



ÁREA ACADÉMICA DE INGENIERÍA EN COMPUTADORES
CE-3102 ANÁLISIS NÚMÉRICO PARA INGENIERÍA

Manual de usuario: manual_FunTras

Estudiantes:

Abarca Aguilar Gabriel 2017110442
Castrillo Muñoz Alejandra 2015155759
Masis Mora Bryan 2013031110
Ramírez Miranda Hernaldo 2016081117

Profesor:

Juan Pablo Soto Quirós
Grupo 01

Fecha de entrega:
18 de octubre, 2020

Tabla de contenido

¿Qué es FunTras?	5
Instalación y Requerimientos	6
Instalación	6
Paso 1	6
Paso 2	6
Paso 3	7
Paso 4	7
NOTA IMPORTANTE	7
Requerimientos	8
Funciones	8
acos_t	8
Uso	8
Formulación Matemática	8
Ejemplo	8
asin_t	9
Uso	9
Formulación Matemática	9
Ejemplo	9
atan_t	9
Uso	9
Formulación Matemática	10
Ejemplo	10
cos_t	10
Uso	10
Formulación Matemática	10
Ejemplo	11
cosh_t	11
Uso	11
Formulación Matemática	11
Ejemplo	11
div_t	12
Uso	12

Formulación Matemática	12
Ejemplo	12
exp_t	13
Uso	13
Formulación Matemática	13
Ejemplo	13
ln_t	14
Uso	14
Formulación Matemática	14
Ejemplo	14
log_t	14
Uso	14
Formulación Matemática	15
Ejemplo	15
power_t	15
Uso	15
Formulación Matemática	15
Ejemplo	16
sin_t	16
Uso	16
Formulación Matemática	16
Ejemplo	16
sinh_t	17
Uso	17
Formulación Matemática	17
Ejemplo	17
sqrt_t	18
Uso	18
Formulación Matemática	18
Ejemplo	18
tan_t	18
Uso	18
Formulación Matemática	19

Ejemplo	19
tanh_t	19
Uso	19
Formulación Matemática	19
Ejemplo	20
pi_t	20
Uso	20
Formulación Matemática	20
Ejemplo	20
root_t	21
Uso	21
Formulación Matemática	21
Ejemplo	21

¿Qué es FunTras?

FunTras es un paquete computacional que provee diecisiete funciones elementales para cálculos matemáticos, en las siguientes secciones se explicará como se realiza la instalación de este paquete para así poder utilizar estas funciones en el lenguaje Octave, las funciones que pertenecen a este paquete son las que se muestran en la siguiente tabla.

Función matemática $f(x)$	Comando para Octave	Función matemática $f(x)$	Comando para Octave
x^{-1}	div_t(x)	e^x	exp_t(x)
$\sin(x)$	sin_t(x)	$\cos(x)$	cos_t(x)
$\tan(x)$	tan_t(x)	$\ln(x)$	ln_t(x)
$\log_a(x)$	log_t(x,a)	a^x	power_t(x,a)
$\sinh(x)$	sinh_t(x)	$\cosh(x)$	osh_t(x)
$\tanh(x)$	tanh_t(x)	\sqrt{x}	sqrt_t(x)
$\sqrt[a]{x}$	root_t(x,a)	$\sin^{-1}(x)$	asin_t(x)
$\tan^{-1}(x)$	atan_t(x)	$\cos^{-1}(x)$	acos_t(x)
π	pi_t()		

Tabla 1. Funciones matemáticas con su correspondiente en Octave

Es importante mencionar que todas las funciones implementadas en este paquete utilizan solamente operaciones elementales suma (+), resta (-), multiplicación (*) y potencia (^). Además de fórmulas matemáticas específicas para la aproximación de algunas funciones.

Instalación y Requerimientos

Instalación

Paso 1

Tener el paquete FunTras.tar.gz almacenado en el ordenador. Como se ve en la imagen 1.

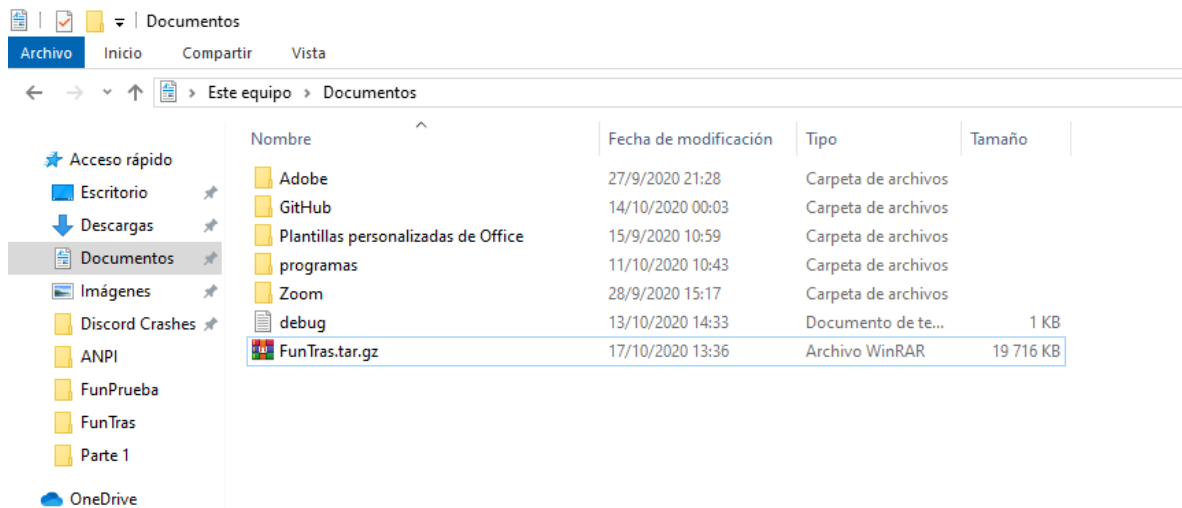


Imagen 1. Paquete almacenado

Paso 2

En Octave (GUI), buscar la ruta en la que se encuentre el paquete.

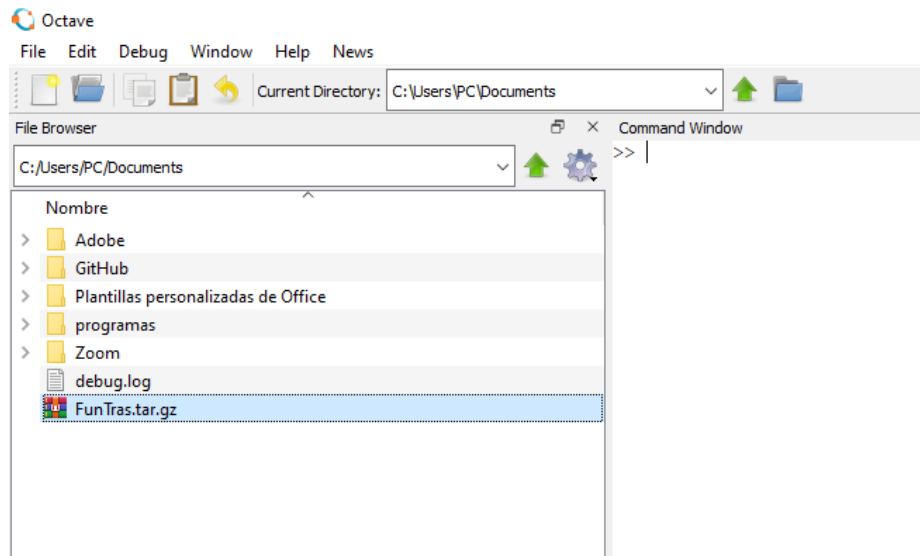


Imagen 2: Ruta del paquete FunTras.tar.gz en Octave (GUI)

Paso 3

Se instala en la consola de comandos de Octave (GUI) escribir `pkg install` + el nombre del paquete, a como se muestra en la imagen 3.

`pkg install FunTras.tar.gz`

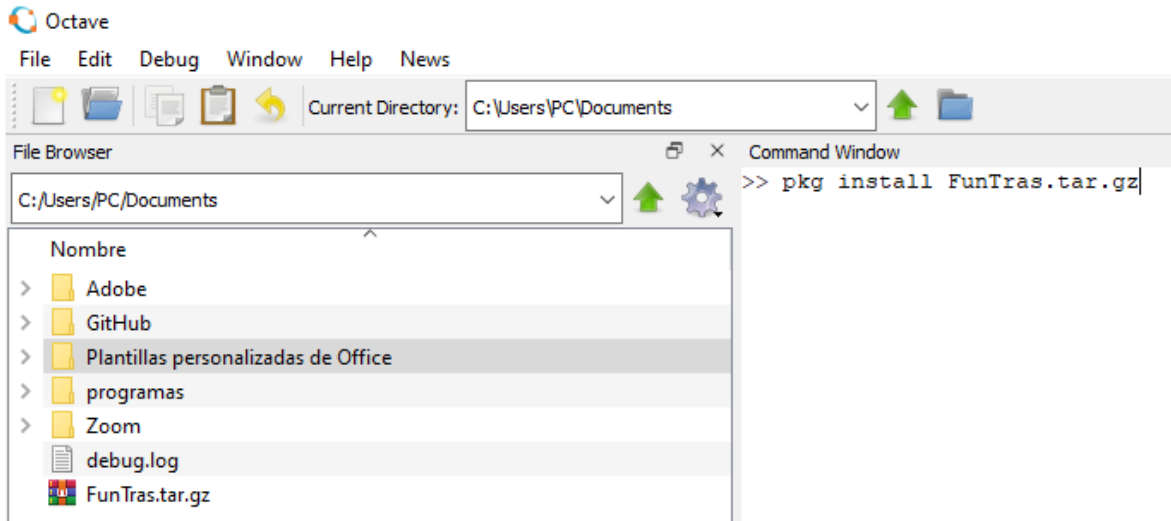


Imagen 3. Instalación del paquete

Paso 4

Se carga el paquete para poder ser utilizado. Utilizando el comando `pkg load funtras` como se muestra en la imagen 4.

`pkg load funtras`

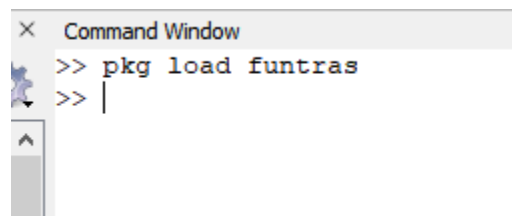


Imagen 4. Cargar el paquete para poder utilizar

NOTA IMPORTANTE

Al cargar el paquete se debe escribir en minúscula

Requerimientos

El principal requisito para la instalación de este paquete de es contar con una versión de Octave igual o mayor a 5.2 ya que esta fue la versión del GNU de Octave en el que se realizó la descripción y cálculo de estas funciones.

Funciones

acos_t

Uso

Esta función calcula el arcocoseno del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`acos_t(a)`

Siendo 'a' el valor del argumento para calcular el arcocoseno.

Formulación Matemática

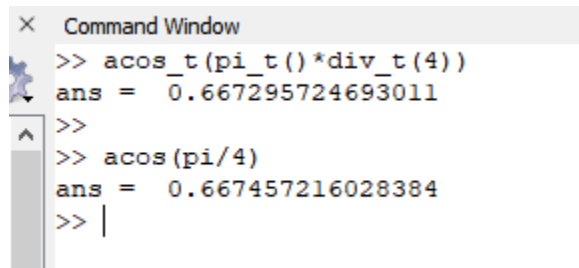
El cálculo del arcocoseno en la implementación viene dada por la siguiente fórmula:

$$\text{acos_t}(a) = \frac{\pi}{2} - \text{asin_t}(a)$$

Con 'a' como el parámetro ingresado en la función.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el arcocoseno de $\pi/4$, de igual forma se calcula con la función `acos` original, para así corroborar el valor.



```
>> acos_t(pi_t()*div_t(4))
ans = 0.667295724693011
>>
>> acos(pi/4)
ans = 0.667457216028384
>> |
```

Imagen 5. Cálculo de `acos_t(pi_t()/4)` y comparado con `acos(pi/4)`

asin_t

Uso

Esta función calcula el arcoseno del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`asin_t(a)`

Siendo 'a' el valor del argumento para calcular el arcoseno.

Formulación Matemática

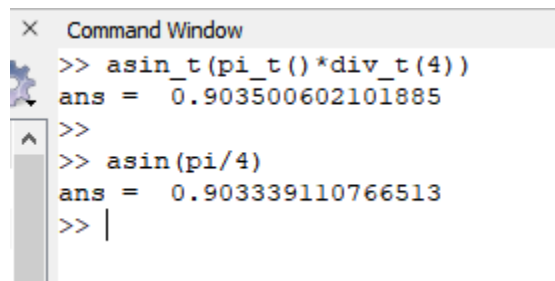
El cálculo del arcoseno en la implementación viene dada por la siguiente fórmula:

$$\text{asin}_t(a) = \sum_{n=0}^k \frac{(2n!)}{4^n (n!)^2 (2n+1)} a^{2n+1}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el arcoseno de $\pi/4$, de igual forma se calcula con la función asin original, para así corroborar el valor.



```
>> asin_t(pi_t()*div_t(4))
ans = 0.903500602101885
>>
>> asin(pi/4)
ans = 0.903339110766513
>> |
```

Imagen 6. Cálculo de `asin_t(pi/4)` y comparado con `asin(pi/4)`

atan_t

Uso

Esta función calcula el arcotangente del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`atan_t(a)`

Siendo 'a' el valor del argumento para calcular el arcotangente.

Se debe tener en cuenta que para el cálculo de la arcotangente se utilizó la Serie de Maclaurin, la cual se limita a valores entre 1 y -1.

Formulación Matemática

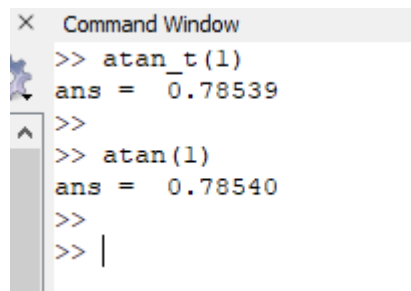
El cálculo del arcotangente a la hora de realizar la implementación se da con la siguiente fórmula:

$$\text{atan_t}(a) = \sum_{n=0}^k (-1)^n \frac{a^{2n+1}}{2n+1}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el arcotangente de 1, de igual forma se calcula con la función atan original, para así corroborar el valor.



```
Command Window
>> atan_t(1)
ans = 0.78539
>>
>> atan(1)
ans = 0.78540
>>
>> |
```

Imagen 7. Cálculo de atan_t(1) y comparado con atan(1)

cos_t

Uso

Esta función calcula el coseno del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`cos_t(a)`

Siendo 'a' el valor del argumento para calcular el coseno.

Formulación Matemática

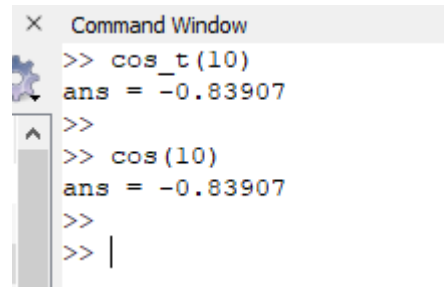
El cálculo del coseno en la implementación viene dada por la siguiente fórmula:

$$\text{cos_t}(a) = \sum_{n=0}^k (-1)^n \frac{a^{2n}}{(2n)!}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el coseno de 10, de igual forma se calcula con la función cos original, para así corroborar el valor.



```
Command Window
>> cos_t(10)
ans = -0.83907
>>
>> cos(10)
ans = -0.83907
>>
>> |
```

Imagen 8. Cálculo de $\cos_t(1)$ y comparado con $\cos(1)$

cosh_t

Uso

Esta función calcula el coseno hiperbólico del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`cosh_t(a)`

Siendo 'a' el valor del argumento para calcular el coseno hiperbólico.

Formulación Matemática

El cálculo del coseno hiperbólico en la implementación viene dada por la siguiente fórmula:

$$\cosh_t(a) = \sum_{n=0}^k \frac{a^{2n}}{(2n)!}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el coseno hiperbólico de 15, de igual forma se calcula con la función cosh original, para así corroborar el valor.

```

X Command Window
>> cosh_t(15)
ans = 1634508.68624
>>
>> cosh(15)
ans = 1634508.68624
>>
>> |

```

Imagen 9. Cálculo de $\cosh_t(15)$ y comparado con $\cosh(15)$

div_t

Uso

Esta función calcula la inversa del valor ingresado, o sea $1/a$, se utiliza con el siguiente comando.

`div_t(a)`

Siendo 'a' el valor que se encontraría en el denominador de la raíz y un 1 en el numerador.

Formulación Matemática

La implementación para esta función es un tanto compleja ya que se debe tomar en cuenta la constante para la precisión relativa de un punto flotante, y con eso se debe implementar la siguiente fórmula:

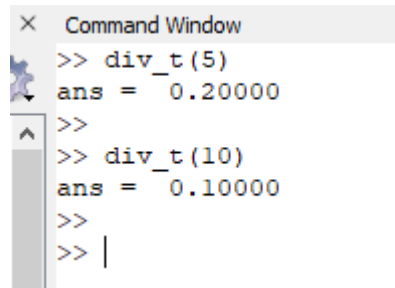
$$\text{div_t}(a) = x_{k+1} = x_k(2 - a \cdot x_k)$$

Con 'a' como el valor ingresado por el usuario y ' x_k ' como valor inicial que varia como se muestra a continuación:

$$x_0 = \begin{cases} \text{eps}^{15} & \text{si } 80! < a \leq 100! \\ \text{eps}^{11} & \text{si } 60! < a \leq 80! \\ \text{eps}^8 & \text{si } 40! < a \leq 60! \\ \text{eps}^4 & \text{si } 20! < a \leq 40! \\ \text{eps}^2 & \text{si } 0! < a \leq 20! \end{cases}$$

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula la división 5 que es un valor de 0.2 y también de 10 que es un valor de 0.1



```
Command Window
>> div_t(5)
ans = 0.20000
>>
>> div_t(10)
ans = 0.10000
>>
>> |
```

Imagen10 . Cálculo de div_t(5) y div_t(10).

exp_t

Uso

Esta función calcula el exponente de base 'e' del parámetro que se le ingrese siendo que se utiliza con el siguiente comando.

exp_t(a)

Siendo 'a' el valor del exponente a calcular para una base 'e'.

Formulación Matemática

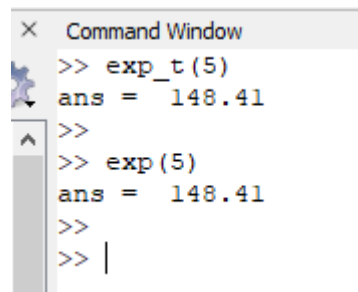
El cálculo de esta función se da gracias a la fórmula expresada a continuación:

$$\exp_t(a) = \sum_{n=0}^k \frac{a^n}{(n)!}$$

Con 'a' como el parámetro ingresado en la función.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el exponente 5 de base 'e', de igual forma se calcula con la función exp original, para así corroborar el valor.



```
Command Window
>> exp_t(5)
ans = 148.41
>>
>> exp(5)
ans = 148.41
>>
>> |
```

Imagen 11. Cálculo de $\exp_t(5)$ y comparado con $\exp(5)$

ln_t

Uso

Esta función calcula el logaritmo natural del parámetro que se le ingrese por lo que se usa con el siguiente comando.

$\ln_t(a)$

Siendo 'a' el valor del argumento para calcular el logaritmo natural.

Formulación Matemática

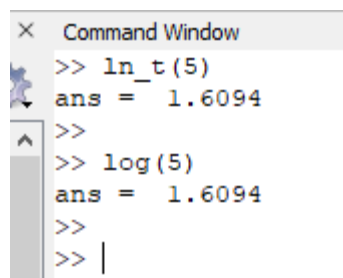
El cálculo de esta función se da gracias a la implementación de la serie:

$$\ln_t(a) = \frac{2(a-1)}{a+1} \sum_{n=0}^k \frac{1}{2n+1} \left(\frac{a-1}{a+1}\right)^{2n}$$

Con 'a' como el parámetro ingresado en la función.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el logaritmo natural 5, de igual forma se calcula con la función log original, para así corroborar el valor.



```
Command Window
>> ln_t(5)
ans = 1.6094
>>
>> log(5)
ans = 1.6094
>>
>> |
```

Imagen 12. Cálculo de $\ln_t(5)$ y comparado con $\log(5)$

log_t

Uso

Esta función calcula el logaritmo en base de un valor ingresado de argumento ingresado de igual manera, se utiliza con el siguiente comando.

$\log_t(x, a)$

Siendo 'x' el argumento al que se le calculará el logaritmo y 'a' siendo la base del logaritmo.

Formulación Matemática

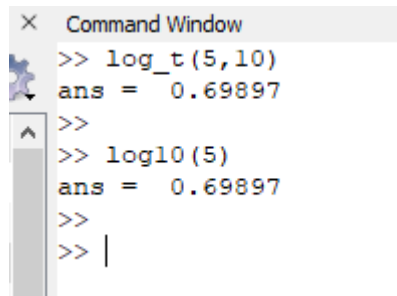
El cálculo de un logaritmo en base 'a' de un argumento 'x' se implementa con la utilización del logaritmo natural previamente calculado, de la siguiente forma se realiza:

$$\log_t(x, a) = \frac{\ln(x)}{\ln(a)} \rightarrow \log_a(x)$$

Con 'x' como argumento al que se le calculará el logaritmo y 'a' siendo la base del logaritmo.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el logaritmo en base 10 con argumento 5, de igual forma se calcula con la función log original, para así corroborar el valor.



```
Command Window
>> log_t(5,10)
ans = 0.69897
>>
>> log10(5)
ans = 0.69897
>>
>> |
```

Imagen 13. Cálculo de log_t(5,10) y comparado con log10(5)

power_t

Uso

Esta función calcula la potencia de un numero ingresado y el valor exponente también ingresado, se utiliza con el siguiente comando.

`power_t(x, a)`

Siendo 'x' el exponente y 'a' siendo la base a la que se le calculará el valor.

Formulación Matemática

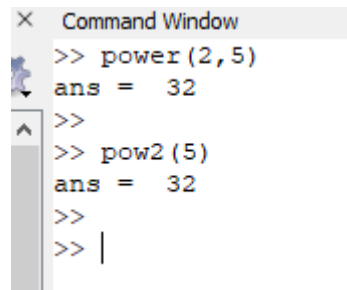
El cálculo de esta potencia se da realizando una multiplicación de iterativa de la base ingresada y se detendrá cuando se cumpla la cantidad de iteraciones que se desea, la cual se puede expresar de la siguiente manera:

$$power_t(x, a) = \prod_{n=1}^x a$$

Con 'x' la cantidad de iteraciones que se deben realizar en la productoria y 'a' siendo el valor a multiplicar.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula 5 elevado a la 5, de igual forma se calcula con la función pow original, para así corroborar el valor.



```

Command Window
>> power(2,5)
ans = 32
>>
>> pow2(5)
ans = 32
>>
>> |

```

Imagen 14. Cálculo de power_t(2,5) y comparado con pow2(5)

sin_t

Uso

Esta función calcula el seno del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`sin_t(a)`

Siendo 'a' el valor del argumento para calcular el seno.

Formulación Matemática

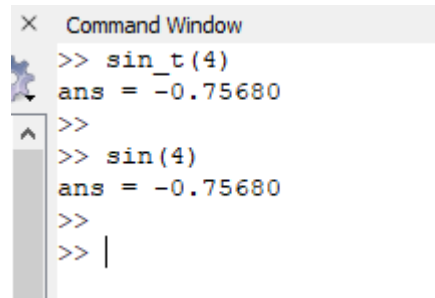
El cálculo del seno en la implementación viene dada por la siguiente fórmula:

$$\text{sen}_t(a) = \sum_{n=0}^k (-1)^n \frac{a^{2n+1}}{(2n+1)!}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el seno de 4, de igual forma se calcula con la función sin original, para así corroborar el valor.



```
Command Window
>> sinh_t(4)
ans = -0.75680
>>
>> sin(4)
ans = -0.75680
>>
>> |
```

Imagen 15. Cálculo de $\sinh_t(4)$ y comparado con $\sin(4)$

\sinh_t

Uso

Esta función calcula el seno hiperbólico del parámetro que se le ingrese por lo que se usa con el siguiente comando.

$\sinh_t(a)$

Siendo 'a' el valor del argumento para calcular el seno hiperbólico.

Formulación Matemática

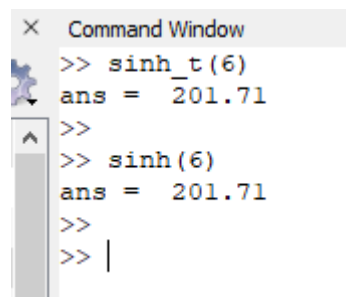
El cálculo del seno hiperbólico en la implementación viene dada por la siguiente fórmula:

$$\sinh_t(a) = \sum_{n=0}^k \frac{a^{2n+1}}{(2n+1)!}$$

Con 'n' con el número de iteraciones y 'a' el valor del parámetro recibido

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el seno hiperbólico de 6, de igual forma se calcula con la función \sinh original, para así corroborar el valor.



```
Command Window
>> sinh_t(6)
ans = 201.71
>>
>> sinh(6)
ans = 201.71
>>
>> |
```

Imagen 16. Cálculo de $\sinh_t(6)$ y comparado con $\sinh(6)$

sqrt_t

Uso

Esta función calcula la raíz cuadrada del parámetro que se le ingrese por lo que se usa con el siguiente comando.

```
sqrt_t(a)
```

Siendo 'a' el valor del argumento para calcular la raíz cuadrada.

Formulación Matemática

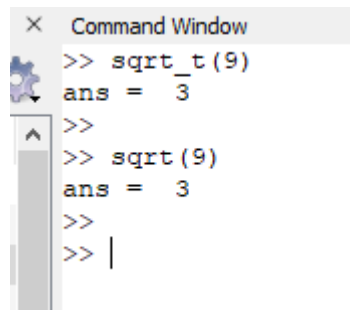
La implementación del cálculo de la raíz cuadrada es en cierta manera, sencilla, ya que la lógica viene dada en la implementación de la función `root_t`, por lo que la raíz cuadrada se puede expresar así:

$$\text{sqrt_t}(a) = \text{root_t}(2, a)$$

Con 'a' como argumento ingresado para calcular su raíz cuadrada y además 'a' > 0.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula la raíz cuadrada de 9, de igual forma se calcula con la función `sqrt` original, para así corroborar el valor.



```
Command Window
>> sqrt_t(9)
ans = 3
>>
>> sqrt(9)
ans = 3
>>
>> |
```

Imagen 17. Cálculo de `sqrt_t(9)` y comparado con `sqrt(9)`

tan_t

Uso

Esta función calcula la tangente del parámetro que se le ingrese por lo que se usa con el siguiente comando.

```
tan_t(a)
```

Siendo 'a' el valor del argumento para calcular la tangente.

Formulación Matemática

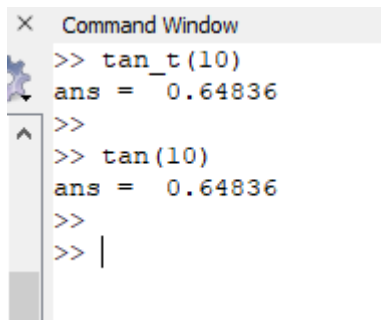
Para el cálculo de la tangente simplemente se debe utilizar las funciones ya implementadas anteriormente, tomando en cuenta que esta es una división del seno entre el coseno; se presenta a continuación:

$$\tan_t(a) = \frac{\sin_t(a)}{\cos_t(a)}$$

Con 'a' como el parámetro ingresado en la función y tomando en cuenta que todas las divisiones implementadas en este paquete se realizan mediante la función `div_t`.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula la tangente de 10, de igual forma se calcula con la función `tan` original, para así corroborar el valor.



```
Command Window
>> tan_t(10)
ans = 0.64836
>>
>> tan(10)
ans = 0.64836
>>
>> |
```

Imagen 18. Cálculo de `tan_t(10)` y comparado con `tan(10)`

`tanh_t`

Uso

Esta función calcula el tangente hiperbólico del parámetro que se le ingrese por lo que se usa con el siguiente comando.

`tanh_t(x)`

Siendo `x` el valor del argumento para calcular el tangente hiperbólico.

Formulación Matemática

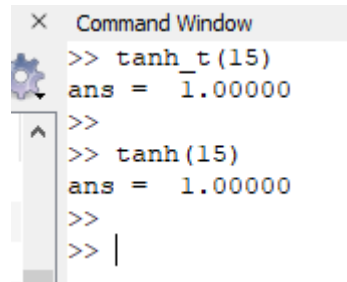
Para el cálculo de la tangente hiperbólica simplemente se debe utilizar las funciones ya implementadas anteriormente, tomando en cuenta que esta es una división del seno entre el coseno; se presenta a continuación:

$$\tanh_t(a) = \frac{\sinh_t(a)}{\cosh_t(a)}$$

Con 'a' como el parámetro ingresado en la función.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el tangente hiperbólico de 15, de igual forma se calcula con la función tanh original, para así corroborar el valor.



```

Command Window
>> tanh_t(15)
ans = 1.00000
>>
>> tanh(15)
ans = 1.00000
>>
>> |
  
```

Imagen 19. Cálculo de $\tanh_t(15)$ y comparado con $\tanh(15)$

pi_t

Uso

Esta función calcula el valor de pi utilizando el siguiente comando.

`pi_t()`

Formulación Matemática

El cálculo del valor aproximado de pi se da mediante la serie de Leibniz dada por la siguiente fórmula:

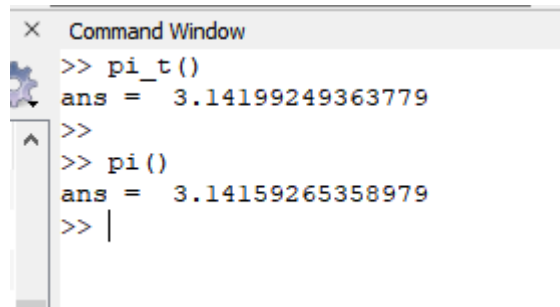
$$\frac{\pi}{4} = \sum_{n=0}^k \frac{(-1)^n}{2n+1}$$

De la cual se despeja el valor de π que dando de la siguiente manera:

$$\pi = \sum_{n=0}^k \frac{4 \cdot (-1)^n}{2n+1}$$

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula el valor de π , de igual forma se calcula con la función pi original, para así corroborar el valor.



```
Command Window
>> pi_t()
ans = 3.14199249363779
>>
>> pi()
ans = 3.14159265358979
>> |
```

Imagen 20. Cálculo de pi_t() y comparado con pi()

root_t

Uso

Esta función calcula la raíz k-ésima de los valores ingresados se utiliza con el siguiente comando.

`root_t(x, a)`

Siendo 'x' el radicando al que se le calculará la raíz y 'a' siendo el índice de la raíz.

Formulación Matemática

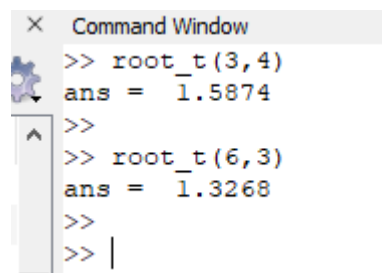
A esta función se le da solución al encontrar el cero de la función $g(a) = x^a - a$ y con eso se puede obtener la siguiente fórmula que da pie a su cálculo:

$$root_t(x, a) = \frac{a}{2} - \frac{\left(\frac{a}{2}\right)^x - a}{\frac{a}{2} \cdot x}$$

Con 'x' el radicando al que se le calculará la raíz y 'a' siendo el índice de la raíz.

Ejemplo

Como ejemplo se puede observar en la siguiente imagen donde se calcula la raíz cuarta de 3, además de otro cálculo la raíz cúbica de 6.



```
Command Window
>> root_t(3,4)
ans = 1.5874
>>
>> root_t(6,3)
ans = 1.3268
>>
>> |
```

Imagen 21. Cálculo de root_t(3,4) y también root_t(6,3)