CSC 121 — Lab Exercise 6

This is a non-credit exercise, which you do not hand in.
You may work on your own or together with another student, as you please.

In this lab, you will work with R "data frames", which are the main way in which data on many variables is handled in R.

Data frames are sort of like lists and sort of like matrices. A data frame holds information on some number of individuals, objects, cases, or whatever, which correspond to the rows of the data frame. For each row, the data frame has values for one or more variables, which can be numbers, logical (TRUE/FALSE) values, or character strings.

You can ask for just the values of one variable, `v`, from a data frame `D`, using `D$v`, the same as extracting a named element of a list. You can also ask for the values of just row `i`, using `D[i,]`, the same as for a matrix. You can also get individual values using expressions such as `D[i,3]` or `D$v[i]`.

The trees data frame. To start, you can look at the "trees" data frame, which is one of R's built-in data sets used for demonstrations. Try the following commands:

```
> help(trees)  # see the documentation on this built-in data frame
> trees        # print it in the console window
> View(trees)  # see it in RStudio's viewing window
> trees$Height # one variable
> trees[5,]    # one tree
```

You can play around with other similar commands and see what you get. You can then make a copy of the data frame and modify it:

```
> mytrees <- trees
> mytrees[2,3] <- 99
```

Try changing values or variables and see what happens. For example, can you figure out how to add one to all the values for `Volume`? What happens if you try to change a variable to one with the wrong length? Can you create a new variable and add it to the data frame? Can you do this with a logical or character variable? Can you add a new row to the data frame at the end?

Reading data from a file. Next, you should create a text file (with whatever editor you like, or with RStudio's menus File > New File > Text File). On the first line, write the names of the variables (the "header"). On later lines, put the values of these variables for each row. Just make up variables and values for an example.

After you've entered some data, save it with some name, and then try reading it with a command such as

```
> data <- read.table ("myfile", header=TRUE)
```

You can then view `data` like you did `trees` above.

Plots and summaries from a data frame. You can generate pairwise scatterplots of all variables with a command like `plot(trees)`. You can also try `boxplot(trees)`, which graphically displays much the same information as you get with `summary(trees)`.

Selecting rows. We often want to look at just a subset of rows in a data frame. We can use R's ability to use a *vector* as a subscript to do this. (This will be covered in more detail later in lectures.)

For example, see what you get with `trees[c(3,5,1,20),]`. (Do you see what's happened to the "row names"?) We can also use a vector of logical values, which we can get by a vector comparison. For example, see what you get with `trees[trees$Height>70,]`.

Creating new columns with ratios. We might be interested in whether we can estimate volume from girth (which is easier to measure). Create a copy of the trees data frame in which you add new columns with the ratios of volume to girth, to the square of girth, and to the cube of girth. How do these ratios seem to relate to height and girth?

Could you use some power of girth (perhaps a fractional power) to predict volume? You could try doing so, adding a new column to the data frame with your predicted volume, and plot the predicted versus true volume to see how well you do.