $m = 4$ and $n = 11$, the plot should contain the points $(1, 4), (2, 6)$, $(3, 8)$, $(4, 10)$. Use the `main` argument from the `plot` command to create a title for your plot. If you can figure it out, create names for the x and y axes. Save the plot to a file.