



1 DE DICIEMBRE DE 2020

# SOLUCIÓN EJERCICIOS PROPUESTOS

MATLAB

DUVÁN ALBEIROMEJIA CORTES  
UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA



## EJERCICIOS PROPUESTOS

- Calcula el resultado de sumar 15 y 6:

```
>> a = 15;
>> b = 6;
>> suma = a + b;
>> suma

suma =

    21
```

- Guarda en la variable x el resultado de sumar 15 y 6:

```
>> x = suma;
```

- Haz que aparezca por pantalla el valor almacenado en la variable x:

```
>> x = suma;
>> disp(x)

    21
```

- Guarda en la variable y el resultado de multiplicar 12 y 2:

```
>> y = 12 * 2

y =

    24
```

- Realiza la suma de las variables x e y:

```
>> suma = x + y

suma =

    45
```

- Guarda en la variable z el resultado de restarle a la variable y la variable x:

```
>> z = y - x
```

- Haz que aparezca por pantalla el valor almacenado en la variable z:

```
>> disp(z)

    3
```

- Calcula el coseno de  $\pi$  (tomando el ángulo en radianes):

```
>> cos(pi)

ans =

    -1
```

- Calcula el coseno de 180o (tomando el ángulo en grados sexagesimales):

```
>> h = cosd(180)

h =

    -1
```

- Calcula la exponencial en 1 (es decir, el número e):

```
>> exp(1)

ans =

    2.7183
```

- Calcula la raíz cuadrada de -16:

```
>> sqrt(-16)

ans =

    0.0000 + 4.0000i
```

- Calcula el resultado de la división de 2 entre 3:

```
>> dividir = 2 / 3

dividir =

    0.6667
```

- Cambia a formato con 15 decimales:

```
>> format long
```

- Vuelve a calcular el resultado de la división de 2 entre 3:

```
>> format long
>> dividir = 2 / 3

dividir =

    0.6666666666666667
```

- Cambia a formato con solo 4 decimales:

```
>> format short
```

- Vuelve a calcular el resultado de la división de 2 entre 3:

```
>> format short
>> dividir = 2 / 3

dividir =

    0.6667
```

- Haz que aparezcan por pantalla las variables que estas utilizando:

```
>> whos

Name          Size          Bytes  Class    Attributes

a             1x1             8  double
ans           1x1            16  double   complex
b             1x1             8  double
dividir       1x1             8  double
sum           1x1             8  double
suma         1x1             8  double
x             1x1             8  double
y             1x1             8  double
z             1x1             8  double
```

- Borra la variable z:

```
>> clear z
|
```

- Vuelve a hacer que aparezcan por pantalla las variables que estas utilizando:

```
>> whos
```

Name	Size	Bytes	Class	Attributes
a	1x1	8	double	
ans	1x1	16	double	complex
b	1x1	8	double	
dividir	1x1	8	double	
sum	1x1	8	double	
suma	1x1	8	double	
x	1x1	8	double	
y	1x1	8	double	

- Crea el vector  $v = (1,2,3,4)$  de modo que no se vuelva a escribir en pantalla:

```
>> v = 1:4;
```

- Crea el vector  $w = (5,6,7,8)$  y deja que lo vuelva a escribir en pantalla:

```
>> w = 5:8
```

w =

5      6      7      8

- Calcula el vector traspuesto de v:

```
>> B = v.'
```

B =

1  
2  
3  
4

- Crea un vector llamado v2 donde sus elementos vayan desde el 2 al 17 creciendo de 3 en 3:

```
>> v2 = 2:3:17
```

v2 =

2      5      8      11      14      17

- Crea un vector  $v3$  donde sus elementos vayan desde el 2 al 20 y que en total tenga 10 elementos:

```
>> v3 = 2:2:20
```

```
v3 =
```

```
     2     4     6     8    10    12    14    16    18    20
```

- Averigua cual es el cuarto elemento del vector  $v3$ :

```
>> v3(:,4)
```

```
ans =
```

```
     8
```

- Crea la matriz  $M = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12)$

```
>> m = [1 2 3 4; 5 6 7 8; 9 10 11 12]
```

```
m =
```

```
     1     2     3     4
     5     6     7     8
     9    10    11    12
```

- Calcula la traspuesta de la matriz  $M$ :

```
>> m = m.'
```

```
m =
```

```
     1     5     9
     2     6    10
     3     7    11
     4     8    12
```

- Halla la fila 2 de la matriz M:

```
>> m(2,:)

ans =

     2     6    10
```

- Calcula el rango de M:

```
>> range(m)

ans =

     3     3     3

>> range(m,2)

ans =

     8
     8
     8
     8
```

- Calcula la traza de la matriz M:

```
>> D = diag(m)

D =

     1
     6
    11
```

- Crea la matriz identidad de tamaño 4:

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

a =

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

>> p = a*inv(a)
Warning: Matrix is close to singular or badly scaled. Result
may be inaccurate. RCOND = 3.265362e-19.

p =

    40    -32    -16     16
    64    -64    -32     32
   160   -128    -64     32
   128   -192    -96     96
```

- Crea la matriz nula de tamaño 3x3:

```
>> nula = zeros(3)

nula =

     0     0     0
     0     0     0
     0     0     0
```

- Crea la matriz cuadrada de unos de tamaño 2x2:

```
>> c = [1 2 ;3 4 ]

c =

     1     2
     3     4
```



- Averigua las dimensiones de la matriz M:

```
>> size(M)

ans =

     3     4
```

- Crea una matriz llamada M2 que tenga por diagonal el vector v y el resto sean todos ceros:

```
v =

     1     2     3     4

>> M2 = diag(v)

M2 =

     1     0     0     0
     0     2     0     0
     0     0     3     0
     0     0     0     4
```

- Haz que aparezca por pantalla la matriz triangular inferior a partir de M:

```
>> tril(M)

ans =

     1     0     0     0
     5     6     0     0
     9    10    11     0
```

- Haz que aparezca por pantalla la matriz triangular superior a partir de M:

```
>> triu(M)

ans =

     1     2     3     4
     0     6     7     8
     0     0    11    12
```

- Calcula una matriz que tenga por elementos todos los elementos de la matriz M elevados al cuadrado

```
>> sqrt(M)

ans =

    1.0000    1.4142    1.7321    2.0000
    2.2361    2.4495    2.6458    2.8284
    3.0000    3.1623    3.3166    3.4641
```

- Elimina de la matriz M su tercera columna

```
>> M(:,3) = []
```

```
M =

     1     2     4
     5     6     8
     9    10    12
```

- Calcula el determinante de M

```
>> d = det(M)

d =

    3.5527e-15
```

- Guarda en p el polinomio:  $x^3 - x^2 - 26x - 24$

```
>> p = [1 -1 -26 -24]
```

```
p =

     1     -1    -26    -24
```

- Calcula las raíces del polinomio p:

```
>> roots(p)

ans =

     6.0000
    -4.0000
    -1.0000
```

- Evalúa el polinomio  $p$  cuando  $x = 1$

```
>> polyval(p,1)
```

```
ans =
```

```
-50
```

- Evalúa el polinomio  $p$  en todos los valores del vector  $w$

```
>> polyval(p,w)
```

```
ans =
```

```
-54      0      88     216
```

- Crea un polinomio  $q$  que tenga por raíces los elementos del vector  $v$

```
>> q = poly(v)
```

```
q =
```

```
1    -10     35    -50     24
```

- Calcula la multiplicación de los polinomios  $p$  y  $q$

```
>> conv(p,q)
```

```
ans =
```

```
1    -11     19    151   -596    436    576   -576
```

- Calcula la división del polinomio obtenido como solución entre el polinomio q

```
>> S = conv(p,q)

S =

    1   -11    19   151  -596   436   576  -576

>> deconv (S,q)

ans =

    1    -1   -26   -24
```

- Escribe por pantalla el valor de los vectores v y w

```
>> disp(v)
    1     2     3     4

>> disp(w)
    5     6     7     8
```

- Calcula el producto de los vectores elemento a elemento

```
>> v.*w

ans =

    5    12    21    32
```

- Calcula el producto escalar de los vectores v y w

```
>> dot(v,w)

ans =

    70
```

- Calcula el producto del vector traspuesto de v con el vector w

```
>> V = v.'
```

```
V =
```

```
1
2
3
4
```

```
>> W = w.'
```

```
W =
```

```
5
6
7
8
```

```
>> V.*W
```

```
ans =
```

```
5
12
21
32
```

- Define el vector  $x = (-10, -9, -8, \dots, 6, 7, 8)$  y el vector y que sea el cuadrado de cada elemento

```
>> x = [-10, -9, -8, 6, 7, 8]
```

```
x =
```

```
-10    -9    -8     6     7     8
```

```
>> y = x,x.^2
```

```
y =
```

```
-10    -9    -8     6     7     8
```

```
ans =
```

```
100    81    64    36    49    64
```