

Medios de Transmisión

Práctica 0

Introducción a MATLAB

1 Introducción a Matlab

Matlab es un entorno para el desarrollo de computaciones y visualizaciones numéricas de alto rendimiento. El elemento de datos básico es la *matriz*. Matlab asigna tamaño a las variables de forma dinámica.

2 Características básicas

Matemáticas simples

Matlab permite hacer las operaciones matemáticas simples como si fuese una calculadora. Los operadores aritméticos son: +, -, ~, \ ó /, ^

Ejemplos:

```
>>a=4*3^2+1 (a=37)
```

```
>>b=(4*3)^2+1 (b=145)
```

```
>>c=3^2-5-6/(3*2) ; (c=3, pero al poner ";" el resultado no es mostrado por pantalla)
```

El *Workspace* de Matlab

Matlab recuerda la historia de los comandos tecleados hasta el momento, así como los valores de las variables que se han creado. Tiene reservado un conjunto de variables especiales entre las que están: ans, pi, inf, j=i= $\sqrt{-1}$ etc.

- Ver la historia de los comandos anteriormente tecleados:

Ctrl-N: comando inmediatamente anterior

Ctrl-P: comandos anteriores

Ctrl-b: cursor izquierdo

Ctrl-f: cursor derecho

- Ver la lista de variables definidas: >>who
- Borrar una variable: >>clear (*lista-variables*)
- Pedir ayuda *on line*: >>help (comando)

3 Características científicas

Funciones matemáticas de interés

abs(), cos(), exp(), imag(), log(), log10(), real(), sin(), sqrt(), tan(), etc.
Ver con el comando **help** que devuelve cada una de estas funciones.

Números complejos

De la forma: $\mathbf{a+bi}$, donde \mathbf{a} es la parte real, \mathbf{b} la parte imaginaria e \mathbf{i} es $\sqrt{-1}$.

Ejemplos:

```
>>c1=1-2i=1-2j=1.0000-2.0000i
>>c2=sqrt(-2)=0+1.4142i
```

4 Arrays y Matrices

Definición y direccionamiento de arrays

Ejemplos:

```
>>a= [ 1 2 3]
>>b=sin(a)
>- c=1:4 (c=[1 2 3 4])
>>d=4 : -1 : 1 (d=[4 3 2 1])
```

Ejemplos:

```
>>a(2) (devuelve el valor en la posición 2 del array a)
>>c(1 : 3) (devuelve los valores desde la posición 1 a la 3 del vector c)
>>c(3)=7 (asigna el valor 7 al elemento de la posición 3)
```

Operaciones con arrays y escalares

Ejemplos :

```
>>a= [2 5 1]
>>a-2 (ans= 0 3 -1)
>>a*4 (ans=8 20 4)
```

Ejemplos:

```
>>a=[2 5 1]
>>b=[4 8 2]
>>a+b (ans=6 13 3)
>>a*b (Error, ya que el * es para la multiplicación de matrices)
>>a .*b (ans=8 40 2) (multiplicación elemento a elemento)
>>b ./a (ans=2 1.6 2) (división elemento a elemento)
>>a.^2 (ans=4 25 1) (potencia elemento a elemento)
```

Ejemplos:

```
>>length(a) (devuelve el número de elementos del vector a)
>>a' (devuelve la traspuesta de a)
```

Definición y direccionamiento de matrices

Ejemplos :

```
>>A= [a;4 3 9] (crea una matriz de 2x3)
>>B=[1 3 5;7 9 11;13 15 17] (crea una matriz de 3x3)
```

Ejemplos:

```
>>B(2,3) (el elemento de la segunda fila y tercera columna)
>>B (:, 2) (el valor en la segunda columna de todas las filas)
>>B (:, :) (la matriz completa)
>>B(1:2,3) (el valor en la columna 3 desde la fila 1 hasta la 2)
```

Operaciones con matrices

Ejemplos :

```
>>B/2 (divide cada elemento de la matriz por 2)
>>B+C (suma de dos matrices de iguales dimensiones)
>>B*C (multiplica dos matrices de dimensiones adecuadas)
>>size(B) (devuelve el número de filas y columnas de la matriz B)
```

Matrices especiales

Ejemplos :

```
>>zeros(3) (matriz 3x3 de todo ceros)
>>ones(2,4) (matriz 2x4 de todo unos)
>>rand(2,1) (matriz 2x1 de números aleatorios uniformemente distribuidos)
>>randn(2) (matriz 2x2 de números aleatorios con distribución normal)
```

5 Funciones gráficas

Ejemplos :

```
>>plot (a,b) (a son las coordenadas del eje x, y b las del eje y)
>>plot (a,b, ' * ' ) (coloca un * en cada punto determinado por a y b)
>>stem(a,b) (genera una gráfica con valores discretos)
>>title() (pone título a la gráfica)
>>xlabel() (pone comentario al eje de las x)
>>ylabel() (pone comentario al eje de las y)
>>axis() (cambia la escala de los ejes de coordenadas)
```

6 Operaciones lógicas y relacionales

- Los operadores lógicos son: **&**, **|**, **~**

- Los operadores relacionales son: **<**, **≤**, **>**, **≥**, **==**, **~=**

Devuelven 1 si el resultado de la evaluación es true y 0 en otro caso.

Ejemplos :

```
>>a=i:8
>>(a > 2)&(a < 6) (devuelve 0 0 1 1 1 0 0 0)
```

7 Control de Flujo

For

```
>>for i=1:10,a(i)=i;end
```

```
>>for i=1:m
```

```
    for j=1:n
```

```
        a(i,j) =i+j ;
```

```
    end
```

```
end (bucles for anidados)
```

If-Else-End

```
>>if n < 0
    a=-1;
elseif n==0
    a=0;
else
    a=1;
end
```

While

```
>>while a > 1
    a=a-1;
end
```

8 Ficheros

Se pueden introducir comandos en un fichero de texto con extensión *.m* y más tarde abrir ese fichero desde Matlab y evaluar los comandos introducidos en él exactamente igual que si se hubiesen tecleado en el **prompt** de Matlab.

Existen dos tipos de ficheros;

1. *Scripts*: son ficheros de comandos. Las variables que se definan en ellos quedan en el entorno.
2. *Funciones*: ficheros que contienen funciones que podrán ser invocadas desde el prompt de Matlab. Pueden recibir parámetros y devolver valores. Las variables definidas son locales a la función. La primera línea ha de ser la declaración de la función.

Ejemplo:

```
function y=mifuncion(x)
% comentarios

    (sentencias)
```

```
y=valor;
```

Ejercicio: Construir una función llamada *factorial* que devuelva como resultado el factorial del número entero que se le pase como parámetro.