>> a=[3 2 ; 2 4]

a =

 3 2

 2 4

%contar cuantos elementos son = 2

sum(a==2)

ans =

 1 1

>> a==2

ans =

 0 1

 1 0

>> sum(a==2)

ans =

 1 1

>> sum(sum(a==2))

ans =

 2

>> %sumar los >2

a.\*(a>2)

ans =

 3 0

 0 4

>> sum(sum(a.\*(a>2)))

ans =

 7

>> %mostrar solo los >2

>> a(a>2)

ans =

 3

 4

>> a(a>0)

ans =

 3

 2

 2

 4

>> x=a(a>2)

x =

 3

 4

>> %ubicar la posicion del elemento =3

>> a

a =

 3 2

 2 4

>> find(a==3)

ans =

 1

>> find(a==4)

ans =

 4

>> find(a==2)

ans =

 2

 3

>> [f c]=find(a==3)

f =

 1

c =

 1

>> [f c]=find(a==4)

f =

 2

c =

 2

>> [f c]=find(a==2)

f =

 2

 1

c =

 1

 2

>> x=[3 5 2 6 1];

>> max(x)

ans =

 6

>> [m i]=max(x)

m =

 6

i =

 4

>> clc

>> %Problema de matriz de ventas (vendedor x producto) y vector de precios

>> v=[5 3 2;4 0 7;1 6 3;4 3 4]

v =

 5 3 2

 4 0 7

 1 6 3

 4 3 4

>> p=[ 2 3 4];

>> %cantidad total de productos vendidos

>> sum(sum(v))

ans =

 42

>> %cantidad vendida por tipo de producto

>> sum(v)

ans =

 14 12 16

>> %vector columna de ventas por cada vendedor

>> v'

ans =

 5 4 1 4

 3 0 6 3

 2 7 3 4

>> sum(v')

ans =

 10 11 10 11

>> sum(v')'

ans =

 10

 11

 10

 11

>> %monto total vendido en $

>> v\*p'

ans =

 27

 36

 32

 33

>> %es el monto de cada vendedor

>> sum(v\*p')

ans =

 128

>> %es el monto de cada vendedor es:

>> sum(v\*p')

ans =

 128

>> %es el monto de cada vendedor es:

>> v\*p'

ans =

 27

 36

 32

 33

>> %el monto total es:

>> sum(v\*p')

ans =

 128

>> %monto total por producto

>> sum(v).\*p

ans =

 28 36 64

>> %matriz de montos:

>> precio=[p;p;p;p]

precio =

 2 3 4

 2 3 4

 2 3 4

 2 3 4

>> v.\*precio

ans =

 10 9 8

 8 0 28

 2 18 12

 8 9 16

>> montos=v.\*precio

montos =

 10 9 8

 8 0 28

 2 18 12

 8 9 16

>> %que vendedor vendio la mayor cantidad de algun producto

>> [f c]=find(v==max(max(v)))

f =

 2

c =

 3

>> v==max(max(v))

ans =

 0 0 0

 0 0 1

 0 0 0

 0 0 0

>> vendedor=f

vendedor =

 2

>> [vendedor c]=find(v==max(max(v)))

vendedor =

 2

c =

 3

>> %que producto se vendio en mayor cantidad

>> %Halle c el vector fila de cantidad vendida por tipo de producto

>> c=sum(v)

c =

 14 12 16

>> %Halle k el producto se vendio en mayor cantidad

>> [m k]=max(sum(v))

m =

 16

k =

 3

>> [m k]=max(c)

m =

 16

k =

 3

>> %intercambiar el mayor elemento con el menor

>> imayor=find(v==max(max(v)))

imayor =

 10

>> imenor=find(v==min(min(v)))

imenor =

 6

>> aux=v(imayor);

>> v(imayor)=v(imenor);

>> v(imenor)=aux;

>> v

v =

 5 3 2

 4 7 0

 1 6 3

 4 3 4

>> clc

>> %intercambiar el mayor elemento con el menor

>> imayor=find(v==max(max(v)));

>> imenor=find(v==min(min(v)));

>> aux=v(imayor);

>> v(imayor)=v(imenor);

>> v(imenor)=aux

v =

 5 3 2

 4 0 7

 1 6 3

 4 3 4

>> %intercambiar el mayor elemento con el menor

>> imayor=find(v==max(max(v)));

>> mayor=max(max(v));

>> imenor=find(v==min(min(v)));

>> menor=min(min(v));

>> v(imayor)=menor;

>> v(imenor)=mayor

v =

 5 3 2

 4 7 0

 1 6 3

 4 3 4

>> v(6)=0

v =

 5 3 2

 4 0 0

 1 6 3

 4 3 4

>> v(10)=7

v =

 5 3 2

 4 0 7

 1 6 3

 4 3 4

>> v

v =

 5 3 2

 4 0 7

 1 6 3

 4 3 4

>> v(4,2)=5

v =

 5 3 2

 4 0 7

 1 6 3

 4 5 4

>> v(2,1:3)

ans =

 4 0 7

>> v(2,:)

ans =

 4 0 7

>> v(:,3)

ans =

 2

 7

 3

 4

>> v

v =

 5 3 2

 4 0 7

 1 6 3

 4 5 4

>> v(2:2:4,:)

ans =

 4 0 7

 4 5 4

>> v(2:2:end,:)

ans =

 4 0 7

 4 5 4

>> x

x =

 3 5 2 6 1

>> length(x)

ans =

 5

>> length(v)

ans =

 4

>> [nf nc]=size(v)

nf =

 4

nc =

 3

>> v(2:2:end,1:2:end)

ans =

 4 7

 4 4

>> v

v =

 5 3 2

 4 0 7

 1 6 3

 4 5 4

>> v(5,5)

{Index exceeds matrix dimensions.

}

>> v(5,5)=3

v =

 5 3 2 0 0

 4 0 7 0 0

 1 6 3 0 0

 4 5 4 0 0

 0 0 0 0 3

>> v(2,2)=[]

{A null assignment can have only one non-colon index.

}

>> %elimina la columna 4

>> v(:,4)=[]

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> r=v

r =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> %eliminar de v la 2da y 4ta fila

>> v(2:2:end,:)=[]

v =

 5 3 2 0

 1 6 3 0

 0 0 0 3

>> v=r

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> v(2,:)=[]; v(4,:)=[]

v =

 5 3 2 0

 1 6 3 0

 4 5 4 0

%no estuvo bien

>> v=r

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> v([2 4],:)=[]

v =

 5 3 2 0

 1 6 3 0

 0 0 0 3

>> v=r

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

x=[v(2,:);[1 2 3];v(3:-1:2,:)]

{Error using <a href="matlab:matlab.internal.language.introspective.errorDocCallback('vertcat')" style="font-weight:bold">vertcat</a>

Dimensions of matrices being concatenated are not consistent.

}

>> v

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> x=[v [1 2 3 4 5]]

{Error using <a href="matlab:matlab.internal.language.introspective.errorDocCallback('horzcat')" style="font-weight:bold">horzcat</a>

Dimensions of matrices being concatenated are not consistent.

}

>> x=[v [1 2 3 4 5]']

x =

 5 3 2 0 1

 4 0 7 0 2

 1 6 3 0 3

 4 5 4 0 4

 0 0 0 3 5

>> clc

>> a=zeros(8,8)

a =

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

 0 0 0 0 0 0 0 0

>> a(1:2:end,1:2:end)=1

a =

 1 0 1 0 1 0 1 0

 0 0 0 0 0 0 0 0

 1 0 1 0 1 0 1 0

 0 0 0 0 0 0 0 0

 1 0 1 0 1 0 1 0

 0 0 0 0 0 0 0 0

 1 0 1 0 1 0 1 0

 0 0 0 0 0 0 0 0

>> a(2:2:end,1:4:end)=1

a =

 1 0 1 0 1 0 1 0

 1 0 0 0 1 0 0 0

 1 0 1 0 1 0 1 0

 1 0 0 0 1 0 0 0

 1 0 1 0 1 0 1 0

 1 0 0 0 1 0 0 0

 1 0 1 0 1 0 1 0

 1 0 0 0 1 0 0 0

>> a(2:2:end,2:4:end)=1

a =

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

>> a=zeros(8,8);

>> a(1:2:end,1:2:end)=1;

>> a(2:2:end,1:4:end)=1;

>> a(2:2:end,2:4:end)=1

a =

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

 1 0 1 0 1 0 1 0

 1 1 0 0 1 1 0 0

>> eye(2)

ans =

 1 0

 0 1

>> eye(5)

ans =

 1 0 0 0 0

 0 1 0 0 0

 0 0 1 0 0

 0 0 0 1 0

 0 0 0 0 1

>> magic(3)

ans =

 8 1 6

 3 5 7

 4 9 2

>> x

x =

 5 3 2 0 1

 4 0 7 0 2

 1 6 3 0 3

 4 5 4 0 4

 0 0 0 3 5

>> x=x(1,:)

x =

 5 3 2 0 1

>> x(:)

ans =

 5

 3

 2

 0

 1

>> v

v =

 5 3 2 0

 4 0 7 0

 1 6 3 0

 4 5 4 0

 0 0 0 3

>> v(:)

ans =

 5

 4

 1

 4

 0

 3

 0

 6

 5

 0

 2

 7

 3

 4

 0

 0

 0

 0

 0

 3

>> diary off