

3.5 FOR Döngüsü

MATLAB'ın dizey tabanlı değişken mantığı esasen dizey elemanlarının topluca kullanılmasına olanak vermektedir. Bu özelliği nedeni ile önceki kuşak dillerde çoklu işlemleri yürütmek için sıkça kullandığımız döngülerin birçoğu gereksiz hale gelmektedir. Ancak kimi hesaplamalar indeksleme ya da yürütme sırası açısından mutlaka döngüler içerisinde yapılmak zorundadır. MATLAB programlama dilinde döngüler `for` deyimi ile başlatılmaktadır. Yazımı en genel haliyle

```
for <indeks=başlangıç:artım:bitiş>
    <işlemler>
end
```

şeklinde dir. Burada indeks döngü sayacı olarak adlandırılmakta ve eşittir işaretinin sağına başlangıç, artım ve bitiş değerleri yazılmaktadır. **indeks** değişkeni kaç ayrı noktada değer alıyorsa `for` ve `end` kelimeleri arasındaki işlemler o kadar tekrarlanır. Örnekler konunun daha iyi anlaşılmasına yardımcı olacaktır.

```
clc
for k=1:5
    disp([num2str(k) 'numaralı döngü adımı işletildi']);
end
```

Yukarıdaki örnek çalıştırıldığında komut satırında aşağıdaki çıktı görüntülenir.

```
1 numaralı döngü adımı işletildi
2 numaralı döngü adımı işletildi
3 numaralı döngü adımı işletildi
4 numaralı döngü adımı işletildi
5 numaralı döngü adımı işletildi
```

Örnekte **k** döngü sayacıdır. Sayacın değerinin 1'den başlayıp 5'e kadar birer artımlarla gideceği belirtilmiştir. Buradaki sayaç değeri daha önce : işlecinin periyodik artan/azalan diziler oluşturmada kullanılmasına benzer şekilde atanabilmektedir. Aşağıda geçerli olan çeşitli sayaç tanımlamaları verilmiştir.

Çizelge 3.1 for döngüsü için çeşitli sayaç tanımları

Döngü sayacı	Tekrar sayısı
<code>for k=1:2:51</code>	26
<code>for t=-3:.25:3</code>	25
<code>for x=10:-1:1</code>	10
<code>for n=1:length(f)</code>	f dizeyinin boyu
<code>for A=[1 3 6 7 8]</code>	5
<code>for J=1:inf</code>	sonsuz

Döngüler birbirini kesmemek kaydı ile iç içe kullanılabilir. Aşağıda buna bir örnek verilmiştir.

```
for i=1:n
    for j=1:m
        E(i,j)=1/(i+j);
    end
end
```

i ve *j* sayaçları sırasıyla *n* ve *m* kez tekrarlanır. Burada dikkat edilmesi gereken konu döngü sayaçlarının aynı zamanda *n* dizeyinin indekslenmesinde kullanılıyor olmasıdır. Bu nedenle bu tür kullanımlarda sayaçların pozitif tam sayı değerler alıyor olması zorunludur.

Döngüler özel bir koşul sağlandığında döngü sayacı tüm değerlerini almadan durdurulmak istenebilir. Bu amaçla **break** deyimi kullanılmaktadır. Döngü tekrarları yürütülürken özel bir koşul sağlandığında sayaçlar son değerine ulaşmadan o adımda döngü durdurulabilir. Aşağıda buna bir örnek verilmiştir.

```
t=0;
for x=1:inf
    t=t+x^2;
    if t>1000
        break
    end
end
```

Burada döngü sayacı birden sonsuza kadar tanımlanmıştır. Döngü durdurulmazsa sonsuza kadar tekrarlanacaktır. Bir **if** deyimi ile denetlenen koşul sağlandığında döngü o adımda duracaktır. Buna benzer olarak özel bir koşul sağlandığında geçerli döngü adımı atlanarak sayacın bir sonraki değerinde devam etmesi sağlanmak istenebilir. Bu amaçla **continue** deyimi kullanılmaktadır. Aşağıda sıfıra bölme işlemini engellemek amacıyla bu deyimin kullanılmasına bir örnek verilmiştir.

```
i=1;
for x=-3:0.1:3
    if x==0
        continue
    end
end
```

```
else
    f(i)=sin(x)/x;
    i=i+1;
end
end
```

Verilen örnekte x , 0 değeri aldığı anda herhangi bir işlem yapılmadan sayacın bir sonraki değerine geçilecektir.

3.6 While Döngüsü

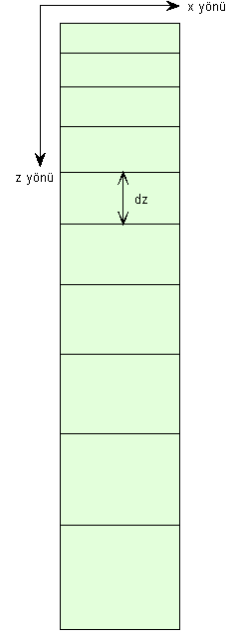
MATLAB programlama dilinde döngü kurmak için diğer bir yol `while` deyimini kullanmaktır. `while` deyimi bir sayaca göre değil bir koşula göre tekrarlanan bir döngü oluşturmaktadır. Başka bir deyişle tanımlanacak bir önerme doğru olduğu sürece `while` ve `end` kelimeleri arasındaki deyimler tekrarlı olarak işletilmeye devam edecektir. Aşağıdaki örnek `while` deyimi ile döngü kurma mantık ve yazılışını açıklamaktadır.

```
eps=0.25;
while eps>0.001
    xf=exp(x)*1/eps;
    eps=eps/2;
end
```

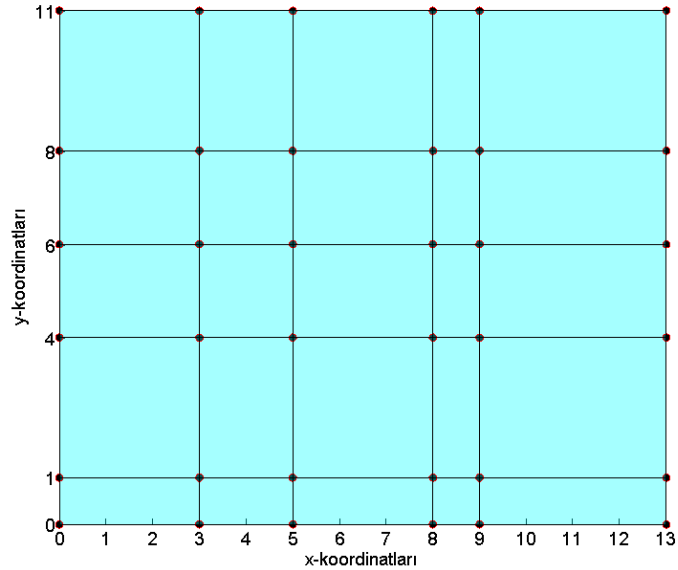
verilen örnekte **eps** değişkenin değeri 0.001'den büyük olduğu sürece döngü içerisindeki ifadelerin işletilmesi tekrarlanacaktır. Döngü içerisinde **eps** değişkenin değeri her adımda yarı yarıya azalmaktadır. Belirlenen değerin altına düştüğünde döngü kendiliğinden duracaktır. `for` döngüsü için kullanılan `continue` ve `break` deyimleri `while` döngüsü için de aynı şekilde kullanılabilir.

1. Aşağıdaki problemi çözen bir MATLAB programı yazınız.

Yandaki şekil bir boyutlu bir yer modelidir. Modelin üst kenarı $z=0$ derinliğindedir. Modeli oluşturan dikdörtgen hücrelerin z yönündeki boyutları ardışık olarak %15 oranında artmaktadır. Toplam 10 dikdörtgen hücre olduğu göz önüne alınarak en alttaki hücrenin alt kenarının -10m derinlikte olması için tüm hücrelerin z yönündeki boyutları ne olmalıdır?



2. Aşağıdaki grafikte bir sahada ölçüm yapılan noktaların konumları işaretlenmiştir. İşaretli noktaların tümünün $(0,0)$ noktasına olan uzaklığını bulan bir MATLAB programı yazınız.



3. pi sayısı aşağıdaki seri yardımı ile virgülden sonra 4 hane doğrulukla hesaplayınız. Karşılaştırma için MATLAB ön tanımlı değişkeni `pi` kullanılacaktır.

$$\pi = 4 \sum_{k=0}^m \frac{(-1)^k}{2k+1}$$

4. Bir ülkenin nüfusu her yıl %1.13 oranında artmaktadır. Buna göre bu ülkenin nüfusunun bugünkü değerinin 1.5 katını kaç yıl sonra geçeceğini bulan bir MATLAB programı yazınız.
5. $\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$ serisini, toplama eklenen terimlerin mutlak değeri 1e-05'den küçük oluncaya kadar hesaplatınız.