# Laboratorio 2

*Alex Avila Santos*

*11 de junio de 2018*

**Ejercicio 4**

**Analiza el siguiente código**

```
loopvec1 <- 5:7
loopvec2 <- 9:6
mat1 <- matrix(NA,length(loopvec1),length(loopvec2))
mat1
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   NA   NA   NA   NA
## [2,]   NA   NA   NA   NA
## [3,]   NA   NA   NA   NA
```

**El siguiente bucle anidado completa mat1 con el resultado de multiplicar cada entero en loopvec1 por cada entero en loopvec2:**

```
for(i in 1:length(loopvec1)){
  for(j in 1:length(loopvec2)){
  mat1[i,j] <- loopvec1[i]*loopvec2[j]
  }
}
mat1
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   45   40   35   30
## [2,]   54   48   42   36
## [3,]   63   56   49   42
```

**a) En aras de una codificación eficiente, reescribe el ejemplo de bucle anidado anterior, donde la matriz mat1 se completó con los múltiplos de los elementos de loopvec1 y loopvec2, utilizando solo un único bucle for.**

```
  #Dado que cada fila es un multiplo del vector2 entonces
  #multiplicamos cada elemento del vector 1 con el vector 2
  for(i in 1:length(loopvec1)){
    mat1[i,] <- loopvec1[i]*loopvec2
  }
  mat1
```

```
##      [,1] [,2] [,3] [,4]
## [1,]   45   40   35   30
## [2,]   54   48   42   36
## [3,]   63   56   49   42
```

**b) Usamos el comando de R**

**switch(EXPR=cadena1,Homer=12,Marge=34,Bart=56,Lisa=78,Maggie=90, NA)**

**para devolver un número basado en el valor proporcionado de una cadena de caracteres individuales.**

**Esta línea no funcionará si cadena1 es un vector de caracteres. Escribe código que tomará un vector de caracteres y devolverá un vector de los valores numéricos apropiados relacionado a la cadena de caracteres. Pruéba tu respuesta con el siguiente vector:**

```
cadena1 <- c("Peter","Homer","Lois","Stewie","Maggie","Bart")
#vector para almacenar los valores numéricos
cadena2 <- c()
#Como solo podemos escribir vectores de longitud 1 entonces lo haremos uno por uno
for(i in 1:length(cadena1))
  cadena2[i] <- switch(EXPR=cadena1[i],Homer=12,Marge=34,Bart=56,Lisa=78,Maggie=90, NA)

cadena2
```

```
## [1] NA 12 NA NA 90 56
```

**c) Supongamos que tenemos una lista llamada unalista que puede contener otras listas, pero supongamos que esas listas miembros de una lista no pueden contener listas. Escribe bucles anidados que puedan buscar cualquier unalista posible definida de esta manera y cuenta cuántas matrices están presentes.**

**Sugerencia: Configura un contador antes de comenzar los bucles que se incrementan cada vez que se encuentra una matriz, independientemente de si se trata de un miembro directo de unalista o si es miembro de una lista miembro de unalista. Verifica tus resultados:**

. La respuesta es 4 si tienes lo siguiente:

```
unalista <- list(aa=c(3.4,1),bb=matrix(1:4,2,2),
cc=matrix(c(T,T,F,T,F,F),3,2),dd="cadena aqui",
ee=list(c("hola","tu"),matrix(c("hola","there"))),
ff=matrix(c("red","green","blue","yellow")))

  #Veamos
  contador <- 0
  for(i in 1: length(unalista)){
    #Examinamos si el miembro directo de la lista "unalista" es una matrix
    if(is.matrix(unalista[[i]])) contador <- contador+1
    #Examinamos si el miembro de una lista miembro de "unalista" es una matrix.
    if(is.list(unalista[[i]])){
        aux <- unalista[[i]]
        for(j in 1:length(aux)){
          if(is.matrix(aux[[j]])) contador <- contador+1
        }
    }
  }
  #imprimos el contador
  contador
```

```
## [1] 4
```

. La respuesta es 0 si tienes lo siguiente:

```
unalista <- list("salio algo raro",as.vector(matrix(1:6,3,2)))

#Veamos
  contador <- 0
  for(i in 1: length(unalista)){
    #Examinamos si el miembro directo de la lista "unalista" es una matrix
    if(is.matrix(unalista[[i]])) contador <- contador+1
    #Examinamos si el miembro de una lista miembro de "unalista" es una matrix.
    if(is.list(unalista[[i]])){
        aux <- unalista[[i]]
        for(j in 1:length(aux)){
          if(is.matrix(aux[[j]])) contador <- contador+1
        }
    }
  }
  #imprimos el contador
  contador
```

```
## [1] 0
```

. La respuesta es 2 si tienes lo siguiente:

```
unalista<- list(list(1,2,3),list(c(3,2),2),list(c(1,2),matrix(c(1,2))),
rbind(1:10,100:91))

#Veamos
  contador <- 0
  for(i in 1: length(unalista)){
    #Examinamos si el miembro directo de la lista "unalista" es una matrix
    if(is.matrix(unalista[[i]])) contador <- contador+1
    #Examinamos si el miembro de una lista miembro de "unalista" es una matrix.
    if(is.list(unalista[[i]])){
        aux <- unalista[[i]]
        for(j in 1:length(aux)){
          if(is.matrix(aux[[j]])) contador <- contador+1
        }
    }
  }
  #imprimos el contador
  contador
```

```
## [1] 2
```