Consider the following matrices

$$A = \begin{bmatrix} 1 & 2 & 1 \\ -1 & 0 & -2 \\ 0 & 3 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}, \quad C = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 3 & 0 \end{bmatrix}$$

and vectors

$$x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \quad y = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$$

Use = to assign variables for the preceding matrices.

```
>> A = [1 2 1; -1 0 -2; 0 3 1];
>> B = [1 2; 3 4; 5 6];
>> C = [2 0; 0 1; 3 0];
>> x = [1; 2; 3];
>> y = [4; 5; 6];
```

Matrix addition and subtraction syntax is the usual one with + and -.

```
>> x+y                % adds the vectors x and y
>> B+C                % adds the matrices B and C
>> B-C                % subtracts C from B
>> A+B                % returns an error, matrices are the wrong size
```

Matrix multiplication and scalar multiplication are both handled with *.

```
>> 2*x                % scales the vector x by 2
>> 3*A                % scales the matrix A by 3
>> A*x                % Multiplies A and x
>> A*B                % Multiplies A and B
>> B*A                % returns an error, matrices are the wrong size
```

Use parentheses to form more complicated expressions.

```
>> A*(x+y) + (A+A)*y
```

See the precendence of the operators: `https://www.mathworks.com/help/matlab/matlab_prog/operator-precedence.html`