

Matlab/Octave: Bestiario de comandos y sentencias

A lo largo de este documento, x e y serán vectores fila o columna, z un número complejo, y A una matriz.

Espacio de trabajo: básicos	
Ctrl + C	Aborta la operación o sentencia actual en la línea de comandos
clc	Limpia la ventana de comandos
clear	Borra todas las variables
clear x A	Borra las variables x y A
diary 'fichero.txt'	Registra en un fichero lo que se hace en la ventana de comandos
diary off	Para el registro
diary on	Reanuda el registro
save fichero	Guarda las variables definidas en el fichero
save fichero x A	Guarda las variables x y A en el fichero
load fichero	Carga las variables almacenadas en el fichero
pwd	Muestra la ubicación del directorio de trabajo actual
dir	Muestra el contenido del directorio de trabajo actual
cd carpeta	Permite acceder a una carpeta del directorio de trabajo actual
help comando	Abre la documentación del comando
lookfor 'texto'	Busca el texto en la documentación de comandos
...	Conecta una misma sentencia escrita en dos líneas seguidas de código
comando;	La ";" suprime la salida del comando
nombreprograma	Ejecuta <i>nombreprograma.m</i>
tic, sentencias, toc	Devuelve el tiempo de ejecución total de las sentencias

Espacio de trabajo: formato y salida	
format short	Muestra n^{os} con 4 decimales
format short e	Muestra n^{os} con 4 decimales en notación exponencial
format long	Muestra n^{os} con 15 decimales
format long e	Muestra n^{os} con 15 decimales en notación exponencial
format rat	Muestra n^{os} en formato racional
% Esto es un comentario	Comentarios
disp('texto')	Muestra el texto
disp(x)	Muestra el contenido de la variable x

Constantes numéricas	
pi	$\pi \simeq 3.1415926535897$
i ó j	Unidad imaginaria $\sqrt{-1}$
Inf	Infinito
NaN	"No es un número" (p.ej., 0/0)
eps	Precisión relativa de máquina en doble precisión (por defecto, $2.2204 \cdot 10^{-16}$)
realmax	Nº positivo más grande en doble precisión, $1.7977 \cdot 10^{308}$
realmin	Nº positivo más pequeño en doble precisión $2.2251 \cdot 10^{-308}$

Operaciones aritméticas y funciones básicas	
1.349	Los decimales de un real se definen CON EL PUNTO "." NO con comas o tildes
3+4, 7*4, 2-6	Suma, producto y resta
8/3, 3\8	División por la derecha y por la izquierda
3^7	Calcula la potencia 3^7
rem(17,3)	Resto de la división de 17 entre 3
sqrt(5)	Calcula la raíz cuadrada $\sqrt{5}$
log(3)	Calcula el logaritmo neperiano $\ln(3)$
log10(100)	Calcula el logaritmo $\log_{10}(100)$
abs(-5)	Calcula el valor absoluto $ -5 $
sin(5*pi/3)	Calcula el seno $\sin(5\pi/3)$
cos(-pi/3)	Calcula el coseno $\cos(-\pi/3)$
exp(3)	Calcula la exponencial e^3

Números complejos	
z=1-2*i ó z=complex(1,-2)	Crea el número complejo $z = 1 - 2i$ (a partir de las partes real e imaginaria)
abs(z)	Módulo (valor absoluto) de z
angle(z)	Argumento de z
z'	Conjugado de z
real(z)	Parte real de z
imag(z)	Parte imaginaria de z
isreal(z)	Devuelve 1 si z es real, 0 si no

Definiendo variables básicas	
a = 3	Define la variable a como 3
b = 4.321	Define la variable b como 4.321
c = 'texto'	Define en c una cadena de caracteres con el texto
cond = logical(1)	Define en $cond$ el valor lógico 1 (<i>true</i>)

Ejemplo: valores lógicos y operaciones	
a = 10;	% Asignamos a "a" el valor de 10
a == 5	% Comprobamos si a es igual a 5
false	
~(a == 5)	% Comprobamos la negación de lo anterior
true	
a == 10	% Comprobamos si a es igual a 10
true	
a >= 5	% Comprobamos si a es mayor o igual a 5
true	
a < 11	% Comprobamos si a es menor que 11
true	
a ~= 4	% Comprobamos si a es no igual a 4
true	
a > 1 && a ~= 10	% Comprobamos si a es mayor que 1 Y % no igual a 10
false	
a > 1 a ~= 10	% Comprobamos si a es mayor que 1 O % no igual a 10
true	
xor(a == 10, a < 100)	% Si a es 10 O (exclusivo) % menor que 100
false	

Vectores y matrices: generación y acceso	
x = [1, 2, 3] ó x = [1 2 3]	Define x como el vector fila [1,2,3]
x = [1; 2; 3] ó x = [1, 2, 3]'	Define x como el vector columna [1,2,3] ^t
7:15	Vector fila con 7, 8, ..., 14, 15
1.1:0.2:3.3	Vector fila con 1.1, 1.3, ..., 3.3
linspace(2, 6.5, 100)	Genera un vector fila con 100 componentes equiespaciadas entre el 2 y el 6.3
A = [1, 2, 3, 4; 5, 6, 7, 8; 9, 10, 11, 12]	Define A como una matriz 3×4
x(2:12)	Del 2º al 12º elemento de x
x(2:end)	Del 2º al último elemento de x
x(1:2:end)	El 1º, 3º, 5º,...hasta el último elemento de x .
A(3,4)	El elemento de la 3ª fila y la 4ª columna de A
A(3,:)	La 3ª fila de A
A(:,4)	La 4ª columna de A
A(2, 1:5)	Del 1º al 5º elemento de la 2ª fila
A([1,3],4)	Los elementos de la 1ª y 3ª fila que se encuentran en la 4ª columna
A(:)	La matriz A vista como vector columna (con los elementos en orden columna)

Matlab/Octave: Bestiario de comandos y sentencias

Vectores y matrices: composición y borrado

[A ; B]	Matriz compuesta por las filas de A sobre las filas de B (con mismo n° de columnas)
[A , B]	Matriz compuesta por las columnas de A seguidas de las de B (con mismo n° de filas)
[3,v; c d]	Matriz compuesta por 3 seguido de la fila de v , sobre las columnas de c seguidas de las de d
A = []	Borra todos los elementos de A
x(4) = []	Elimina la 4ª componente de x
A(3,:) = []	Elimina la 3ª fila A

Operaciones de vectores y matrices

3 * x	Multiplifica cada elemento de x por 3
x + 2	Suma 2 a cada elemento de x
x + y	Suma elemento a elemento los vectores x e y
A * y	Producto de una matriz y un vector
A * B	Producto (matricial) de dos matrices $A \cdot B$
A .* B	Producto (elemento a elemento) de dos matrices
A ^ 3	La matriz (cuadrada) A elevada a la 3ª potencia
A .^ 3	La matriz con los elementos de A elevados a la 3ª potencia
A'	Traspuesta de A
inv(A)	Inversa de A
A / 3	Divide cada elemento de A por 3
3 ./ A	Devuelve la matriz donde cada elemento es 3 dividido por el correspondiente de A
A / B	Devuelve $A \cdot B^{-1}$
A \ B	Devuelve $A^{-1} \cdot B$
A ./ B	División (elemento a elemento) de dos matrices

Funciones auxiliares de vectores y matrices

length(x)	Nº de componentes de x
size(A)	Tamaño de A
[m,n]=size(A)	Asigna a m el nº de filas y a n el nº de columnas de A
sum(x)	Suma todos los elementos de x
sum(A)	Vector de sumas de cada columna de A
prod(x)	Multiplifica todos los elementos de x
prod(A)	Vector de productos de cada columna de A
sort(x)	Ordena ascendentemente los elementos de x
sort(A)	Ordena ascendentemente de forma independiente cada columna de A
max(x)	Valor máximo de x
max(A)	Vector con el máximo de cada columna de A
min(x)	Valor mínimo de x
min(A)	Vector con el mínimo de cada columna de A

Ejemplo: estructura básica de un programa

```
% CALCULO DEL AREA DE UN CIRCULO

% Entrada de datos
r = input('Introduce el radio del círculo: ');

% Algoritmo
A = pi*r^2;

% Salida de datos
fprintf('El area del círculo es %.3f \n', A);
```

Entrada y salida de datos

a=input('Introduce dato:')	Saca en pantalla el texto de entrada de dato y se lo asigna a a al presionar Enter
c=input('¿Nombre?', 's')	Asigna la cadena de caracteres introducida por usuario
disp(A) ó disp('texto')	Muestra A o texto
fprintf('Es a=%f \n ', a)	Escribe en pantalla el texto combinado con el dato de a
c=sprintf('Es a=%f \n ', a)	Almacena en c la cadena de caracteres del texto combinado con el dato de a

Formato de salida de datos

%f	Formato en coma flotante (escribe con 6 decimales)
%d	Formato como enteros, lógicos,...
%s	Formato de cadenas de caracteres
%-6.3f	Salida de datos en coma flotante, justificado a izqda. (con -), con 6 caracteres mínimos reservados para escritura, escrito con 3 decimales
%4.2d	Salida de datos de enteros etc, justificado a dcha. (sin -), con 4 caracteres mínimos reservados para escritura, con un mínimo de 2 dígitos
%7s	Salida de cadenas de caracteres, justificado a dcha. (sin -) con mínimo de anchura de 7 caracteres

Generación de elementos aleatorios

rand()	Genera nº aleatorio equiprobable en el intervalo ABIERTO (0,1)
rand(3)	Genera una matriz 3×3 de nºs aleatorios equiprobables en (0,1)
rand(4,2)	Genera una matriz 4×2 de reales aleatorios equiprobables en (0,1)
2+13*rand()	Genera nº aleatorio equiprobable en (2,15)
fix(1+6*rand())	Genera nº ENTERO aleatorio equiprobable en el intervalo CERRADO [1,6]

Ejemplos: Sentencias condicionales

```
n=input('Introduce un entero: ');
if n<0 % caso n negativo
    disp('el número es negativo')
elseif n==0 % caso n cero
    disp('el número es cero')
else % caso restante
    disp('el número es positivo')
end
```

Ejemplos: Bucle for (I)

```
for i = 1:10
    disp(i);
end
```

Ejemplos: Bucle for (III)

```
% Bucle con paso no trivial
for i=1.5:0.1:2
    disp(i)
end
```

Ejemplos: Bucle for (III)

```
% Bucle sobre un vector predefinido
for i=[4,1,1,-2,0.4]
    disp(i)
end
```

Ejemplos: Bucle while (I)

```
% Bucle while como un for
i = 0;
while i < 7
    disp(i);
    i = i + 1;
end
```

Ejemplos: Bucle while (II)

```
% Debes introducir un dato no vacío...
x=input('Introduce un número: ');
while isempty(x)
    x=input('Debes introducir un número: ');
end
fprintf('Has introducido el %f \n',x)
```