Tutorial 12 - Really programming in R

R relies on variables and RAM

We work in scripts so that we can rerun all of our saved commands and regenerate the same R environment each time we work on a project/assignment.

This is necessary because R relies on variables and RAM. These things are temporarily stored in the R environment, but "disappear" when R is closed.

Be sure you are working in and saving your scripts!

Parentheses, Square Brackets, Curly Braces, oh my!

`(  )` using functions

`[ ]` subsetting: accessing a portion of the contents of a data structure

`{  }` control flow: for loops, if-else statements, defining functions

for loops in R

In Bash, we looped through files in directories, with code like

```
for file in *.txt

do

     echo $file

done
```

In R, we commonly load full files into R and need to combine sets of integers with subsetting to loope through these data structures.

```
myDF <- read.csv("myData.csv")

for(line in 1:10){

     x <- myDF[line,]

     print(x)

}
```

for loop Review

Given a data frame called `data` that has 4 rows and 3 columns (with column names `A`, `B`, `C`), what code, USING LOOPS, would:

- print each row, one at a time
- print each element of the 3rd column, one at a time

Tips for working with loops

- when writing a loop, work on a single case, get it to work, and then generalize for each row
- think of the index variable as placeholder for each integer in the set you'll loop through; this is a bit like variables (`$1` or `$file`) in `bash`
- use print statements to figure out which parts are working and which are not; this is analogous to using `echo` in `bash` loops

if-else

This is a useful way to let your code make decisions for you

Given the outcome of a logic test, one or more behaviors can occur

```
if(x > 0){
     print("x is positive")
}else if(x < 0){
     print("x is negative")
}else{
     print("x is equal to zero")
}
```

if-else in a loop

You can even use if-else statements in loops. Actually this is where they are most useful!

Walk through:

Use a for loop to calculate the average sepal length for each species in the iris data set. Don't cheat and use the `mean()` function!

Challenge:

- Use a for loop and if-else statement to find the minimum petal width of Setosa iris in the iris data set. Don't cheat and use the `min()` function!

Challenge: more complex if-else statement

Create an R script that prints the max value of a vector containing three numbers. Don't create any additional variables to accomplish this task.

Challenge: a tough one

Create a script that reads a series of integers from a text file ("findRuns.txt" available on Sakai) and reports two things: 1) the index for the element at the beginning of runs of repeated values and 2) the length of the corresponding runs.

For example, in the vector 0, 1, 2, 2, 3, 4 your script should store and/or return the values 3 (because the repeated 2's begin in element 3 of the vector) and 2 (because there are two 2's in a row).

Your solution should include a for loop and at least one if-else statement. First, decompose the problem. Use comments to outline the steps to solve this challenge.

Exercise 12

Due in two weeks (12/2) because of Thanksgiving

Fork from TA's Github repository