

Solución Problem Set 3

Santiago Lozano

5 de abril de 2020

2)

Escriba un programa que pida al usuario ingresar un número entero, y que imprima “par” si el número es par e “impar” si el número es impar. Agregue a su programa un código que genere una advertencia en caso de que el usuario ingrese algo diferente a un número entero: Error. El usuario debe ingresar un número entero. (Investigue por su cuenta cómo lograr dicha validación y la generación del mensaje.)

```
while (TRUE) {  
  x <- readline(prompt = "Ingrese el valor:")  
  if(is.na(as.integer(x))){  
    message("Debe ingresar un número")  
  }else if (!(as.numeric(x)==as.integer(x))){  
    message("Debe ingresar un número entero")  
  }else{  
    if ((as.integer(x) %% 2) ==0){  
      print(paste("El número",x,"es par"))  
      break  
    }else{  
      print(paste("El número",x,"es impar"))  
      break  
    }  
  }  
}
```

que despliega

```
Ingrese el valor:b  
Debe ingresar un número  
Ingrese el valor:c  
Debe ingresar un número  
Ingrese el valor:pi  
Debe ingresar un número  
Ingrese el valor:e  
Debe ingresar un número  
Ingrese el valor:4  
[1] "El número 4 es par"  
Warning messages:  
1: In as.integer(x) : NAs introduced by coercion  
2: In as.integer(x) : NAs introduced by coercion  
3: In as.integer(x) : NAs introduced by coercion  
4: In as.integer(x) : NAs introduced by coercion
```

3)

Escriba un for loop que imprima todos los múltiplos de 3, desde 40 hasta 0 en orden decreciente. Esto es, 39, 36, 33, ..., 3, 0.

```
s <-c()
for(i in 40:0){
  if((i %% 3) == 0){
    s <- c(s,i)
  }
}
s

## [1] 39 36 33 30 27 24 21 18 15 12 9 6 3 0
```

4)

Escriba un loop que imprima todos los números entre 6 y 30 que no son divisibles por 2, 3 o 5.

```
s <-c()
for(i in 6:30){
  if(!((i %% 3) == 0)|((i %% 2) == 0)|((i %% 5) == 0)){
    s <- c(s,i)
  }
}
s

## [1] 7 11 13 17 19 23 29
```

5)

Escriba un programa llamado “Adivine el número”. El computador generará aleatoriamente un entero entre 1 y 100. El usuario digita un número y el computador responde “Menor” si el número aleatorio es menor que el escogido por el usuario, “Mayor” si el número aleatorio es mayor, y “Correcto!” si el usuario adivina el número. El jugador puede continuar ingresando números hasta que adivine correctamente.

Ejemplo: - El número aleatorio es 79. - El computador muestra el texto “Adivine el número entre 1 y 100” y espera a que el usuario lo digite. - El usuario digita el número que está abajo en *itálicas*. - El computador devuelve uno de tres textos, según el caso: “Mayor”, “Menor”, o “Correcto!”. Adivine el número entre 1 y 100: 40 Mayor Adivine el número entre 1 y 100: 70 Mayor Adivine el número entre 1 y 100:80 Menor Adivine el número entre 1 y 100: 77 Mayor Adivine el número entre 1 y 100: 79 Correcto!

```
gues <- sample(1:100,1)
while(TRUE){
  alea <- as.numeric(readline(prompt = "Adivine el número entre 1 y 100: "))
  if(alea < gues){
    message("Menor")
  }else if(alea > gues){
    message("Mayor")
  }else{
    message(";Correcto!")
    break
  }
}

Adivine el número entre 1 y 100: 4
Menor
Adivine el número entre 1 y 100: 20
Menor
Adivine el número entre 1 y 100: 40
```

```
Mayor
Adivine el número entre 1 y 100: 35
Menor
Adivine el número entre 1 y 100: 38
Mayor
Adivine el número entre 1 y 100: 37
¡Correcto!
```

6)

Escriba una función que ordene (de forma ascendente y descendente) un vector según sus valores.

Tomando el vector

```
ord <- sample(1:100,20)
ord
```

```
## [1] 20 46 67 23 9 66 56 47 96 22 79 87 12 41 14 11 17 38 92 95
```

e implementando el programa

```
for (i in 1:(length(ord)-1)){
  #print(ord[i])
  for (j in (i+1):(length(ord))){
    #print(ord[j])
    if ((ord[j]) < (ord[i])){
      p <- ord[j]
      q <- ord[i]
      ord[i] <- p
      ord[j] <- q
    }
  }
}
```

obtenemos

```
ord
```

```
## [1] 9 11 12 14 17 20 22 23 38 41 46 47 56 66 67 79 87 92 95 96
```

7)

Escriba una función que agregue un elemento a un vector

```
agre <- function(xx,ee){
  z <- c(xx,ee)
  return(z)
}
```

implementando

```
f <-agre(c(3,"a"),4)
f
```

```
## [1] "3" "a" "4"
```