Basic Course on R: Programming Structures 2 Practical

Elizabeth Ribble*

14-18 May 2018

Contents

_____

*emcclel3@msudenver.edu

1 Part A: Scope

1. For each of the following sets of commands, give the value that will be returned by the last command. Try to answer without using R.

a)
```r
w <- 5
f <- function(y) {
   return(w + y)
}
f(y = 2)
```

b)
```r
w <- 5
f <- function(y) {
   w <- 4
   return(w + y)
}
f(y = 2)
```

2. Among the variables `w`, `d`, and `y`, which are global to `f()` and which are local?

```r
w <- 2
f <- function(y) {
  h <- function() {
    d <- 3
    return(w + y)
  }
  return(d * h())
}
```

3. Do the following in R.

a) Try:
```r
myFun1 <- function() {
   a <- 2
   b <- 3
   myFun2(3)
}
myFun2 <- function(y) {
   return(y + a + b)
```

```
    }
myFun1()
```

What happens?

b) Now try:

```
a <- 1
b <- 2
myFun1()
```

What happens?

4. What value for w will be printed in the last line below? Try to answer without using R.

```
w <- 1
f <- function(y) {
  g <- function() {
    w <<- 3
    return(2)
  }
  return(g())
}
f(y = 1)
w
```

5. What value for w will be printed in the last line below? Try to answer without using R.

```
w <- 1
f <- function(y) {
  w <- 2
  g <- function() {
    w <<- 3
    return(2)
  }
  return(g())
}
f(y = 1)
w
```

2   Part B: `if()` Statements, `warning()`, and `stop()`

The functions `warning()` and `stop()` are used to print a warning message and to stop
the execution of the function call and print an error message. For example:

```r
noNegMean <- function(x) {
  if(all(x < 0)) {
    stop("All values in x are negative")
    }
  if(any(x < 0)) {
    x[x < 0] <- 0
    warning("Negative values in x replaced by zero")
    }
  return(mean(x))
  }
```

1. The file **nonegmean.txt** contains the above code; source it into R and then pass
   `noNegMean()` a vector containing some negative and some positive values. What
   happens?

2. What happens when you pass `noNegMean()` a vector containing all negative values?

3. Write a function `ratio()` that takes two arguments, `x` and `y`, and attempts to
   compute the ratio `x/y`. If both `x == 0 & y == 0`, the function should stop and
   print an error message about dividing 0 by 0. If `y == 0` (but not `x`), the function
   should print a warning message about dividing by 0, and then return `x/y` (which
   will be `Inf`). In all other cases, it should return `x/y`.

   Test your `ratio()` function first using two nonzero values for `x` and `y`, then using
   a nonzero `x` but `y = 0`, and finally using `x = 0` and `y = 0`.