Metodo: La conjetura de ULAM

Lenguaje utilizado: R

Integrantes:


```
Castillo Reyes Eduardo Armando
Vázquez Portuguez José Antonio
Maravilla Pérez Vianey
```

Docente

```
M. en C. Cristal Karina Galindo Durán
```


4 de septiembre de 2021

Problema:

La conjetura de Collatz, conocida también como conjetura 3n+1 o conjetura de Ulam (entre otros nombres), fue enunciada por el matemático Lothar Collatz en 1937, y a la fecha no se ha resuelto.
Enunciado
Sea la siguiente operación, aplicable a cualquier número entero positivo:

- Si el número es par, se divide entre 2.
- Si el número es impar, se multiplica por 3 y se suma 1.

Formalmente, esto equivale a una función:

Si N es par:

$$f(n) = n/2$$

Si N es impart

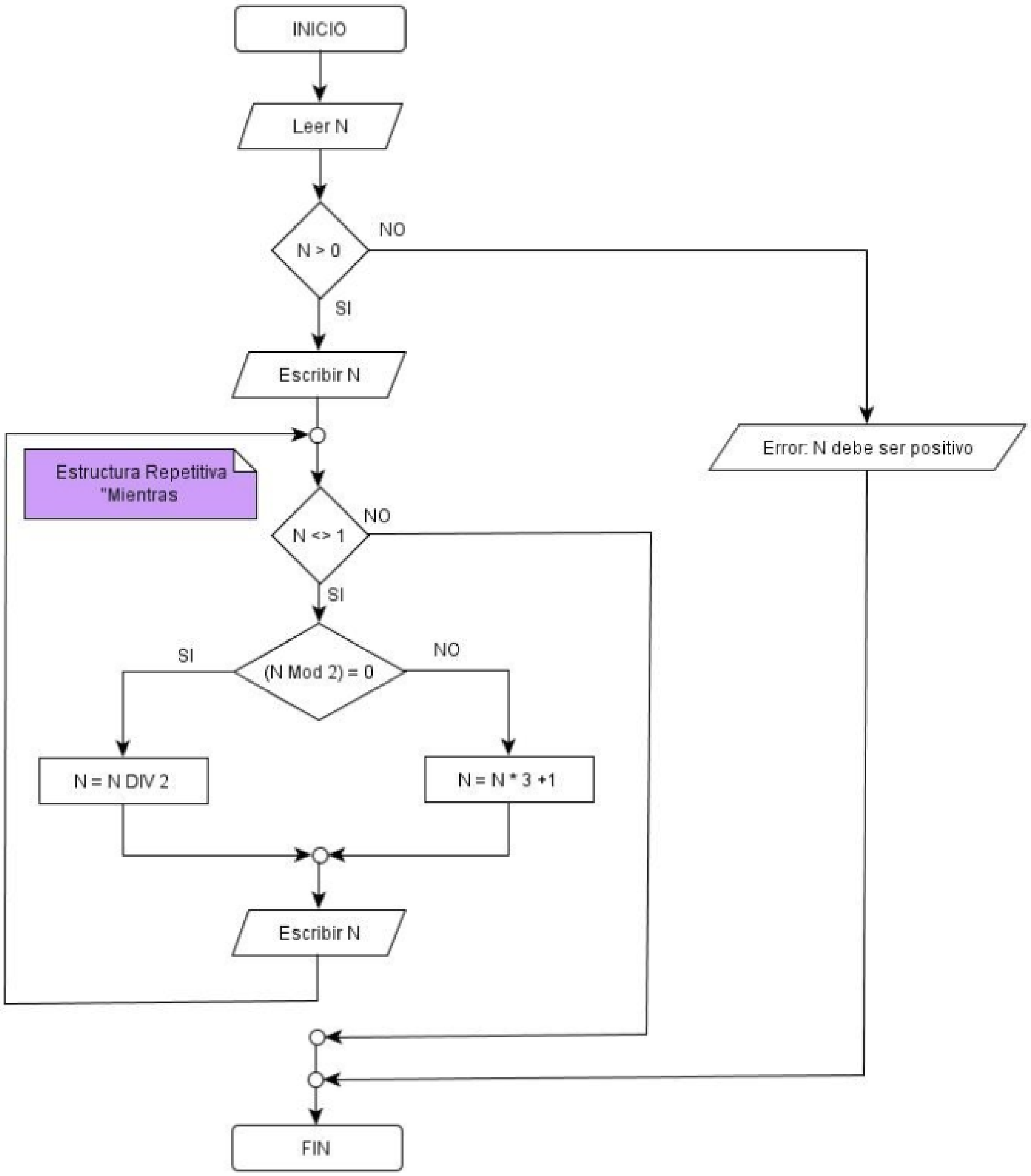
$$3n + 1$$

Consideraciones

Para la validación del correcto funcionamiento del programa se utilizaron las siguientes restricciones en el programa:

- -Solo entrada/lectura de números Naturales
- -Solo números menores a 232 (tamaño de Integer)

Diagrama de flujo



PSEUDOCODIGO

Escribir "Favor ingresar el número: "

Leer numero

valor <- número

Mientras(valor > 1)

Si (Valor 2 = 0) Entonces

valor <- trunc(valor / 2)

Escribir valor

Sino

valor = (valor * 3) + 1

Escribir valor

FinSi

FinMientras

Escribir "El numero: ", numero, "Tiene como conjetura de ulam consecutivamente: ", valor

FinProceso

CODIGO

In [2]:

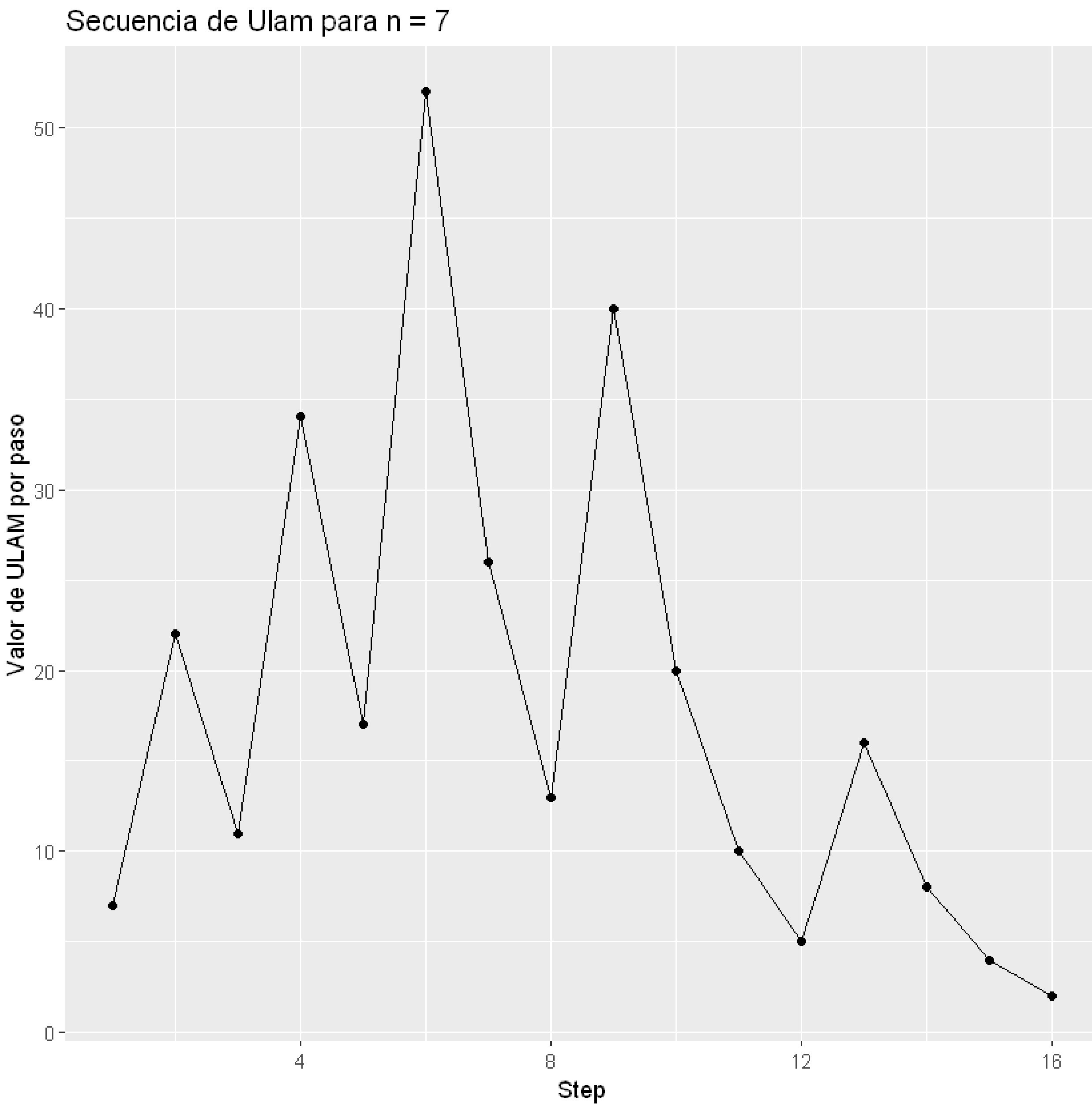
```
ulam <- function(n, limit = 10000) {
  # validacion:
  n <- suppressWarnings(as.integer(n))
  isValid <- !is.na(n) && n > 1
  if (!isValid ) {
    return(cat("Se necesita un entero mas grande que  1.  Prueba de nuevo."))
  }
  # definimos ulam:
  ulamRegla <- function(m) {
    if ( m %% 2 == 0) {
      return(m/2)
    } else {
      return(3*m + 1)
    }
  }

  library(ggplot2)

  # Grabamos el numero inicial por que se cambia despues:
  inicial <- n
  numeros <- numeric(limit)
  contador <- 0
  while ( n > 1 & contador < limit) {
    contador <- contador + 1
    numeros[contador] <- n
    n <- ulamRegla(n)
  }
  howMany <- min(contador, limit)
  cat("La conjetura de ulam tiene:  ", howMany, " elelementos.\n", sep = "")
  show <- readline("Quieres visualizarla (y/n)?  ")
  if ( show == "y" ) {
    print(numeros[1:howMany])
  }
  plotTitle <- paste0("Secuencia de Ulam para n = ", inicial)
  steps <- 1:howMany
  ggplot(mapping = aes(x = steps, y = numeros[1:howMany])) +
    geom_point() + geom_line() +
    labs( x = "Step", y = "Valor de ULAM por paso",
          title = plotTitle)
}
```

Pruebas de escritorio

In [3]:

```
ulam(7)
```

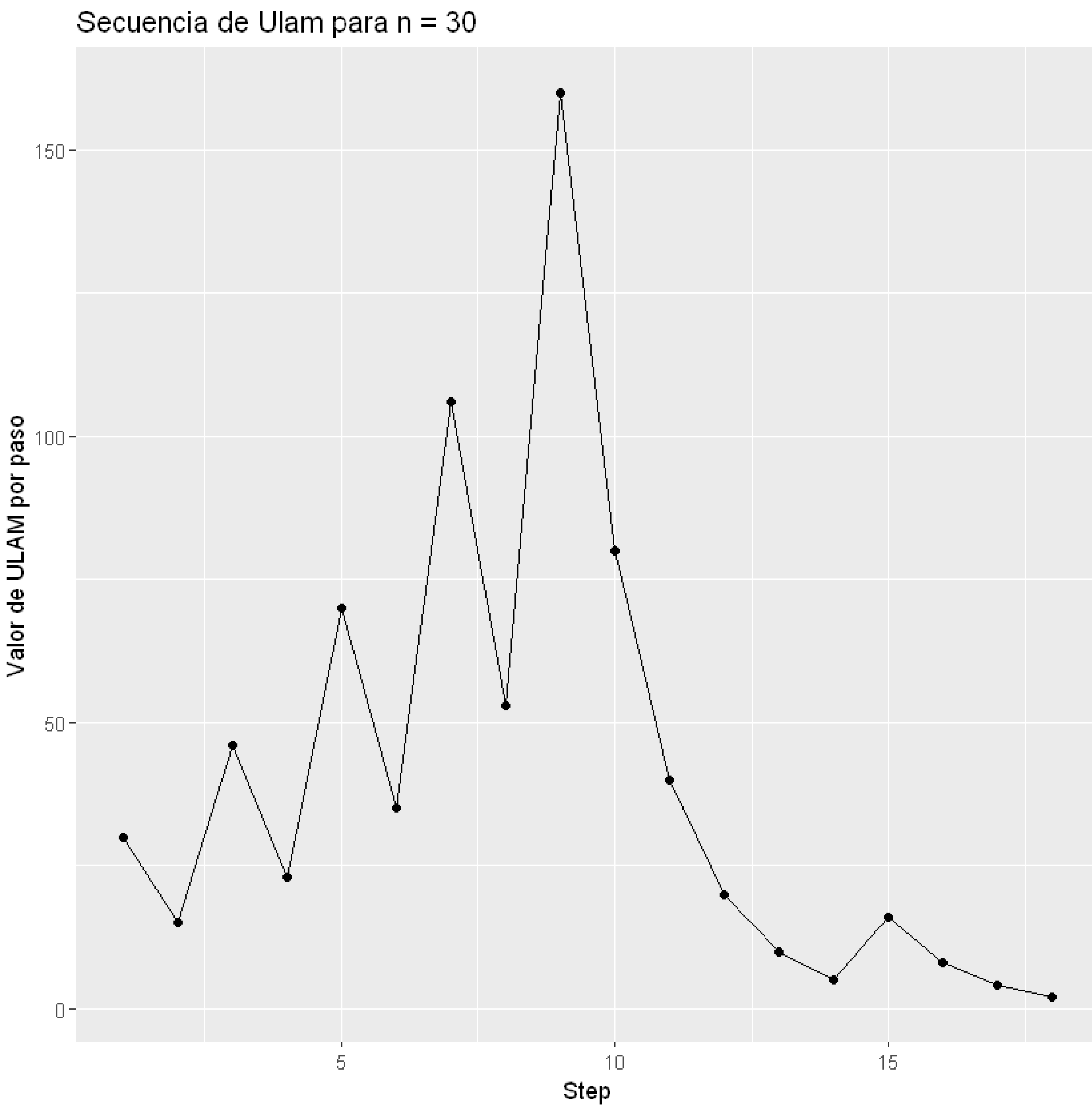
```
Registered S3 methods overwritten by 'ggplot2':
  method         from 
  [.quosures     rlang
  c.quosures     rlang
  print.quosures rlang
Quieres visualizarla (y/n)?  y
La conjetura de ulam tiene:  16 elelementos.
 [1]  7 22 11 34 17 52 26 13 40 20 10  5 16  8  4  2
```



In [4]:

```
ulam(30)
```

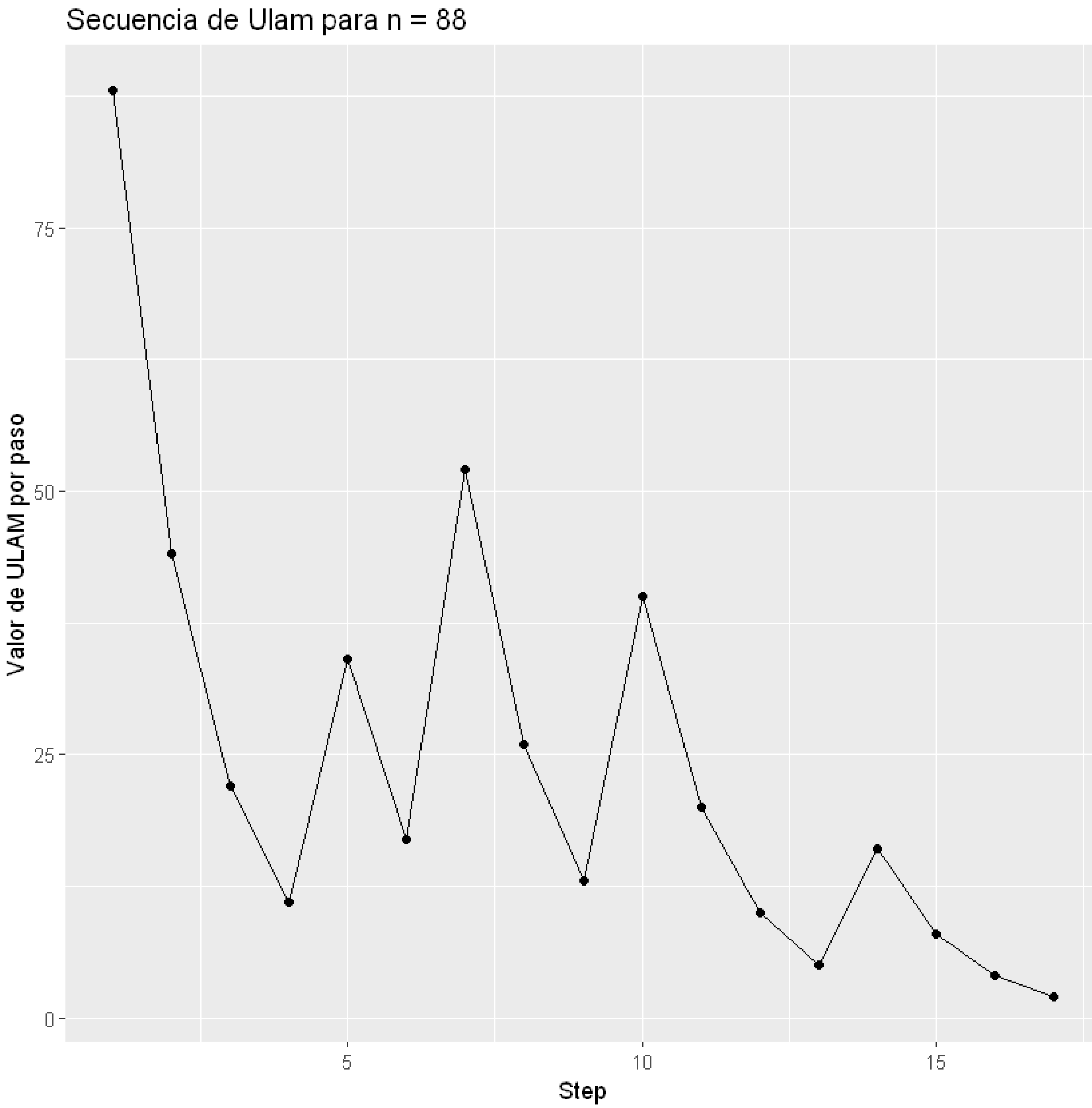
```
Quieres visualizarla (y/n)?  y
La conjetura de ulam tiene:  18 elelementos.
 [1]  30  15  46  23  70  35 106  53 160  80  40  20  10   5  16   8   4   2
```



In [5]:

```
ulam(88)
```

```
Quieres visualizarla (y/n)?  y
La conjetura de ulam tiene:  17 elelementos.
 [1] 88 44 22 11 34 17 52 26 13 40 20 10  5 16  8  4  2
```

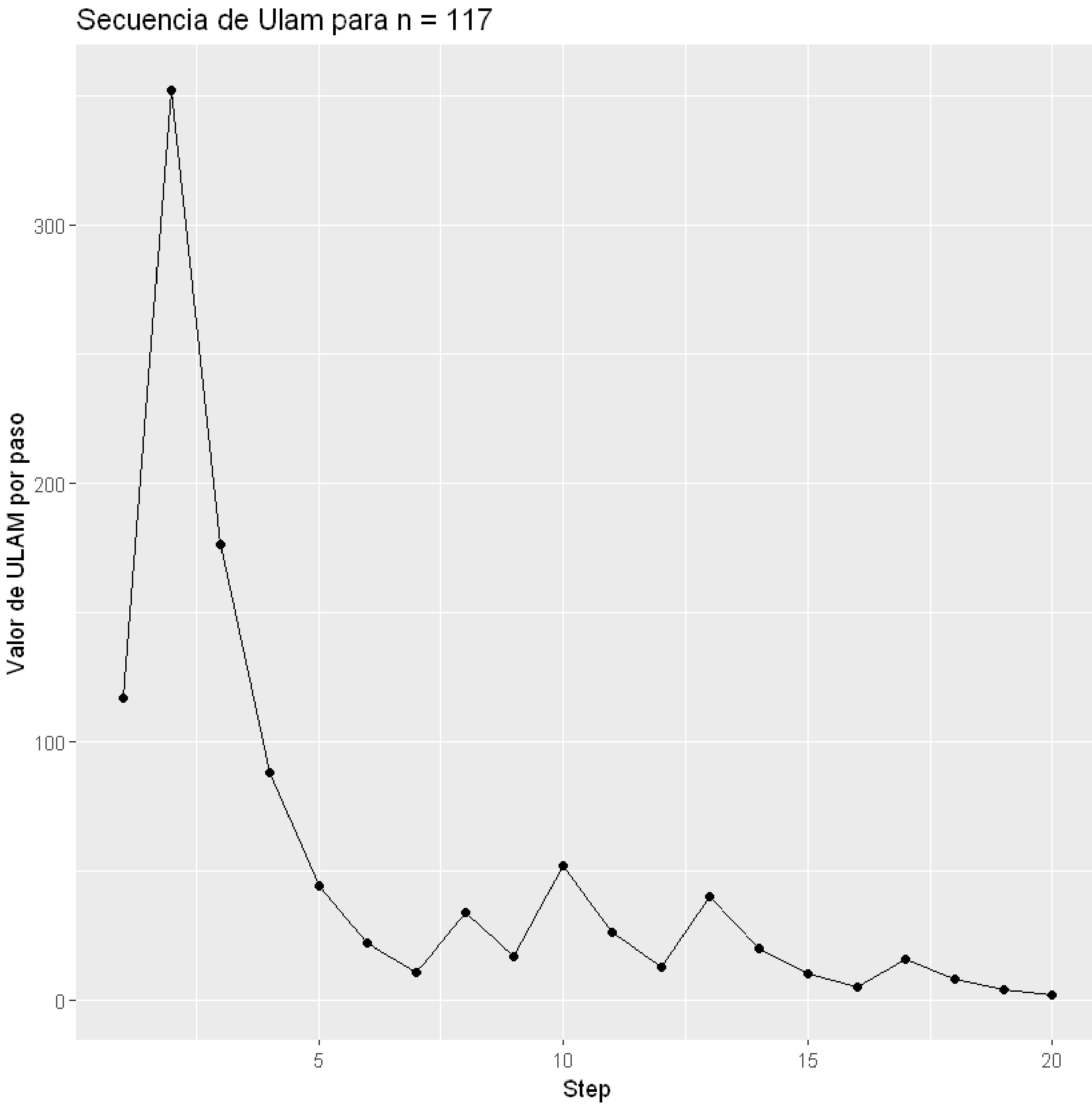


In [6]:
```
ulam(0)
```

```
Se necesita un entero mas grande que  1.  Prueba de nuevo.
```

In [7]:
```
ulam(117)
```

```
Quieres visualizarla (y/n)?  y
La conjetura de ulam tiene:  20 elelementos.
 [1] 117 352 176  88  44  22  11  34  17  52  26  13  40  20  10   5  16   8   4
[20]   2
```



Conclusión

Se demostró que existen operaciones aplicables a cualquier entero positivo, sea número par o número impar. Donde para los números pares f(n)=n2y los números impares f(n)=3n+1. Debido a lo anterior se supo demostrar que no existen ciclos cerrados o secuencias divergentes de numeros impares que no complan la conjetura, por lo tanto el proceso de multiplicar por 3 cualquier número impar y al resultado sumarle uno, no puede contener ciclos cerrados, ni el proceso es infinito.

```
In [ ]:
```