

**1.**

```
>> a=[1,2;3,6]
a =
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 6 \end{bmatrix}$$

```
>> b=[2,-3;1,4]
b =
```

$$\begin{bmatrix} 2 & -3 \\ 1 & 4 \end{bmatrix}$$

```
>> c=[-2,1;3,2]
c =
```

$$\begin{bmatrix} -2 & 1 \\ 3 & 2 \end{bmatrix}$$

**2.**

**a)**

```
>> a + b
ans =
```

$$\begin{bmatrix} 3 & -1 \\ 4 & 10 \end{bmatrix}$$

```
>> b + a
ans =
```

$$\begin{bmatrix} 3 & -1 \\ 4 & 10 \end{bmatrix}$$

**b)**

```
>> (a + b) + c
ans =
```

$$\begin{bmatrix} 1 & 0 \\ 7 & 12 \end{bmatrix}$$

```
>> a + (b + c)
ans =
```

$$\begin{bmatrix} 1 & 0 \\ 7 & 12 \end{bmatrix}$$

**c)**

```
>> 6*(a + b)
ans =
```

$$\begin{bmatrix} 18 & -6 \\ 24 & 60 \end{bmatrix}$$

```
>> 6*a + 6*b
ans =
```

$$\begin{bmatrix} 18 & -6 \\ 24 & 60 \end{bmatrix}$$

**d)**

```
>> a*(b + c)
ans =
```

```
      8      10
     24      30
```

```
>> a*b + a*c
ans =
```

```
      8      10
     24      30
```

**e)**

**(i)**

```
>> a*b
ans =
```

```
      4      5
     12     15
```

```
>> a*c
ans =
```

```
      4      5
     12     15
```

**(ii)**

```
>> a*b
ans =
```

```
      4      5
     12     15
```

```
>> b*a
ans =
```

```
     -7     -14
     13      26
```

**f)**

**(i)**

```
>> [a*b(:,1), a*b(:,2)]
```

```
ans =
```

```
      4      5
     12     15
```

**(ii)**

```
>> [a(1,:) * b; a(2,:) * b]
```

```
ans =
```

```
      4      5
     12     15
```

**(iii)** (i) The first column is made by all values in the first column of b multiplied by a

and the second column is made by all the values in the second column of b multiplied by a

- (ii) This one shows all the values in the first row of a multiplied by b make the first row and all the values in the second row of a multiplied by b make the second row.

**3)**

```
>> m=2*eye(3)
```

m =

```

2     0     0
0     2     0
0     0     2
```

```
>> n=zeros(2)
```

n =

```

0     0
0     0
```

```
>> p=triu(3*ones(3))
```

p =

```

3     3     3
0     3     3
0     0     3
```

```
>> q=4*ones(2,4)
```

q =

```

4     4     4     4
4     4     4     4
```

**4)**

```
>> g=zeros(6)
```

g =

```

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
```

```
>> g(1:2,1:2) = a(1:2,1:2)
```

g =

```

1     2     0     0     0     0
3     6     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
```

```
>> g(3:4,3:4) = b(1:2,1:2)
```

g =

1	2	0	0	0	0
3	6	0	0	0	0
0	0	2	-3	0	0
0	0	1	4	0	0
0	0	0	0	0	0
0	0	0	0	0	0

```
>> g(5:6,5:6) = c(1:2,1:2)
```

g =

1	2	0	0	0	0
3	6	0	0	0	0
0	0	2	-3	0	0
0	0	1	4	0	0
0	0	0	0	-2	1
0	0	0	0	3	2

```
>> g(1:2,3:4) = eye(2)
```

g =

1	2	1	0	0	0
3	6	0	1	0	0
0	0	2	-3	0	0
0	0	1	4	0	0
0	0	0	0	-2	1
0	0	0	0	3	2

```
>> g(3:4,5:6) = eye(2)
```

g =

1	2	1	0	0	0
3	6	0	1	0	0
0	0	2	-3	1	0
0	0	1	4	0	1
0	0	0	0	-2	1
0	0	0	0	3	2

```
>> g(5:6,1:2) = eye(2)
```

g =

1	2	1	0	0	0
3	6	0	1	0	0
0	0	2	-3	1	0
0	0	1	4	0	1
1	0	0	0	-2	1
0	1	0	0	3	2

5)

a)

```
>> h=g(1:3,1:3)
```

h =

1	2	1
3	6	0
0	0	2

**b)**

```
>> e=h
```

e =

1	2	1
3	6	0
0	0	2

```
>> e(3,3) = 5
```

e =

1	2	1
3	6	0
0	0	5

**c)**

```
>> f=h
```

f =

1	2	1
3	6	0
0	0	2

```
>> f(2,:) = []
```

f =

1	2	1
0	0	2

**d)**

g(:, :) means everything from the first row to the last row from the first column to the last column

g(:) means everything from beginning to the end

**e)**

entering g(8) looks for the 8th entry counting from the top left corner to the bottom right going down the rows and moving to the top of the next row after it reaches the end. g(16) proves this.

**f)**

g(10,1) is asking for the value in the 10<sup>th</sup> row and the 1<sup>st</sup> column. There are only 6 rows so it exceeds the matrix bounds

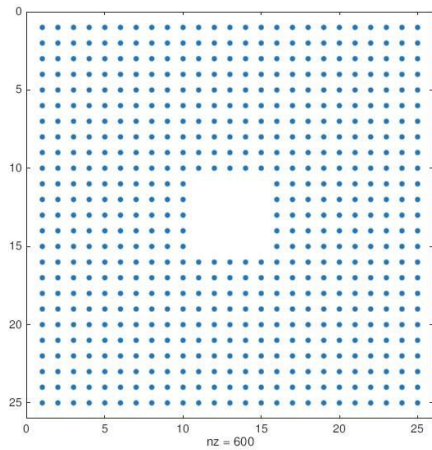
**g)**

g(g>x) is asking for all values in matrix g that are greater than x.

g(g>x) = y changes all values in g that are greater than x to y

6)

```
>> A = ones(25);
>> A(11:15,11:15) = 0;
```



7)

```
>> A = diag(1:6) + diag(7:11, 1) + diag(12:15, 2)
A =
```

1	7	12	0	0	0
0	2	8	13	0	0
0	0	3	9	14	0
0	0	0	4	10	15
0	0	0	0	5	11
0	0	0	0	0	6

```
>> A = A + triu(A,1)
A =
```

1	14	24	0	0	0
0	2	16	26	0	0
0	0	3	18	28	0
0	0	0	4	20	30
0	0	0	0	5	22
0	0	0	0	0	6

8)

(a)

```
>> A(2,:) = A(2,:) - 4*A(1,:)
A =
```

1	2	3
0	-3	-6
7	8	9

All the values in row 2 equal the difference of all the values in the second row of A and 4 times all the values in the first row of A.

**b)**

```
>> A(3,:) = 7*A(1,:) - A(3,:)
A =
```

1	2	3
0	-3	-6
0	6	12

**c)**

```
>> A(2,:) = -3 * A(2,:)
A =
```

1	2	3
0	9	18
0	6	12