INTR4051/SOSS7091
Lab 2: Vectors, Indexing, and Functions

Name: _____

Create an RStudio project called `lab2`. Within it, do this assignment in an R Markdown file called `yoursurnameLab2.Rmd`. Start each question with a third-level header — i.e., something like:

```
### Question 1a
```

And remember to put your R code in code chunks. When you are done, upload the R Markdown file *and* the "knitted" HTML file to your Sakai dropbox, along with the PDF file from the last question.

Working with vectors

1. Create the following vectors in R as "elegantly" as you can. That is, try to use "patterned data" operators/functions (like ":", `rep()`, `seq()`, and others), and avoid purely "manual" data entry using `c()` where you can. (If you really don't see the pattern, don't waste too much time. You can use `c()` to make the vector and come back later if you figure out a better way.)
   (a) **x1** $= (2, 4, 6, 8, 10)$
   (b) **x2** $= (2, 4, 6, 10)$
   (c) **x3** $= (107.5, 108, 108.5, 109, 109.5)$
   (d) **x4** $= (11, 22, 33, 11, 22, 33)$
   (e) **x5** $= (11, 11, 22, 22, 33, 33)$
   (f) **x6** $= (3, 3, 1, 1, 1, -1)$
   (g) **x7** $= (4, 9, 15, 22)$
   (h) **x8** $= (-3, 3, 3, 9, 45)$
   ("Bonus" to anyone who can get this one without using `c()`. The function you need was covered in lecture, but the pattern is hard to see.)

Working with indexes

2. Create the following vectors solely by applying a square-bracket index to the specified vector (which should already be in your environment because you created if for question 4).

   For example, if the question was:

   xExample $= (2, 4)$, by indexing `x1`,

   The answer could be:

   ```
   xExample <- x1[1:2]
   ```
   or
   ```
   xExample <- x1[c(1,2)]
   ```

   (a) **x10** $= (2, 4)$, by indexing `x1`
   (b) **x11** $= (107.5, 109)$, by indexing `x3`
   (c) **x12** $= (109, 107.5)$, by indexing `x3`
   (d) **x13** $= (109.5, 109, 108.5, 108, 107.5)$, by indexing `x3`
   (e) **x14** $= (15, 22, 4, 9, 4)$, by indexing `x7`

Analyzing SACU GDP data

The Southern African Customs Union (SACU) consists of five countries — Botswana, Lesotho, Namibia, South Africa, and eSwatini.

A spreadsheet containing the gross domestic product (GDP) of each of these countries for each year from 2000 through 2014 is in the file `sacu.RData`, available for this lab on Sakai.

(The figures are from the World Bank's World Development Indicators Online, which is freely accessible on the World Bank website. The figures are denominated in "constant 2005 international dollars" — which means they are corrected for inflation and are intended to be internationally comparable.)

3. Load the file into your environment (workspace) in a code chunk using the following command:
   `load(file="sacuGDP.RData")`

   Check that you have new data objects in your environment. In the same chunk get R to display each vector.

4. Use an index to find the most recent GDP figure for South Africa (knowing that the vectors run in "time order," from 2000 to 2014).

5. Use a function to find to find the lowest GDP figure for Swaziland.

6. Use a function (or two) to figure out whether Lesotho's GDP ever declined year-on-year during the period 2000–2014.

7. Use R to figure out which country had the second-highest average GDP over the decade.

8. Use vector arithmetic to calculate a vector called `sacu` that contains the total GDP for SACU in each year. (Hint: You do not need a named function, just the "+" arithmetic operator). Get R to display the vector.

9. Enter the following command: `names(sacu) <- 2000:2014`
   Get R to display `sacu` again. What has changed?

10. Make a "line" plot of `sacu`. (See the slides for lecture 2 if you are unclear about this.) If that works and you are feeling colorful, add the "option" argument `col="red"` to the `plot()` command (remember that arguments must be separated by commas). See how the plot changes.

11. If your plot command was successful, enter the following three commands in a code chunk:

    ```
    pdf(file="lab2_SACUplot.pdf")
    your plot command from the previous question here
    dev.off()
    ```

    Check the "Files" tab in RStudio's lower-right pane to see if you can find the PDF file in your project/working directory. Double-click it to view the contents.