

# Matlab

Ders 2 - 11/10/2022

## İçerik Listesi

- [input](#)
  - [input Fonksiyonu](#)
- [fprintf](#)
  - [fprintf fonksiyonu](#)
- [Diziler](#)
- [Kümeler \(Hücre dizileri\)](#)
- [Struct](#)

## input

### input Fonksiyonu

Kullanıcıdan input almak için `input` fonksiyonu kullanılır.

```
a = input('a = ')
```

'a = ' yerine tırnak içine herhangi bir şey yazılabilir ve kullanıcıya bilgi verilebilir.

**ÖRNEK:** Kullanıcıdan istenilen üç adet sayının ortalamasını bulan MATLAB programı

```
a = input('a = ');  
b = input('b = ');  
c = input('c = ');  
ort = (a+b+c)/3;  
disp(ort);
```

## fprintf

### fprintf fonksiyonu

fprintf fonksiyonu bir değişkenin değerini yazdırmak için % kullanır

```
a = 5;  
fprintf('a = %d \n', a);
```

`\n` = new line yani yeni satır

`\t` = tab yani 4 boşluk

```
a = 5;
b = 7;
fprintf('a = %d \t b = %d \n', a, b);
```

`%d` tam sayı için kullanılırken `%f` (float) ondalıklı sayı için (6 ondalığa kadar yazdırır) kullanılır.

```
a = 5;
b = 22/7;
fprintf('a = %d \t b = %f \n', a, b);
```

`%.2f` gibi 2 yerine yazılarak kaç basamak ondalık yazılacağı belirtilebilir. String yazdırmak için `%s` kullanılır. Char için de `%c` kullanılır.

```
a = 5;
b = 22/7;
c = "OGU";
d = 'F';
fprintf('a = %d \t b = %f \n c = %s, d = %c\n', a, b, c, d);
```

`%g` ise sayıları yazdırmak için kullanılır ve f ile d arasında kararı kendi verir. 5 ondalık yazdırır.

**ÖRNEK:** Kullanıcıdan istenilen 4 adet sayı için aşağıdaki  $x$  değerini hesaplayan MATLAB programı

$$x = |a^2 - \sin(2bc) + \sqrt[12]{d^5}|$$

```
a = input('a = ');
b = input('b = ');
c = input('c = ');
d = input('d = ');
x = abs(a^2 - sin(2*b*c) + (d^5)^(1/12));
disp(x);
```

**ÖRNEK:** Kullanıcıdan istenilen yarıçap değeri için bir dairenin çevresini ve alanını bulan program yazınız

$$\text{Çevre} = 2\pi r$$

$$\text{Alan} = \pi r^2$$

```
r = input('Yarıçap değeri = ');
cevre = 2*pi*r;
alan = pi*r^2;
fprintf('Cevre = %f \nAlan = %f\n', cevre, alan);
```

## Diziler

Diziler tanımlanırken [] kullanılır.

`A = [1,2,3,4,5]` bu şekilde arasına `,` koyarsanız satır vektörü olur. Eğer `B = [1;2;3;4;5]` şeklinde `;` koyarak kullanırsanız sütun vektörü oluşur.

```
A =  
  
    1    2    3    4    5  
  
B =  
  
    1  
    2  
    3  
    4  
    5
```

Dizinin belli bir elemanını almak için kaçınca elemanı olduğunu parantez içinde yazın. Örn `A(2)` demek 2. elemanı demek. Aynı zamanda `A(2) = x` (x herhangi bir değer) diyerek o elemanı değiştirebilirsiniz. Matrislerde eleman alırken `M(satır,sütun)` şeklinde alınabilir. Ardışık sayı dizisi oluşturmak şu şekilde yapılabilir `B = 1:200`, B 1den 200e kadar olan sayılar dizisi. Aynı zamanda `B = 1:5:200` diyerek 5er 5er artır diyerek kullanılabilir. `10:-1:1` gibi de kullanılabilir, `1:0.1:10` gibi de.

- Vektörü ters çevirmek için `flip` fonksiyonu kullanılabilir, `flip(A)` gibi.
- `linspace` fonksiyonu ile x den y ye a elemanlı dizi oluşturabilirsiniz. Örneğin `linspace(1,10,2)` çıktısı `1 10` olur. Elemanlar doğrusal artan ayarlanır.
- `zeros` fonksiyonu ile 0 matrisi oluşturulabilir. `zeros(satır,sütun)` gibi, aynı şekilde `ones` ile birler matrisi oluşturulabilir.

- `rand` fonksiyonu kullanılarak elemanları rastgele 0 ile 1 arasında olan bir matris oluşturulabilir, `rand(satır,sütun)` şeklinde. \
- `randi` fonksiyonu belirtilen sayılar arasında, istenilen satır ve sütun sayısı ile, istenilen sayıda matris verir.

1 - 4 arası rastgele sayı.

```
>> randi(4)
```

```
ans =
```

```
2
```

1-4 arası sayılardan oluşan 2×3 lük matris

```
>> randi(4,2,3)
```

```
ans =
```

```
3    4    4
4    1    4
```

1-2 arası sayılardan oluşan 4×2 lik 3 tane matris

```
>> randi(2,4,2,3)
```

```
ans(:, :, 1) =
```

```
1    2
2    2
1    2
1    2
```

```
ans(:, :, 2) =
```

```
1    2
2    2
2    1
2    2
```

```
ans(:, :, 3) =
```

```
1 1
2 1
1 2
1 2
```

2-4 arası sayı

```
>> randi([2,4])
```

```
ans =
```

```
2
```

- `round` fonksiyonu yuvarlamaya yarar. `round(2.1)` çıktısı 2 dir. 2. parametre olarak kaç ondalık olacağı belirtilebilir. `rand(22/7,2)` nin çıktısı 3.1400 dir.
- `length` fonksiyonu uzunluk almak için kullanılır.
- `size` fonksiyonu boyut için kullanılır. Çıktı satır sütun şeklindedir.
- `sum` fonksiyonu topla demek. Dizinin elemanları toplamını verir.
- `min` ve `max` fonksiyonları adı üstünde min ve max elemanları döner.
- `:` işareti hepsi demek.

## Kümeler (Hücre dizileri)

`{}` kullanılarak oluşturulur. Elemanları her şey olabilir.

```
s = {1,2,3,'a'}
```

```
s(1)
```

```
% 1. elemanı olan alt küme
```

```
s{1}
```

```
% 1. eleman
```

## Struct

```
ogrenci.no = 123  
ogrenci.isim = 'Ali'
```

```
ogrenci =  
  
  struct with fields:  
  
    no: 123  
  
ogrenci =  
  
  struct with fields:  
  
    no: 123  
    isim: 'Ali'
```

[< Önceki](#) | [Sonraki >](#)