

Práctica 1.- Introducción a Matlab y Números Complejos.

1. Introducción

Matlab es un programa que permite realizar cálculos (numéricos y simbólicos), representaciones gráficas, programación de algoritmos, ...

Al trabajar en Matlab hay que tener en cuenta que el programa distingue entre mayúsculas y minúsculas, es decir, no es lo mismo escribir `>> x = 1` que `>> X = 1`, considerará x y X como dos cosas diferentes.

Al escribir en Matlab, las líneas que terminen en `;` se ejecutarán pero no se mostrará el resultado. Por ejemplo, si escribimos `>> x = 2 * 3 + 1;` no nos mostrará el resultado, aunque si posteriormente escribimos `>> x` nos dará como resultado 7.

El comando `>> clear all` borra todas las variables con las que se ha trabajado hasta ahora. Si escribimos `>> clear x` borrará sólo la variable x .

El comando de ayuda de Matlab es `>> help`. Si escribimos dicho comando seguido del nombre de otro comando o función, nos dará información sobre ello. Por ejemplo, al teclear `>> help cos`, nos proporcionará información sobre cómo funciona la función coseno y una lista de funciones relacionadas.

Utilizamos la orden `>> diary nombredelarchivo.txt` para guardar (en un archivo de texto dentro del directorio actual) lo que vayamos escribiendo en cada sesión. Podemos añadir los comentarios que necesitemos anteponiendo el símbolo `%`, así podremos elaborar nuestros propios apuntes de Matlab. Si queremos dejar de guardar los comandos y salidas en el archivo de texto creado, usaremos el comando `>> diary off`.

Las operaciones básicas en Matlab son:

- La suma y la diferencia: `+` y `-`

Ejemplos:

- $3 + 5$
- $3 - x$

- El producto y el cociente: `*` y `/`

Ejemplos:

- $3 * 2$
- $4/3$

- Las potencias: `^`

Ejemplos:

- $2 \wedge 5$
- $3 \wedge (1/2)$

Además, Matlab tiene predefinidas algunas variables y funciones:

- $\text{sqr}(x)$ nos da la raíz cuadrada de x .
- $\sin(x)$, $\cos(x)$, $\tan(x)$, ... nos dan las razones trigonométricas de x .
- $\exp(x)$ nos da e^x . El número e no está definido previamente, así que para obtenerlo habría que escribir $\exp(1)$.
- $\log(x)$ nos da el logaritmo neperiano de x . Para calcular $\log_a(x)$, hay que aplicar las transformaciones habituales: $\log_a(x) = \frac{\ln(x)}{\ln(a)}$.
- $\text{abs}(x)$ nos da el valor absoluto de x .
- π es el número π .
- Inf es ∞ .
- NaN indica indeterminación. (Not a Number).
- eps nos da el número más pequeño que conoce Matlab.

Ejercicio.- Realiza las siguientes operaciones usando Matlab:

- (a) $\sqrt{3^2 + 1} - \sin(\pi)$
- (b) $e^{2^2+1} + \frac{3}{2}$
- (c) $\log_3(9) + \ln(e) + |\log_{10}(0,1)|$
- (d) $\frac{3^2+5}{\frac{1}{2}+2}$
- (e) $\sqrt[5]{\sqrt{3} + 2}$

En las salidas de Matlab se puede controlar el formato numérico, aunque esto no tiene nada que ver con la precisión con la que se realizan los cálculos de forma interna. Si queremos que se muestren en pantalla los resultados con sólo 5 dígitos, se teclea `>> format short`, y si queremos ver en pantalla los resultados con 15 dígitos, se teclea `>> format long`. Si escribimos `>> format rat`, Matlab busca una aproximación racional del resultado, es decir, lo aproxima por una fracción. Por ejemplo, el número π lo aproxima por la fracción 355/113.

Si se desea obtener el resultado en forma decimal y con un número concreto de decimales, se usa la instrucción `>> vpa(ans,n)`, donde *ans* (o el nombre que se elija) es el nombre de la salida y *n* es el número total de dígitos que usará Matlab para representar su valor.

2. Números Complejos

Realiza por escrito los siguientes ejercicios y comprueba después los resultados usando Matlab.

1. Calcula

a) $\frac{(3-2i)(2+3i)}{3-4i}$

b) i^{2019}

2. Calcula $(1+i)^4$ de las siguientes maneras:

a) En forma polar.

b) Usando el binomio de Newton.

3. Dados $z = 1 + i$ y $w = \frac{9\pi}{4} + i \ln \sqrt{2}$ se pretende hallar el argumento del número complejo z^w cuyo módulo es 1. Para ello seguiremos los siguientes pasos:

a) Calcular $r = |z|$ y $\alpha = \arg(z)$, y comprobar los resultados con Matlab usando los comandos `abs` y `angle`. Recuerda usar el comando `>> sym()`.

b) Hallar la expresión exponencial del módulo de z^w .

c) Hallar k para que $|z^w|$ sea igual a 1.

d) Calcular el argumento de z^w para $k = 1$.

Universidad
de Huelva