

## Tema 3: Nociones básicas de MATLAB.

Matrices, variables, números, operadores aritméticos, expresiones, funciones intrínsecas, operadores relacionales y lógicos.

### 3. Nociones básicas de MATLAB

Ingresar matrices.

Para ingresar una matriz, solo hay que respetar algunas convenciones:

- Separar los elementos de una fila con espacios en blanco o comas.
- Usar el punto y coma, ; , para indicar el fin de cada fila.
- Encerrar la lista de elementos con corchetes, [ ].

Por ejemplo, escribiendo en la ventana de comandos:

```
>> A = [ 1 2 3 4; 5 6 7 8; 9 10 11 12; 13 14 15 16 ]
```

MATLAB mostrará la matriz recién ingresada:

A =

```
    1     2     3     4
    5     6     7     8
    9    10    11    12
   13    14    15    16
```

Una vez que se ha ingresado la matriz, automáticamente será recordada en el entorno de trabajo de MATLAB. Podemos referirnos a ella simplemente escribiendo A.

## Expresiones.

Como la mayoría de los otros lenguajes de programación, MATLAB provee expresiones matemáticas, pero estas expresiones involucran matrices. Las expresiones se componen de:

- Números
- Variables
- Operadores
- Funciones

## Números

Para los números se usa la notación decimal convencional, con la opción del punto decimal y se manejan signos más y menos. La notación científica usa la letra e para especificar el factor de potencia de diez. Los números complejos usan indistintamente la i o la j como sufijo para la parte imaginaria.

Algunos ejemplos de números escritos correctamente son:

-99    0.01    1.60210e-20    -3.14159j    3e5i

Los números tienen una precisión de 15 dígitos significativos decimales y un rango entre  $10^{-308}$  y  $10^{+308}$ .

## Números

Los formatos de números admisibles en MATLAB son:

- Notación decimal estándar.

2, +2, -105, 3.1416, -2.5

Nota: MATLAB no hace una distinción entre enteros y reales (todos los números son reales de doble precisión). A pesar de ello, es buen estilo de programación distinguir entre 2 (entero) y 2.0 (real).

- Notación científica

0.31416e+1, 0.000031416e5, 314.16e-2

- Números imaginarios

3\*i 3\*j -3.1416\*i 2.3e4\*i

- Internamente, MATLAB almacena los números en formato de doble precisión (estándar IEEE):

- Precisión: 15 dígitos significativos
- Rango:  $[10^{-308}, 10^{308}]$

## Variables

Las variables no requieren ninguna declaración de tipo o dimensiones. Cuando MATLAB encuentra un nuevo nombre de variable, automáticamente crea la variable y le asigna el espacio de almacenamiento adecuado. Si la variable ya existe cambia su contenido.

Por ejemplo,

```
>> num_estudiantes = 25
```

crea una matriz de 1-por-1 llamada num\_estudiantes y guarda el valor 25 en su único elemento.

## Variables

Una variable se puede interpretar como una unidad de almacenamiento de datos en la memoria del computador. Una variable puede almacenar: números reales, complejos, vectores, matrices, texto y estructuras.

- MATLAB no requiere la declaración del tipo de variable, ni especificar su dimensión.

Nota: Esto es debido a que todas las variables son consideradas como matrices de tamaño variable.

- El nombre de una variable está compuesto por cualquier combinación de letras, números y/o el carácter "\_", con la única restricción de que el primero de ellos debe ser una letra. Además, la longitud del nombre de una variable es de a lo sumo 63 de estos símbolos.
- MATLAB distingue mayúsculas y minúsculas para variables.
- Variables lógicas: Las variables de MATLAB pueden ser utilizadas como variables lógicas:
  - Un valor igual a 0 equivale a falso.
  - Un valor distinto de 0 equivale a verdadero.

## Variables

Precisión y formatos: Por defecto Matlab usa el formato corto para desplegar valores numéricos, pero se pueden usar otros

>> format long (14 cifras significativas)

>> format short (5 cifras significativas)

>> format short e (notación exponencial corta)

>> format long e (notación exponencial larga)

>> format rat (aproximación racional)

**Obs.** Los cálculos en Matlab siempre se efectúan en precisión doble. El comando format sólo actúa sobre lo que se muestra en pantalla.



## Variables y funciones predefinidas

ans	Posee el último valor ingresado, que no fue asignado a variable
i	Unidad imaginaria (valor: $\sqrt{-1}$ )
j	Unidad imaginaria
pi	Número $\pi$ (3.14159265358979...)
eps	Precisión relativa de real (doble precisión) (valor: $2^{-52}$ )
realmax	Valor máximo de un real (doble precisión) (valor: $(1+eps)*2^{1023}$ )
realmin	Valor absoluto mínimo de un número real (valor: $2^{-1022}$ )
Inf	Infinito (ejemplo: $1/0$ . MATLAB opera con infinitos)
NaN	Not-a-Number (ejemplo: $0/0$ . Operaciones con NaN generan NaN)

## Operaciones entre variable reales

### Operadores elementales o básicos

operación	símbolo	ejemplo
adición, $a + b$	+	$6 + 9$
sustracción, $a - b$	-	$5 - 8$
multiplicación, $a * b$	*	$-2.34 * 4$
división, $a / b$	/	$0.98 / 4.56$
potenciación, $a^b$	^	$2.34 ^ 9.5$

## Operaciones entre variable reales

Regla de precedencia en una expresión:

Las operaciones se ejecutan respetando el orden de precedencia siguiente:

$\wedge, * \text{ y } /$  ,     $+$  y  $-$   
mayor                    a                    menor

En una expresión con operadores de igual nivel de precedencia, la evaluación se realiza de izquierda a derecha.

El uso de paréntesis puede alterar este orden.

Ejemplos:

$$5 + 6 / 2 = 8 \quad \text{y} \quad (5 + 6) / 2 = 5.5$$

$$4 * 25 + 6 * 52 = 412 \quad \text{y} \quad 4 * (25 + 6 * 52) = 1348$$

$$2 / 7^3 = 0.0058 \quad \text{y} \quad (2 / 7)^3 = 0.0233$$

## Funciones sobre variables reales

Sobre variables reales existen muchas funciones científicas en MATLAB

```
>> help elfun
```

### Trigonometric.

**sin** - Sine.  
**sinh** - Hyperbolic sine.  
**asin** - Inverse sine.  
**asinh** - Inverse hyperbolic sine.  
**cos** - Cosine.  
**cosh** - Hyperbolic cosine.  
**acos** - Inverse cosine.  
**acosh** - Inverse hyperbolic cosine.  
**tan** - Tangent.  
**tanh** - Hyperbolic tangent.  
**atan** - Inverse tangent.  
**atan2** - Four quadrant inverse tangent.  
**atanh** - Inverse hyperbolic tangent.  
**sec** - Secant.  
**sech** - Hyperbolic secant.  
**asec** - Inverse secant.  
**asech** - Inverse hyperbolic secant.  
**csc** - Cosecant.  
**csch** - Hyperbolic cosecant.  
**acsc** - Inverse cosecant.  
**acsch** - Inverse hyperbolic cosecant.  
**cot** - Cotangent.  
**coth** - Hyperbolic cotangent.  
**acot** - Inverse cotangent.  
**acoth** - Inverse hyperbolic cotangent.

### Exponential.

**exp** - Exponential.  
**log** - Natural logarithm.  
**log10** - Common (base 10) logarithm.  
**log2** - Base 2 logarithm and dissect floating point number.  
**pow2** - Base 2 power and scale floating point number.  
**sqrt** - Square root.  
**nextpow2** - Next higher power of 2.

### Rounding and remainder.

**fix** - Round towards zero.  
**floor** - Round towards minus infinity.  
**ceil** - Round towards plus infinity.  
**round** - Round towards nearest integer.  
**mod** - Modulus (signed remainder after division).  
**rem** - Remainder after division.  
**sign** - Signum.

## Funciones sobre variables reales

Ejemplo:

	fix	floor	round	ceil
-2.2	-2	-3	-2	-2
3.1	3	3	3	4
3.9	3	3	4	4
0.9	0	0	1	1

redondeo  
hacia 0

redondeo  
hacia  $-\infty$

redondeo  
hacia  
entero más  
próximo

redondeo  
hacia  $+\infty$

**Obs.** round(-3.5) da en como resultado -4, mientras que round(3.5) da 4

## Funciones sobre variables reales

Definición:

$\text{rem}(x,y)$  es  $x - n \cdot y$  donde  $n = \text{fix}(x/y)$

$\text{mod}(x,y)$  es  $x - n \cdot y$  donde  $n = \text{floor}(x/y)$

Ejemplo:

$\text{mod}(8,3) \rightarrow 2, \quad \text{rem}(8,3) \rightarrow 2,$

$\text{mod}(-8,3) \rightarrow 1, \quad \text{rem}(-8,3) \rightarrow -2$

---

Definición:

$\text{sign}(x)$  retorna 1 si  $x$  es mayor que 0, 0 si  $x$  es 0 y -1 si  $x$  es menor que 0.

Ejemplo:

$\text{sign}(-3) \rightarrow -1, \quad \text{sign}(8.1) \rightarrow 1, \quad \text{sign}(0) \rightarrow 0.$

## Variables complejas

Están definidas por su parte real y su parte imaginaria

Ejemplo:

```
>> a = 1 - 2i, b = 3 * (2 - sqrt(-1) * 3)
```

a → 1 - 2i

b → 6 - 9i

Los operadores de adición, sustracción, multiplicación y división de números complejos usan los mismos símbolos que para variables reales.

---

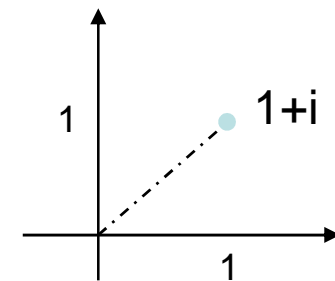
## Funciones sobre variables complejas:

abs - módulo.  
angle - ángulo.  
complex - construye un complejo a partir de (re, im).  
conj - conjugado complejo.  
imag - parte imaginaria.  
real - parte real.

Ejemplo:

```
>> angle(1 + i)  
ans → 0.7854
```

```
>> abs(1 + i)  
ans → 1.4142
```



## Operaciones entre variable reales

### Operadores relacionales

símbolo	operación
>	mayor que
>=	mayor o igual que
<	menor que
<=	menor o igual que
==	igual que
~=	distinto que

Ejemplos:

- $1 + 1 > 2$ , `ans`  $\rightarrow$  0
- $(6.8 + 5 < 45) * 3$ , `ans`  $\rightarrow$  3



## Operaciones entre variable reales

### Operadores lógicos

operador		descripción
a & b	conjunción "y"	retorna 1 si las expresiones a y b son ambas verdaderas, y 0 en cualquier otro caso
a   b	disyunción "o"	retorna 0 si las expresiones a y b son ambas falsas, y 1 en cualquier otro caso
~ a	negación	Negado de a, retorna 1 si a es 0, 0 en caso contrario

### Ejemplos:

- $-2 \sim = 1$ ,     $\text{ans} \rightarrow 1$
- $\sim 1$ ,     $\text{ans} \rightarrow 0$
- $(36 < 4) | (1 > 2)$ ,     $\text{ans} \rightarrow 0$

## Operaciones entre variable reales

### Tablas de comportamiento de los operadores lógicos

negación

A	$\sim A$
0	1
1	0

conjunción

A	B	A & B
0	0	0
0	1	0
1	0	0
1	1	1

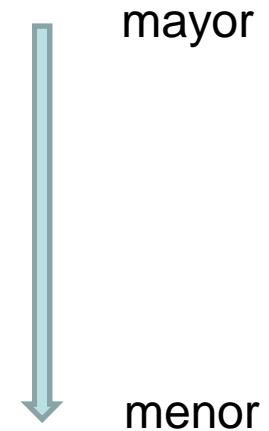
disyunción

A	B	A   B
0	0	0
0	1	1
1	0	1
1	1	1

## Operaciones entre variable reales

Regla de precedencia en una expresión para los operadores (mayor a menor)

1. paréntesis ( ), potencia ^
2. signo positivo +, signo negativo -, negación ~
3. multiplicación \*, división /
4. adición +, sustracción -
5. menor que <, menor o igual que <=, mayor que >,  
mayor o igual que >=, igual que ==, distinto que ~=
6. y &
7. o |



- Ejemplos:
- $1 + 1 > 2$ , ans  $\rightarrow$  0
  - $-2 \sim= 1$ , ans  $\rightarrow$  1
  - $1 + (6.8 + 5 < 45) * 3$ , ans  $\rightarrow$  4
  - $\sim(1 + (6.8 + 5 < 45) * 3)$ , ans  $\rightarrow$  0
  - $\sim(1 + (6.8 + 5 < 45))$ , ans  $\rightarrow$  0
  - $36 < 4 | (1 > 2 ^ 2 == 2)$ , ans  $\rightarrow$  0