Introduction to the Social Web

300958 Social Web Analysis

Week 2 Lab Solutions

Copy and past the below code into the RStudio text editor, save it and source it:

```r
> main <- function() {
+     f = findFactors(1012)
+     print(f)
+ }
>
> findFactors <- function(x) {
+     factors = c()
+     for (a in 1:x) {
+         if ((x %% a) == 0) {
+             factors = c(factors, a)
+         }
+     }
+     return(factors)
+ }
>
> main()
```

- What is the code doing?

Two functions are defined (main and findFactors), then main is called. main calls findFactors with the argument 1012, then prints the return value. findFactors searches for the set of factors of its argument x, by testing if the numbers from 1 to x divide into x.

- What is the `%%` operator?

This is the remainder operator. E.g. 10 %% 3 gives 1.

- What do you expect the output to be if we change 1012 to 16? Try it to check your answer.

The output will be 1, 2, 4, 8, 16.

- Can you think of a better way to write the `findFactors` function? If so, change the function (Hint: factors come in pairs, so we only need to check from 1 to $\sqrt{x}$).

```r
> findFactors <- function(x) {
+     factors = c()
+     upper = floor(sqrt(x))
+     for (a in 1:upper) {
```

```
+        if ((x %% a) == 0) {
+            factors = c(factors, a)
+        }
+    }
+    remainingFactors = rev(x/factors)
+    if (floor(sqrt(x)) == sqrt(x)) {
+        remainingFactors = remainingFactors[-1]
+    }
+    return(c(factors, remainingFactors))
+ }
>
> findFactors(16)
```

```
[1]  1  2  4  8 16
```

- Bonus task: Using the function `length`, write a function to test if a given number is prime.

```
> isPrime <- function(x) {
+    if (length(findFactors(x)) == 2) {
+        return(TRUE)
+    }
+    return(FALSE)
+ }
>
> isPrime(10)
```

```
[1] FALSE
```

```
> isPrime(19)
```

```
[1] TRUE
```