# La conjetura de Ulam En lenguaje R

### Integrantes

Castillo Reyes Eduardo Armando Maravilla Pérez Vianey Vázquez Portuguez José Antonio

#### **Docente**

M. en C. Cristal Karina Galindo Durán

4 de septiembre del 2021

#### Tabla de contenido

Problema	1
Consideraciones	1
Diagrama de flujo	2
Pseudocódigo	3
Pruebas de escritorio	4
Conclusión	5

#### **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:  $f: N \to N$ 

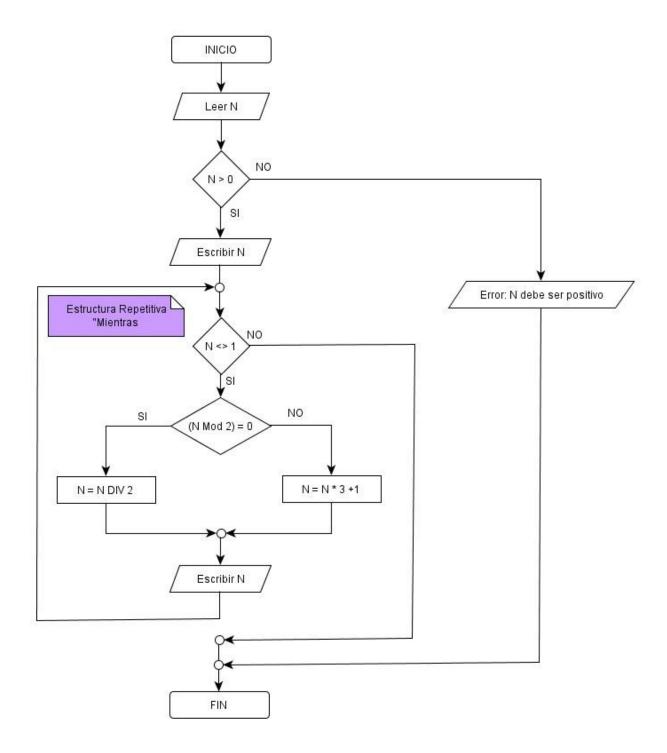
$$f(n) = \left\{ egin{array}{ll} rac{n}{2}, & ext{si } n ext{ es par} \ 3n+1, & ext{si } n ext{ es impar} \end{array} 
ight.$$

#### 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 2<sup>32</sup> (tamaño de Integer)

## Diagrama de flujo



## Pseudocódigo

```
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
```

## Pruebas de escritorio

Descripción de contencia	Contencia		ie\/elis	inicial	numara n	contrate	ebeur	numeros
escripción de sentencia	Sentencia	n		inicial	numeros[]	contador	snow	numeros
ectura de numero	n <- suppressWarnings(as.integer(n))	7						
alidación de numero y mayor a 1	isValid <- !is.na(n) && n > 1		true					
eclaracion de inicial con valor n	inicial <- n			7				
eclaracion de vector con limite de 1000	numeros <- numeric(limit=1000)				0			
Peclaración de contador con valor 0	contador <- 0					0		
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
sumento de contador	contador <- contador + 1					1		
signación de valor en el vector	numeros[contador] <- n				[1]=7			
lamada a la funcion ulam	n <- ulamRegla(n)							
condicion de residuo par igual a 0	m %% 2 == 0							
tetorna el valor a la mitad	return(m/2)							
tetorna el valor al triple más uno	return(3*m + 1)	22						
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1					2		
signación de valor en el vector	numeros[contador] <- n				[2] = 22			
lamada a la funcion ulam	n <- ulamRegla(n)				[-]			
condicion de residuo par igual a 0	m %% 2 == 0							
etorna el valor a la mitad	return(m/2)	11						
		- "						
tetorna el valor al triple más uno	return(3*m + 1) n > 1 & contador < limit							
condicion de paro con n mayor a 1 y contador menor al limite								
umento de contador	contador <- contador + 1				ron - 44	3		
signación de valor en el vector	numeros[contador] <- n				[3] = 11			
lamada a la funcion ulam	n <- ulamRegla(n)							
condicion de residuo par igual a 0	m %% 2 == 0							
etorna el valor a la mitad	return(m/2)							
tetorna el valor al triple más uno	return(3*m + 1)	34						
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1					4		
signación de valor en el vector	numeros[contador] <- n				[4] = 34			
lamada a la funcion ulam	n <- ulamRegla(n)							
condicion de residuo par igual a 0	m %% 2 == 0							
tetorna el valor a la mitad	return(m/2)	17						
tetorna el valor al triple más uno	return(3*m + 1)							
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1					5		
signación de valor en el vector	numeros[contador] <- n				[5] = 17	5		
lamada a la funcion ulam	n <- ulamRegla(n)							
condicion de residuo par igual a 0	m %% 2 == 0							
tetorna el valor a la mitad	return(m/2)							
letorna el valor al triple más uno	return(3*m + 1)	52						
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1					6		
signación de valor en el vector	numeros[contador] <- n				[6] = 52			
lamada a la funcion ulam	n <- ulamRegla(n)							
condicion de residuo par igual a 0	m %% 2 == 0							
tetorna el valor a la mitad	return(m/2)	26						
tetorna el valor al triple más uno	return(3*m + 1)							
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1					7		
signación de valor en el vector	numeros[contador] <- n				[7] = 26			
lamada a la funcion ulam	n <- ulamRegla(n)							
ondicion de residuo par igual a 0	m %% 2 == 0							
etorna el valor a la mitad	return(m/2)	13						
etorna el valor al triple más uno	return(3*m + 1)							
ondicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1				101 - 10	8		
signación de valor en el vector	numeros[contador] <- n				[8] = 13			
amada a la funcion ulam	n <- ulamRegla(n)							
ondicion de residuo par igual a 0	m %% 2 == 0							
etorna el velor a la mitad	return(m/2)							
etorna el valor al triple más uno	return(3*m + 1)	40						
condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
umento de contador	contador <- contador + 1				101 - 40	9		
signación de valor en el vector	numeros[contador] <- n				[9] = 40			
lamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0 Retorna el valor a la mitad	m %% 2 == 0	20						
	return(m/2)	20						
Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite	return(3*m + 1) n > 1 & contador < limit							

Aumento de contador	contador <- contador + 1			10	
Asignación de valor en el vector	numeros[contador] <- n		[10] = 20		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	10			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limi	itén > 1 & contador < limit				i
Aumento de contador	contador <- contador + 1			11	
Asignación de valor en el vector	numeros[contador] <- n		[11] = 10		
Llamada a la funcion ulam	n <- ulamRegla(n)		• •		
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	5			
Retorna el valor al triple más uno	return(3*m + 1)				i
Condicion de paro con n mayor a 1 y contador menor al limi					
Aumento de contador	contador <- contador + 1			12	
Asignación de valor en el vector	numeros[contador] <- n		[12] = 5		
Llamada a la funcion ulam	n <- ulamRegla(n)		[.2]		
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor a la rintada Retorna el valor al triple más uno	return(3*m + 1)	16			
Condicion de paro con n mayor a 1 y contador menor al limi		10			
Aumento de contador	contador <- contador + 1			13	1
Asignación de valor en el vector	numeros[contador] <- n		[13] = 16	13	
Llamada a la funcion ulam	n <- ulamRegla(n)		[13] - 10		
Condicion de residuo par igual a 0	m %% 2 == 0				
		8			
Retorna el valor a la mitad	return(m/2)	0			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limi				4.4	
Aumento de contador	contador <- contador + 1			14	
Asignación de valor en el vector	numeros[contador] <- n		[14] = 8		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	4			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limi	ten > 1 & contador < limit				į .
Aumento de contador	contador <- contador + 1			15	
Asignación de valor en el vector	numeros[contador] <- n		[15] = 4		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				1
Retorna el valor a la mitad	return(m/2)	2			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limi	ten > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			16	
Asignación de valor en el vector	numeros[contador] <- n		[16]=2		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	1			į .
Retorna el valor al triple más uno	return(3*m + 1)				
Encuentra el minimo entre contador y limit	howMany <- min(contador, limit)				
Cuadro de texto para visualizar grafica	show <- readline("Quieres visualizarla (y/n)? ")			у	
Condición de valor en show igual a 'y'	show == "V"			,	
The state of the s					7 22 11 34 17 52 26 13 40
Impresión de los numeros	print(numeros[1:howMany])				20 10 5 16 8 4 2
Impresion as les humeres	printing increase in the printing in the print				20 10 0 10 0 4 2

Prueba de escritorio 1. Número = 30				
Descripción de sentencia	Sentencia	n isValid i	nicial numeros[] co	ntador show numeros
Lectura de numero	n <- suppressWarnings(as.integer(n))	7		
Validación de numero y mayor a 1	isValid <- !is.na(n) && n > 1	true		
Declaracion de inicial con valor n	inicial <- n		7	
Declaración de vector con limite de 1000	numeros <- numeric(limit=1000)		n .	
Declaración de contador con valor 0	contador <- 0		ш	0
Condicion de paro con n mayor a 1 y contador menor al limi				•
Aumento de contador	contador <- contador + 1			
			141.00	1
Asignación de valor en el vector	numeros[contador] <- n		[1]=30	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)	15		
Retorna el valor al triple más uno	return(3*m + 1)			
Condicion de paro con n mayor a 1 y contador menor al limi	te n > 1 & contador < limit			
Aumento de contador	contador <- contador + 1			2
Asignación de valor en el vector	numeros[contador] <- n		[2] = 15	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)			
Retorna el valor al triple más uno	return(3*m + 1)	46		
Condicion de paro con n mayor a 1 y contador menor al limi				
Aumento de contador	contador <- contador + 1			3
Asignación de valor en el vector	numeros[contador] <- n		[3] = 46	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)	23		
Retorna el valor al triple más uno	return(3*m + 1)			
Condicion de paro con n mayor a 1 y contador menor al limi Aumento de contador	contador <- contador + 1			4
Asignación de valor en el vector	numeros[contador] <- n		[4] = 23	7
Llamada a la funcion ulam	n <- ulamRegla(n)		.,	
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)			
Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limi	return(3*m + 1)	70		
Condición de paro con n'mayor a 1 y contador menor arilmi Aumento de contador	contador <- contador + 1			5
Asignación de valor en el vector	numeros[contador] <- n		[5] = 70	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)	35		
Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limi	return(3*m + 1)			
Aumento de contador	contador <- contador + 1			6
Asignación de valor en el vector	numeros[contador] <- n		[6] = 35	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)	106		
Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limi	return(3*m + 1)	100		
Aumento de contador	contador <- contador + 1			7
Asignación de valor en el vector	numeros[contador] <- n		[7] = 106	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	n <- ulamRegla(n) m %% 2 == 0 return(m/2)	53		

Retorna el valor al triple más uno	return(3*m + 1)			
Condicion de paro con n mayor a 1 y contador menor al limit				
Aumento de contador	contador <- contador + 1			8
Asignación de valor en el vector	numeros[contador] <- n		[8] = 53	
Llamada a la funcion ulam	n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)			i
Retorna el valor al triple más uno	return(3*m + 1)	160		
Condicion de paro con n mayor a 1 y contador menor al limit				
Aumento de contador	contador <- contador + 1			9
Asignación de valor en el vector	numeros[contador] <- n		[9] = 160	
Llamada a la funcion ulam	n <- ulamRegla(n)		[0] - 100	
Condicion de residuo par igual a 0				-
	m %% 2 == 0	80		1
Retorna el valor a la mitad	return(m/2)	80		į
Retorna el valor al triple más uno	return(3*m + 1)			
Condicion de paro con n mayor a 1 y contador menor al limit				
Aumento de contador	contador <- contador + 1			10
Asignación de valor en el vector	numeros[contador] <- n		[10] = 80	i
Llamada a la funcion ulam	∤n <- ulamRegla(n)			
Condicion de residuo par igual a 0	m %% 2 == 0			
Retorna el valor a la mitad	return(m/2)	40		
Retorna el valor al triple más uno	return(3*m + 1)			i
Condicion de paro con n mayor a 1 y contador menor al limit				
Aumento de contador	contador <- contador + 1			11
Asignación de valor en el vector	numeros[contador] <- n		[11] = 40	
Llamada a la funcion ulam	n <- ulamRegla(n)		[11]-40	į
	m %% 2 == 0			
Condicion de residuo par igual a 0		20		
Retorna el valor a la mitad	return(m/2)	20		
Retorna el valor al triple más uno	return(3*m + 1)			i
Condicion de paro con n mayor a 1 y contador menor al limit				
Aumento de contador	contador <- contador + 1			12
Asignación de valor en el vector	numeros[contador] <- n		[12] = 20	1
Asignación de valor en el vector Llamada a la funcion ulam	numeros[contador] <- n  n <- ulamRegla(n)		[12] = 20	
-			[12] = 20	
Llamada a la funcion ulam	n <- ulamRegla(n)	10	[12] = 20	
Llamada a la funcion ulam Condicion de residuo par igual a 0	n <- ulamRegla(n) m %% 2 == 0	10	[12] = 20	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1)	10	[12] = 20	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limitr	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit	10	[12] = 20	13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limit Aumento de contador	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1	10		13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n	10	[12] = 20 [13] = 10	13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	n <- ulamRegla(n)	10		13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0			13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2)	10 5		13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2)			13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit			
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1		[13] = 10	13
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador < limit contador <- contador <- n			
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1		[13] = 10	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador < limit contador <- contador <- n		[13] = 10	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n)		[13] = 10	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0		[13] = 10	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Aumento de contador Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador) <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador) <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2)	5	[13] = 10	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de residuo par igual a 0 Retorna el valor a ltriple más uno Condicion de paro con n mayor a 1 y contador menor al limite	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador <- imit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador <- contador + 1 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit	5	[13] = 10	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la riple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) s <- 2 == 0 return(m/2) return(3*m + 1) r > 1 & contador < limit contador <- contador < limit contador <- contador < limit	5	[13] = 10 [14] = 5	
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor al triple más uno Condicion de residuo par igual a 0 Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador <- limit contador <- contador <- limit	5	[13] = 10	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n)	5	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de valor en el vector Llamada a la funcion ulam Condicion de valor en el vector Llamada a la funcion ulam Condicion de valor en el vector Llamada a la funcion ulam Condicion de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0	5 16	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(a*m + 1) return(m/2) return(m/2)	5	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Retorna de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador) <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador) <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2)	5 16	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la ripipe más uno Condicion de paro con n mayor a 1 y contador menor al limite	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit	5 16	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador <- limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3"m + 1) n > 1 & contador <- limit contador <- contador <- limit	5 16	[13] = 10 [14] = 5 [15] = 16	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la riple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con mayor a 1 y contador menor al limite Condicion de paro con mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador < limit contador <- contador <- limit	5 16	[13] = 10 [14] = 5	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor an la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(3"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(m/2) return(s"m + 1) n > 1 & contador < limit contador <- contador + 1 numeros(contador] <- n n <- ulamRegla(n)	5 16	[13] = 10 [14] = 5 [15] = 16	14
Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la mitad Retorna el valor a la riple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con n mayor a 1 y contador menor al limite Condicion de paro con mayor a 1 y contador menor al limite Condicion de paro con mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit contador <- contador < limit contador <- contador <- limit	5 16	[13] = 10 [14] = 5 [15] = 16	14

Re	torna el valor a la mitad torna el valor al triple más uno ndicion de paro con n mayor a 1 y contador menor al limite	return(m/2) return(3*m + 1) n > 1 & contador < limit	4			
Aui	mento de contador	contador <- contador + 1			17	1
Asi	gnación de valor en el vector	numeros[contador] <- n		[16]=4		
	<u> </u>			11		
Lla	mada a la funcion ulam	n <- ulamRegla(n)				
Co	ndicion de residuo par igual a 0	m %% 2 == 0				
	torna el valor a la mitad	return(m/2)	2			
Re	torna el valor al triple más uno	return(3*m + 1)				
	ndicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aui	mento de contador	contador <- contador + 1			18	
Asi	gnación de valor en el vector	numeros[contador] <- n		[16]=2		
Lla	mada a la funcion ulam	n <- ulamRegla(n)		-		
Co	ndicion de residuo par igual a 0	m %% 2 == 0				
Re	torna el valor a la mitad	return(m/2)	1			
Re	torna el valor al triple más uno	return(3*m + 1)				
En	cuentra el minimo entre contador y limit	howMany <- min(contador, limit)				1
Cu	adro de texto para visualizar grafica	show <- readline("Quieres visualizarla (y/n)? ")			у	
Co	ndición de valor en show igual a 'y'	show == "y"				
						7 22 11 34 17
						52 26 13 40 20
Im	presión de los numeros	print(numeros[1:howMany])				10 5 16 8 4 2

Prueba de escritorio 3. Número = 0							
Descripción de sentencia	Sentencia	n i	sVali inicial	numeros[]	contador	show i	numeros
Lectura de numero	n <- suppressWarnings(as.integer(n))	0		_			
Validación de numero y mayor a 1	isValid <- !is.na(n) && n > 1	f	alse				
Declaracion de inicial con valor n	inicial <- n		null				
Declaracion de vector con limite de 1000	numeros <- numeric(limit=1000)			0			
Declaración de contador con valor 0	contador <- 0				0		
Condicion de paro con n mayor a 1 y contador menor al limit	n > 1 & contador < limit						
Aumento de contador	contador <- contador + 1				1		
Asignación de valor en el vector	numeros[contador] <- n			[1]=30			
Llamada a la funcion ulam	n <- ulamRegla(n)						
Condicion de residuo par igual a 0	m %% 2 == 0						
Retorna el valor a la mitad	return(m/2)	15					
Retorna el valor al triple más uno	return(3*m + 1)						
Encuentra el minimo entre contador y limit	howMany <- min(contador, limit)						
Cuadro de texto para visualizar grafica	show <- readline("Quieres visualizarla (y/n)? ")					у	
Condición de valor en show igual a 'y'	show == "y"						
Impresión de los numeros	print(numeros[1:howMany])						null

	Prueba de escritorio 1. Número= 88				números			
Descripción de sentencia	Sentencia	n	isValid	inicial		contador	show	numeros
Lectura de numero	n <- suppressWarnings(as.integer(n))	88						
Validación de número y mayor a 1	isValid <- !is.na(n) && n > 1		true					
Declaración de inicial con valor n	inicial <- n			88				
Declaración de vector con límite de 1000	e vector con límite de 1000 números <- numeric(limit=1000)				0			
Declaración de contador con valor 0	contador <- 0					0		
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					1		
Asignación de valor en el vector	numeros[contador] <- n				[1]=88			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	44						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					2		
Asignación de valor en el vector	numeros[contador] <- n				[2] = 44			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	22						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					3		
Asignación de valor en el vector	numeros[contador] <- n				[3] = 22			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)							
Retorna el valor al triple más uno	return(3*m + 1)	11						
Condición de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					4		
Asignación de valor en el vector	numeros[contador] <- n				[4] = 11			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	34						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					5		
Asignación de valor en el vector	numeros[contador] <- n	_			[5] = 34			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)							
Retorna el valor al triple más uno	return(3*m + 1)	17						
Condición de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					6		
Asignación de valor en el vector	números[contador] <- n	_			[6] = 17			
Llamada a la función ulam	n <- ulamRegla(n)							
Condición de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	52						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					7		
Asignación de valor en el vector	números[contador] <- n				[7] = 52			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	26						

Retorna el valor al triple más uno	return(3*m + 1)				
Condición de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			8	
Asignación de valor en el vector	números[contador] <- n		[8] = 26		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor al triple más uno	return(3*m + 1)	13			
Condición de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			9	
Asignación de valor en el vector	números[contador] <- n		[9] = 13		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	40			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			10	
Asignación de valor en el vector	numeros[contador] <- n		[10] = 40		
Llamada a la función ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	20			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			11	
Asignación de valor en el vector	numeros[contador] <- n		[11] = 20		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	10			
Retorna el valor al triple más uno	return(3*m + 1)	1 1			
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador		$\rightarrow$			
				12	
	contador <- contador + 1		[12] = 10	12	
Asignación de valor en el vector	numeros[contador] <- n		[12] = 10	12	
Asignación de valor en el vector Llamada a la funcion ulam	numeros[contador] <- n n <- ulamRegla(n)		[12] = 10	12	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0		[12] = 10	12	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2)	5	[12] = 10	12	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1)	5	[12] = 10	12	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit	5	[12] = 10		
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1	5		12	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n	5	[12] = 10		
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par iqual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n)	5			
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0	5			
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2)				
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1)				
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit			13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < limit		[13] = 5		
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- in n <- ulamRegla(n) m %% 2 -= 0 return(m/2) return(3*m + 1) n > 1 & contador <- limit contador <- contador + 1 numeros[contador] <- n			13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- in n <- ulamRegla(n) n %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador <- limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n)		[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- n n <- ulamRegla(n) n %% 2 == 0	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador <- limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2)		[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor a la mitad	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) return(3*m + 1)	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- limit	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador < n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador <- limit contador <- contador <- limit	16	[13] = 5	13	
Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam Condicion de residuo par igual a 0 Retorna el valor a la mitad Retorna el valor al triple más uno Condicion de paro con n mayor a 1 y contador menor al limite Aumento de contador Asignación de valor en el vector Llamada a la funcion ulam	numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador + 1 numeros[contador] <- n n <- ulamRegla(n) m %% 2 == 0 return(m/2) return(3*m + 1) n > 1 & contador < limit contador <- contador <- conta	16	[13] = 5	13	

Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					16		
Asignación de valor en el vector	numeros[contador] <- n			[16]	=4			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	2						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					17		
Asignación de valor en el vector	numeros[contador] <- n			[17]	=2			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)							
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					18		
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	1						
Retorna el valor al triple más uno	return(3*m + 1)							
Encuentra el minimo entre contador y limit	howMany <- min(contador, limit)							
Cuadro de texto para visualizar grafica	show <- readline("Quieres visualizarla (y/	n)? "	)				у	
Condición de valor en show igual a 'y'	show == "y"							
								88 44 22
								11 34 17 52 26 13
								40 20 10
Impresión de los numeros	print(numeros[1:howMany])							5 16 8 4 2

Descripción de contensio	Contoneia		ioVolid	iniaiaL	numercell	contoder	chove	numeree
Descripción de sentencia	Sentencia		Isvalid	iniciai	numeros[]	contador	snow	numeros
Lectura de numero	n <- suppressWarn ings(as.integer (n))	7						
Validación de numero y mayor a 1	isValid <- !is.na(n) && n > 1		true					
Declaracion de inicial con valor n	inicial <- n			7				
Declaracion de vector con limite de 1000	numeros <- numeric(limit= 1000)				0			
Declaración de contador con valor 0	contador <- 0					0		
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					1		
Asignación de valor en el vector	numeros[conta dor] <- n				[1]=7			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)							
Retorna el valor al triple más uno	return(3*m + 1)	22						
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							
Aumento de contador	contador <- contador + 1					2		
Asignación de valor en el vector	numeros[conta dor] <- n				[2] = 22			
Llamada a la funcion ulam	n <- ulamRegla(n)							
Condicion de residuo par igual a 0	m %% 2 == 0							
Retorna el valor a la mitad	return(m/2)	11						
Retorna el valor al triple más uno	return(3*m + 1)							
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit							

Aumento de contador	contador <- contador + 1			3	
Asignación de valor en el vector	numeros[conta dor] <- n		[3] = 11		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor al triple más uno	return(3*m + 1)	34			
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			4	
Asignación de valor en el vector	numeros[conta dor] <- n		[4] = 34		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	17			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			5	
Asignación de valor en el vector	numeros[conta dor] <- n		[5] = 17		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor al triple más uno	return(3*m + 1)	52			
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			6	

Asignación de valor en el vector	numeros[conta dor] <- n		[6] = 52		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	26			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			7	
Asignación de valor en el vector	numeros[conta dor] <- n		[7] = 26		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	13			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			8	
Asignación de valor en el vector	numeros[conta dor] <- n		[8] = 13		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor al triple más uno	return(3*m + 1)	40			
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			9	
Asignación de valor en el vector	numeros[conta dor] <- n		[9] = 40		

Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	20			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			10	
Asignación de valor en el vector	numeros[conta dor] <- n		[10] = 20		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	10			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			11	
Asignación de valor en el vector	numeros[conta dor] <- n		[11] = 10		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	5			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			12	
Asignación de valor en el vector	numeros[conta dor] <- n		[12] = 5		
Llamada a la funcion ulam	n <- ulamRegla(n)				

Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)				
Retorna el valor al triple más uno	return(3*m + 1)	16			
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			13	
Asignación de valor en el vector	numeros[conta dor] <- n		[13] = 16		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	8			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			14	
Asignación de valor en el vector	numeros[conta dor] <- n		[14] = 8		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	4			
Retorna el valor al triple más uno	return(3*m + 1)				
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit				
Aumento de contador	contador <- contador + 1			15	
Asignación de valor en el vector	numeros[conta dor] <- n		[15] = 4		
Llamada a la funcion ulam	n <- ulamRegla(n)				
Condicion de residuo par igual a 0	m %% 2 == 0				
Retorna el valor a la mitad	return(m/2)	2			

Retorna el valor al triple más uno	return(3*m + 1)						
Condicion de paro con n mayor a 1 y contador menor al limite	n > 1 & contador < limit						
Aumento de contador	contador <- contador + 1				16		
Asignación de valor en el vector	numeros[conta dor] <- n			[16]=2			
Llamada a la funcion ulam	n <- ulamRegla(n)						
Condicion de residuo par igual a 0	m %% 2 == 0						
Retorna el valor a la mitad	return(m/2)	1					
Retorna el valor al triple más uno	return(3*m + 1)						
Encuentra el minimo entre contador y limit	howMany <- min(contador, limit)						
	show <- readline("Quier es visualizarla (y/n)? ")						
Cuadro de texto para visualizar grafica Condición de valor en show igual a 'y'	show == "y"					у	
Impresión de los numeros	print(numeros[ 1:howMany])						7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2
impresion de los números							

#### Conclusión

Se demostró que existen operaciones aplicables a cualquier entero positivo, sea número par o número impar. Donde para los números pares  $f(n) = \frac{n}{2}$ y los números impares f(n) = 3n + 1. Debido a lo anterior se supo demostrar que no existen ciclos cerrados o secuencias divergentes de números impares que no cumplan 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.