



Tutorial Gmsh 4.6.0

Operadores y funciones en Gmsh

Steven Vanegas Giraldo

Universidad Nacional de Colombia
Sede Manizales
2020



1. Comentarios y comandos
2. Operadores
 - 2.1. Operadores generales
 - 2.2. Operadores de relación
 - 2.3. Operadores booleanos
3. Funciones
4. Estructuras de los ciclos y condicionales
 - 4.1. Condicional If
 - 4.2. Ciclo For
5. Expresiones útiles
 - 5.1. Lista
 - 5.2. Creación de variables

```
1 // comentario de una linea
2
3 /* comentarios de
4 varias lineas */
```

La constante π se puede obtener con `Pi`.

Cada comando debe terminar en un ;

Operador	Acción	Ejemplo
=	Asignación	$x = 7$
+	Adición	$x = 7 + 5$
-	Sustracción	$x = 7 - 5$
*	Multiplicación	$x = 7 * 7$
/	División	$x = 8 / 2$
%	Módulo	$x = 8 \% 3$
^	Potenciación	$x = 8 ^ 2$
+=	Asigna suma	$x = 7 ; x += 7$
-=	Asigna resta	$x = 7 ; x -= 7$
*=	Asigna multiplicación	$x = 7 ; x *= 7$
/=	Asigna división	$x = 7 ; x /= 7$

Operador	Tipo de relación
<code>==</code>	Igualdad
<code>!=</code>	Desigualdad
<code>></code>	Mayor
<code>>=</code>	Mayor o igual
<code><</code>	Menor
<code><=</code>	Menor o igual

Operador	Operación booleana
!	NOT lógico
& &	AND lógico
	OR lógico

Función	Operación
$\text{Sin}(x)$	Seno de x
$\text{Cos}(x)$	Coseno de x
$\text{Tan}(x)$	Tangente de x
$\text{Asin}(x)$	Seno inverso de x
$\text{Acos}(x)$	Coseno inverso de x
$\text{Atan}(x)$	Tangente inversa de x
$\text{Atan2}(y, x)$	Tangente inversa de y/x
$\text{Sinh}(x)$	Seno hiperbólico de x
$\text{Cosh}(x)$	Coseno hiperbólico de x
$\text{Tanh}(x)$	Tangente hiperbólico de x

Función	Operación
Exp (x)	e^x
Fabs (x)	Valor absoluto de x
Log (x)	Logaritmo natural de x
Log10 (x)	Logaritmo en base 10 de x
Sqrt (x)	Raíz cuadrada de x
Max (x1, x2)	Máximo de 2 argumentos
Min (x1, x2)	Mínimo de 2 argumentos
Rand (x)	Número aleatorio entre 0 y x
Hypot (x1, x2)	$\sqrt{x1^2 + x2^2}$
Floor (x)	Redondea x
Round (x)	Redondea x
Ceil (x)	Redondea x


```
1  If (condicion1)
2      // segmento de codigo cuando condicion1
3      // es verdadera
4
5  Elself (condicion2)
6      // segmento de codigo cuando condicion1
7      // es falsa y condicion2 es verdadera
8
9  ...
10
11 Else
12     // segmento codigo cuando son
13     // falsas las condiciones anteriores
14
15 EndIf
```

```
1 For i In {x1:x2} // itera i desde el valor de x1
2                 // al valor de x2 con un incremento
3                 // uno
4
5     // segmento de código dentro del ciclo
6
7 EndFor
```

```
1 For i In {x1:x2:paso} // itera i desde el valor de
2                       // al valor de x2 con un
3                       // incremento de paso
4                       // paso puede ser positivo
5                       // o negativo
6
7       // segmento de codigo dentro del ciclo
8
9 EndFor
```

Una estructura útil en el manejo de información es la lista, la sintaxis es la siguiente:

`cadena = {elemento1, elemento2, ..., elementon}`

Para acceder a un elemento dentro de una lista se presenta la siguiente sintaxis:

`cadena[#elemento]`

La numeración de los elementos de una lista empieza desde cero.

El operador : simplifica la nomenclatura de indexación {pi : pf : paso}

```
1 // se crea una lista cualquiera
2 lista = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
3 // se obtiene el quinto elemento (5 en este caso)
4 quinto_elemento = lista[4];
5 // indexacion de lista se obtiene --> {2, 4, 6, 8}
6 lista_2 = lista[{1:8:2}];
```

Se puede hacer referencia a una secuencia de etiquetas numéricas de entidades:

{1, 2, 3, 4, 5, 6, 7, 8} → {1:8}

Una estructura útil para la creación de variables en la concatenación de cadenas de texto, este presenta la siguiente sintaxis:

$\text{cadena} \sim \{\text{variable}\} = \text{cadena_variable}$

```
1 For i In {1:3}
2
3     x~{i} = i;
4
5 EndFor
```

```
1 x_1 = 1;
2 x_2 = 2;
3 x_3 = 3;
```
