Week 2 Quiz

CALIFICACIÓN DEL ÚLTIMO ENVÍO

100%

1. Suppose I define the following function in R **1 / 1 puntos**

```
cube <- function(x, n) {
        x^3
}
```

What is the result of running

```
cube(3)
```

in R after defining this function?

- (•) The number 27 is returned
- ( ) A warning is given with no value returned.
- ( ) The users is prompted to specify the value of 'n'.
- ( ) An error is returned because 'n' is not specified in the call to 'cube'

✓ **Correcto**

Because 'n' is not evaluated, it is not needed even though it is a formal argument.

2. The following code will produce a warning in R. **1 / 1 puntos**

```
x <- 1:10
if(x > 5) {
        x <- 0
}
```

Why?

- ( ) There are no elements in 'x' that are greater than 5
- ( ) You cannot set 'x' to be 0 because 'x' is a vector and 0 is a scalar.

- The expression uses curly braces.
- (●) 'x' is a vector of length 10 and 'if' can only test a single logical statement.
- The syntax of this R expression is incorrect.

✓ **Correcto**

3. Consider the following function **1 / 1 puntos**

```
f <- function(x) {
        g <- function(y) {
                y + z
        }
        z <- 4
        x + g(x)
}
```

If I then run in R

```
z <- 10
f(3)
```

What value is returned?

- 4
- (●) 10
- 16
- 7

✓ **Correcto**

4. Consider the following expression: **1 / 1 puntos**

```
x <- 5
y <- if(x < 3) {
        NA
} else {
        10
}
```

What is the value of 'y' after evaluating this expression?

- (●) 10
- ( ) NA
- ( ) 5
- ( ) 3

✓ **Correcto**

5. Consider the following R function **1 / 1 puntos**

```
h <- function(x, y = NULL, d = 3L) {
        z <- cbind(x, d)
        if(!is.null(y))
                z <- z + y
        else
                z <- z + f
        g <- x + y / z
        if(d == 3L)
                return(g)
        g <- g + 10
        g
}
```

Which symbol in the above function is a free variable?

- (●) f
- ( ) z
- ( ) d
- ( ) L
- ( ) g

✓ **Correcto**

6. What is an environment in R? **1 / 1 puntos**

- ( ) a special type of function
- (●) a collection of symbol/value pairs

- a list whose elements are all functions
- an R package that only contains data

✓ **Correcto**

7. The R language uses what type of scoping rule for resolving free variables? **1 / 1 puntos**

- dynamic scoping
- compilation scoping
- lexical scoping
- global scoping

✓ **Correcto**

8. How are free variables in R functions resolved? **1 / 1 puntos**

- The values of free variables are searched for in the working directory
- The values of free variables are searched for in the environment in which the function was called
- The values of free variables are searched for in the environment in which the function was defined
- The values of free variables are searched for in the global environment

✓ **Correcto**

9. What is one of the consequences of the scoping rules used in R? **1 / 1 puntos**

- All objects can be stored on the disk
- Functions cannot be nested

- R objects cannot be larger than 100 MB
- All objects must be stored in memory

Correcto

10. In R, what is the parent frame? **1 / 1 puntos**

- It is always the global environment
- It is the environment in which a function was called
- It is the package search list
- It is the environment in which a function was defined

Correcto