Table of Contents

Explores some basic concepts about unsigned integer numbers.

```
d = 75;
u = uint8(d) % ######### 8 #####
% bits = bitget(u, 1:8) % ###??? A ##### bit ###
% bits = fliplr(bits); % #####
bits = dec2bin(u)
number = bin2dec(bits)
```

```
u =
  uint8
   75
bits =
    '1001011'
number =
    75
```

Explores some basic concepts about signed integer numbers.

```
d = 200
u = uint8(d)
bits = dec2bin(u) % dec2bin : #########d########
t = int8(u)
s = typecast(u, 'int8') % ##################
a = int16(u)
bits = dec2bin(a)
```

```
d =
   200
```

```
u =
  uint8
   200
bits =
    '11001000'
t =
  int8
   127
s =
  int8
   -56
a =
  int16
   200
bits =
    '11001000'
```

Shows how the computer stores a real number

```
d = 28
a = typecast(d, 'uint64')
b = dec2bin(a, 64)
```

```
d =
    28
a =
  uint64
   4628574517030027264
b =
    '0100000000111100000000000000000000000000000000000000000000000000'
```

Shows the number of significant digits of a floating-point number

```
format short;
format compact;
a = 1234.56789012345678901234;
fprintf('%.20f\n', a)
format long;
a;
b = single(a); % single(X)#X#########
fprintf('%.20f\n', b)
```

```
1234.56789012345689116046
1234.56787109375000000000
```

Explores some basic concepts about characters and strings.

```
a = 'A';
```

```
b = a + 1;
char(65) % ########
char('A' + 2)
c = ['A', 'B', 'C']
d = ['AB', 'C']
e = ['A', 66, 67]
f = 'ABC'
f(1)
f(2)
f(3)
```

```
ans =
    'A'
ans =
    'C'
c =
    'ABC'
d =
    'ABC'
e =
    'ABC'
f =
    'ABC'
ans =
    'A'
ans =
    'B'
ans =
    'C'
```

Prints an ASCII Codes table.

```
fprintf('    0 1 2 3 4 5 6 7 8 9\n')
for row = 3:12
    fprintf('%2d  ', row)
    for column = 0:9
        code = row*10+column;
        if (code < 32) || (code > 126)
            fprintf('  ')
        else
            fprintf('%c ', code)
        end
    end
    fprintf('\n')
end
```

```
    0 1 2 3 4 5 6 7 8 9
 3        ! " # $ % & '
 4  ( ) * + , - . / 0 1
 5  2 3 4 5 6 7 8 9 : ;
 6  < = > ? @ A B C D E
 7  F G H I J K L M N O
 8  P Q R S T U V W X Y
 9  Z [ \ ] ^ _ ` a b c
10  d e f g h i j k l m
11  n o p q r s t u v w
12  x y z { | } ~
```

Explores some basic concepts about logical data.

```
a = true
b = false
c = 6 > 5
d = 6 < 5
e = (6 > 5)*10
f = false*10+true*2
g = (6 > 5) & (6 < 5)
h = (6 > 5) | (6 < 5)
k = logical(5)
m = 5 | 0
n = (-2) & 'A'
```

```
a =
 logical
  1
b =
 logical
  0
c =
 logical
  1
d =
 logical
  0
e =
  10
f =
   2
g =
 logical
  0
h =
 logical
  1
k =
 logical
  1
m =
 logical
  1
n =
 logical
  1
```

Explores some basic concepts about arrays.

```
a = 5
b = [5]
c = 5*ones(1,1)
D = ones(2, 3)
e = [1, 2, 3, 4, 5]
f = [1 2 3 4 5]
g = [1:5]
h = 1:5
k = 1:1:5
m = linspace(1, 5, 5)
```

```
a =
     5
b =
     5
c =
     5
D =
     1     1     1
     1     1     1
e =
     1     2     3     4     5
f =
     1     2     3     4     5
g =
     1     2     3     4     5
h =
     1     2     3     4     5
k =
     1     2     3     4     5
m =
     1     2     3     4     5
```

Explores some basic concepts about arrays

```
clear all clc
a = zeros(1,5)
a(1,5) = 8
a(5) = 9
a([1, 2, 4]) = [8, 7, 6]
a(1:4) = [2, 3, 4, 5]
[rows, cols] = size(a)
len = length(a)
b = a
c = a(1:5)
d = a(3:5)
e = a(3:length(a))
f = a(3:end)
f(5) = 10
```

```
a =
     0     0     0     0     0
a =
     0     0     0     0     8
a =
     0     0     0     0     9
a =
     8     7     0     6     9
a =
     2     3     4     5     9
rows =
     1
cols =
     5
len =
     5
b =
     2     3     4     5     9
c =
     2     3     4     5     9
d =
     4     5     9
e =
     4     5     9
f =
     4     5     9
f =
     4     5     9     0    10
```

Explores some basic concepts about arrays.

```
a = [1, 2; 3, 4; 5, 6]
b = 1:6
c = reshape(b, 3, 2)
d = reshape(b, 2, 3)
e = d'
c(:,3) = [7, 8, 9]
c(4,:) = [10, 11, 12]
c(4,:) = []
c(:,2:3) = []
```

```
a =
     1     2
     3     4
     5     6
b =
     1     2     3     4     5     6
c =
     1     4
     2     5
     3     6
d =
     1     3     5
     2     4     6
```

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

Explores some basic concepts about arrays.

```
a = reshape(1:6, 3, 2)
b = [7; 8; 9]
c = horzcat(a, b) % ########
d = [a, b]
e = b'
f = vertcat(d, e) % ########
g = [d; e]
h = fliplr(c) % #####
k = flipud(c) % #####
```

```
a =
     1     4
     2     5
     3     6
b =
     7
     8
     9
c =
     1     4     7
     2     5     8
     3     6     9
d =
     1     4     7
     2     5     8
     3     6     9
e =
     7     8     9
f =
     1     4     7
     2     5     8
     3     6     9
     7     8     9
g =
     1     4     7
     2     5     8
     3     6     9
     7     8     9
h =
     7     4     1
     8     5     2
     9     6     3
k =
     3     6     9
     2     5     8
     1     4     7
```

Demonstrates the use of sum, cumsum, prod, cumprod, diff, min, and max.

```
a = 1:5
b = sum(a) % ######
c = cumsum(a)
% cumsum(A)#A###########1#########A#####
% ##A#####cumsum(A)####A#########
% ##A#####cumsum(A)####A##########
% ##A#######cumsum(A)###########

d = prod(a) % ##a########
e = cumprod(a)
% cumprod(A) #A###########1#########A######
% ##A####cumprod(A)####A###########
% ##A#####cumprod(A)####A###########
% ##A#######cumprod(A)##########

f = diff(a) % #######
A = reshape(1:9, 3, 3)
g = sum(A)
B = cumsum(A)
h = prod(A)
C = cumprod(A)
D = diff(A)
p = min(a)
q = max(a)
r = min(A) % ##A#####min(A)##############
s = max(A) % ##A#####max(A)##############
```

```
a =
     1     2     3     4     5
b =
    15
```

```
c =
     1     3     6    10    15
d =
   120
e =
     1     2     6    24   120
f =
     1     1     1     1
A =
     1     4     7
     2     5     8
     3     6     9
g =
     6    15    24
B =
     1     4     7
     3     9    15
     6    15    24
h =
     6   120   504
C =
     1     4     7
     2    20    56
     6   120   504
D =
     1     1     1
     1     1     1
p =
     1
q =
     5
r =
     1     4     7
s =
     3     6     9
```

Demonstrates the use of arithmetic operators for matrices.

```
A = reshape(1:6, 2, 3)
B = reshape(7:12, 2, 3)
C = A+B
D = A-B
E = B'
F = A*E
a = [3, 6]
b = a/F
c = b*F
G = F^2
H = A.*B
K = A./B
M = A.^2
```

```
P = A+10
Q = A-10
R = A*1.5
S = A/2
```

```
A =
     1     3     5
     2     4     6
B =
     7     9    11
     8    10    12
C =
     8    12    16
    10    14    18
D =
    -6    -6    -6
    -6    -6    -6
E =
     7     8
     9    10
    11    12
F =
    89    98
   116   128
a =
     3     6
b =
 -12.999999999999723   9.999999999999789
c =
     3     6
G =
       19289       21266
       25172       27752
H =
     7    27    55
    16    40    72
K =
   0.142857142857143   0.333333333333333   0.454545454545455
   0.250000000000000   0.400000000000000   0.500000000000000
M =
     1     9    25
     4    16    36
P =
    11    13    15
    12    14    16
Q =
    -9    -7    -5
    -8    -6    -4
R =
   1.500000000000000   4.500000000000000   7.500000000000000
   3.000000000000000   6.000000000000000   9.000000000000000
S =
   0.500000000000000   1.500000000000000   2.500000000000000
   1.000000000000000   2.000000000000000   3.000000000000000
```

Demonstrates the use of arithmetic operators for vectors.

```
a = 1:4
b = 5:8
c = a+b
d = a-b
e = a*(b')
f = (a')*b
g = a/b
h = a.*b
k = a./b
m = a.^2

a =
     1     2     3     4
b =
     5     6     7     8
c =
     6     8    10    12
d =
    -4    -4    -4    -4
e =
    70
f =
     5     6     7     8
    10    12    14    16
    15    18    21    24
    20    24    28    32
g =
   0.402298850574713
h =
     5    12    21    32
k =
  Columns 1 through 3
   0.200000000000000   0.333333333333333   0.428571428571429
  Column 4
   0.500000000000000
m =
     1     4     9    16
```

Demonstrates the use of arithmetic operators for scalars.

```
a = 6
b = 4
c = a+b
d = a-b
e = a*b
f = a/b
```

```
g = a^2
h = a.*b
k = a./b
m = a.^2
```

```
a =
     6
b =
     4
c =
    10
d =
     2
e =
    24
f =
   1.500000000000000
g =
    36
h =
    24
k =
   1.500000000000000
m =
    36
```

Published with MATLAB® R2018a