

MATLABAPUNTES1.pdf



AlexMeriina



Matemáticas I



1º Grado en Ingeniería Informática



**Escuela Técnica Superior de Ingeniería
Universidad de Huelva**

Ya puedes imprimir desde Wuolah

Tus apuntes sin publi y al mejor precio

1 Añadir a la cesta

2 Cola de impresión

3 Impresión

4 Copistería Lowcost

Te enviamos los apuntes a casa

Recogelos en tu copistería más cercana

COMANDOS MATLAB

ACLARACIONES: A LA HORA DE ESCRIBIR NUMEROS LO HAREMOS SIN COMILLAS COMO EN C++, MIENTRAS QUE A LA HORA DE ESCRIBIR LETRAS O FRASES LO HAREMOS MEDIANTE COMILLAS SIMPLES.

Syms x: declara una/s variable/s no numérica/s. si queremos declarar mas de una deberemos separarlas por espacios.

Diary nombrearchivo: comienza a guardar nuestro progreso.

Diary off: deja de hacerlo.

Clear all: borra todas las variables (incluida ans).

Clear x: borra una variable.

Exp (exponente de e): $e^{\text{lo que metemos dentro del paréntesis}}$.

Format long, short, rat: formato corto, largo y aproximado.

Vpa (ans, n): Resultado final en decimales donde ans (o el nombre que se elija) es el nombre de la salida y n es el número total de dígitos que usa Matlab para representar su valor.

Pretty (ans): Respuesta dada de manera más vistosa.

-NÚMEROS COMPLEJOS-

Abs (): Nos da el valor absoluto de lo de dentro del paréntesis. $Y = \text{abs}(x-3)$.

Angle (): Nos devuelve el ángulo de un complejo.

Solve (): Si queremos resolver una ecuación $f(x)$, la introduciremos dentro de su paréntesis.

Subs (x==y): Comprueba si dos soluciones son equivalentes. 0=falsa; 1 = verdadera.

También podremos usar esta expresión para sustituir las x de una función y comprobar su valor: **subs (y, x0):** donde y es la función y x0 el valor que sustituye las x.

F(a)==Y(b): Comprueba si es cierta una expresión.

Limit (f(x), x, n°, right /left): Calcula el límite de una $f(x)$.

-REPRESENTACIÓN GRÁFICA-

Ezplot(f(x), [a, b]): A y b serán los valores máximos y mínimos del rango de representación de la gráfica.



ECUACION DE SEGUNDO GRADO

```
function[x1, x2] = ecu2(a, b, c)
%Para resolver la ecuación de coeficientes a, b y c escribiremos la
orden >> ecu2(a, b, c);
Delta = b^2 - 4 * a * c; %Se calcula el discriminante
if Delta < 0
disp('No hay soluciones reales 0')
elseif Delta > 0
x1 = (-b - sqrt(Delta))/(2 * a);
x2 = (-b + sqrt(Delta))/(2 * a);
disp('Hay dos soluciones reales 0'), x1, x2;
else
x1 = -b/(2 * a);
x2 = -b/(2 * a);
disp('Solo hay una solucion real'), x1, x2;
end
end
```

TEOREMA DE BOLZANO, BISECCION.

```
function [c,err,yc,iter] = bisection(f,a,b,delta)
%UNTITLED Summary of this function goes here
% Detailed explanation goes here
ya = feval(f, a);
yb = feval(f, b);
if ya*yb>0
disp('no se puede aplicar el teorema de bolzano');
return
end
err=b-a;
iter=0;
while err>=delta
c=(a+b)/2;
yc =feval(f,c);
if yc==0
a=c;
b=c;
elseif yb*yc>0
b=c;
yb=yc;
else
a=c;
ya=yc;
end
err=b-a;
iter=iter+1;
end
disp('El error es menor que 0'), err;
yc = vpa(feval(f, c), 10);
disp('El valor de la funcion en c es 0'), yc;
disp('El numero de iteraciones ha sido 0'), iter;
end
```

ESTRUCTURA FOR, WHILE E IF/ELSE

```
- if expresión lógica
instrucciones
elseif expresión lógica
instrucciones
elseif expresion lógica
instrucciones
...
else
instrucciones
end

- for indice = inicio: incremento: final /// SI NO PONEMOS INCREMENTO
SE INTERPRETARA QUE ES UNO.
instrucciones
end

- while expresion logica
instrucciones
end
```