reference: https://wiki.octave.org/Using_Octave

variable assignment

```
a = 1;
```

comments

```
% this is comment; % is both supported by Octave and Matlab
```

Command evaluation

```
t = 99 + 1
```

```
t = 100
```

```
t = 100
```

```
t = 100
```

```
t = 99 + 1;
disp(t)
```

```
   100
```

Elementary math

```
x = 3/4*pi;
y = sin(x)
```

```
y = 0.7071
```

Matrices

```
rowVec = [8 6 4]
```

```
rowVec = 1×3
     8     6     4
```

```
size(rowVec)
```

```
ans = 1×2
     1     3
```

```
columnVec = [8;6;4]
```

```
columnVec = 3×1
     8
     6
     4
```

```
size(columnVec)
```

```
ans = 1×2
     3     1
```

```
mat = [8,6,4;2,0,-2]
```

```
mat = 2×3
     8     6     4
     2     0    -2
```

```
size(mat)
```

```
ans = 1×2
     2     3
```

Linear Algebra

```
columnVec * rowVec
```

```
ans = 3×3
    64    48    32
    48    36    24
    32    24    16
```

```
rowVec * columnVec
```

```
ans = 116
```

```
columnVec'
```

```
ans = 1×3
     8     6     4
```

Accessing Elements

```
mat(2,3)
```

```
ans = -2
```

control flow

```
x = zeros(50,1);
for i = 1:2:100
    x(i) = i^2;
end %endfor is used in Octave, Octave is also support end syntax
```

```
% doc(':')
disp(x)
```

```
     1
     0
     9
     0
    25
     0
    49
     0
    81
     0
   121
     0
   169
     0
   225
     0
   289
     0
   361
     0
   441
     0
   529
     0
   625
     0
   729
     0
   841
     0
   961
     0
  1089
     0
  1225
     0
  1369
     0
  1521
     0
  1681
     0
  1849
     0
  2025
     0
  2209
     0
  2401
     0
  2601
     0
  2809
     0
  3025
     0
  3249
     0
  3481
     0
  3721
```

```
     0
  3969
     0
  4225
     0
  4489
     0
  4761
     0
  5041
     0
  5329
     0
  5625
     0
  5929
     0
  6241
     0
  6561
     0
  6889
     0
  7225
     0
  7569
     0
  7921
     0
  8281
     0
  8649
     0
  9025
     0
  9409
     0
  9801
```

```
y = zeros(50,1);
k = 1;
step = 2;
while (k<=100)
    y(k) = k^2;
    k = k + step;
end
disp(y)
```

```
     1
     0
     9
     0
    25
     0
    49
     0
    81
     0
   121
     0
   169
     0
```

```
 225
   0
 289
   0
 361
   0
 441
   0
 529
   0
 625
   0
 729
   0
 841
   0
 961
   0
1089
   0
1225
   0
1369
   0
1521
   0
1681
   0
1849
   0
2025
   0
2209
   0
2401
   0
2601
   0
2809
   0
3025
   0
3249
   0
3481
   0
3721
   0
3969
   0
4225
   0
4489
   0
4761
   0
5041
   0
5329
   0
5625
   0
5929
   0
6241
```

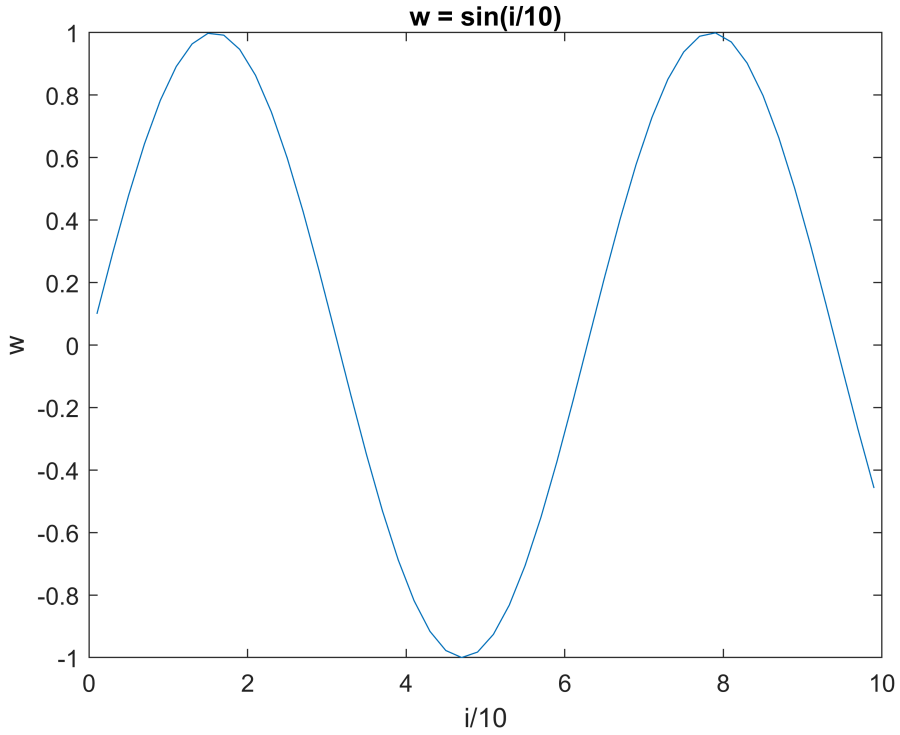
```
     0
  6561
     0
  6889
     0
  7225
     0
  7569
     0
  7921
     0
  8281
     0
  8649
     0
  9025
     0
  9409
     0
  9801
```

vectorization

```
i = 1:2:100;
x = i.^2;
y = x + 9;
z = y./i;
w = sin(i/10);
```

plotting

```
plot(i/10, w);
title('w = sin(i/10)')
xlabel('i/10')
ylabel('w')
```



Strings

```
firstString = 'hello world'; %octave support '' for string (also "")
secondString = '!';
disp([firstString, secondString])
```

```
hello world!
```

```
fprintf('%s %.10f\n', 'The number is: ', 10)
```

```
The number is:  10.0000000000
```

If-else

```
if 3>2
    disp('hello')
else
    disp('world')
end
```

getting help

```
doc('plot') %personally recommand use this format
```

octave forge packages

```
% not needed for PHYS4150 course
```