



# Bilgisayar Programlama 2

Ders Notu: 4

# Listeler

- Bir senaryo olarak bir kullanıcıdan otuz test puanı almamız ve bu puanlarla işlemler gerekebilir, örneğin onları sıralamak gibi. puan1, puan2, ..., puan30 gibi otuz değişken oluşturabiliriz, ancak bu çok zahmetli olurdu. Ardından puanları sıralamak son derece zor olurdu. Çözüm, listeleri kullanmaktır.

## Temel Bilgiler

- **Listeler oluşturma:**

- Listenin başlangıcını ve sonunu belirtmek için köşeli parantez kullanın ve öğeleri virgülle ayırılır.

```
L = [1, 2, 3]
```

- **Boş liste:** Boş liste [] şeklindedir. Bu, 0 veya "" değerine karşılık gelen listedir.

- **Uzun listeler:** Eğer girilmesi gereken uzun bir liste varsa, aşağıdaki gibi birkaç satıra bölünebilir:

```
nums = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,  
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31,  
        32, 33, 34, 35, 36, 37, 38, 39, 40]
```

- **Girdi:** Kullanıcının bir liste girmesine izin vermek için eval(input()) kullanılabilir.

```
1 L = eval(input('Bir liste giriniz: '))  
2 print('İlk eleman: ', L[0])  
3 print('Tüm elemanlar: ', L)
```

- **Veri tipleri:** Listeler her türlü veri tipini, hatta başka listeleri bile içerebilir. Örneğin, aşağıdaki liste geçerli bir listedir:

```
[1, 2.718, 'abc', [5, 6, 7]]
```

# Listeler

- **Listeler** için de **String** metinleri için olduğu gibi aynı şekilde çalışan bir yapı vardır.
- **len:** L kümesindeki öge sayısı len(L) ile verilir.
- **in:** `in` operatörü, bir listenin bir şey içerip içermediğini gösterir. Örnekler:

```
1 L = [1,2,3,4]
2 if 2 in L:
3     print('Liste 2 içeriyor.')
4 if 0 not in L:
5     print('Listede hiç 0 yok.')
```

- İndeksleme ve dilimleme — Bunlar tıpkı dizelerde olduğu gibi çalışır.
- Örneğin, L[0], L listesinin ilk ögesidir ve L[:3] ilk üç ögeyi verir.
- **index** ve **count** — Bu yöntemler, **stringler** için olduğu gibi aynı şekilde çalışır.
- **+**, **\*** ve **+** operatörü bir listeyi diğerinin sonuna ekler. **\*** operatörü ise bir listeyi tekrarlar. Örnekler:

Expression	Result
[7, 8]+[3, 4, 5]	[7, 8, 3, 4, 5]
[7, 8]*3	[7, 8, 7, 8, 7, 8]
[0]*5	[0, 0, 0, 0, 0]

- Son örnek, özellikle sıfırlardan oluşan bir listeyi hızlıca oluşturmak için kullanışlıdır.

# Listeler

- Döngüler — Dizeler için geçerli olan iki döngü türü, listeler için de geçerlidir. Aşağıdaki iki örnek de bir listenin öğelerini tek tek, ayrı satırlara yazdırır.

```
1 L = [1, 4, 'a', 'x']
2
3 for i in range(len(L)):
4     print(L[i])
```

```
1 L = [1, 4, 'a', 'x']
2
3 for item in L:
4     print(item)
```

- Sol döngü, döngü değişkeni i'yi kullanarak döngünün neresinde olduğunuzu takip etmeniz gereken problemler için kullanışlıdır. Eğer buna gerek yoksa, biraz daha basit olan sağ döngüyü kullanılabilir.
- **Dahili Fonksiyonlar (Built-in functions)**
- Listeler üzerinde çalışan çeşitli yerleşik işlevler mevcuttur. Kullanışlı fonksiyonlar:

Function	Description
<code>len</code>	returns the number of items in the list
<code>sum</code>	returns the sum of the items in the list
<code>min</code>	returns the minimum of the items in the list
<code>max</code>	returns the maximum of the items in the list

- Örnek, aşağıdaki kod bloğu L listesindeki değerlerin ortalamasını hesaplar:

```
average = sum(L) / len(L)
```

# Liste Fonksiyonları

- Bazı listeleme yöntemleri:

Method	Description
<code>append(x)</code>	adds <code>x</code> to the end of the list
<code>sort()</code>	sorts the list
<code>count(x)</code>	returns the number of times <code>x</code> occurs in the list
<code>index(x)</code>	returns the location of the first occurrence of <code>x</code>
<code>reverse()</code>	reverses the list
<code>remove(x)</code>	removes first occurrence of <code>x</code> from the list
<code>pop(p)</code>	removes the item at index <code>p</code> and returns its value
<code>insert(p, x)</code>	inserts <code>x</code> at index <code>p</code> of the list

- Liste metotları ve string metotları arasında farklılık vardır: String metotları orijinal stringi değiştirmez, ancak liste metotları orijinal listeyi değiştirir.

<i>wrong</i>	<i>right</i>
<code>s.replace('X', 'x')</code>	<code>s = s.replace('X', 'x')</code>
<code>L = L.sort()</code>	<code>L.sort()</code>

- Listeleme için farklı yöntem de mevcuttur. Bunlarla ilgili dokümantasyona ulaşmak için Python'da **help(list)** yazılabilir.

# Listeler

- Listelerin kopyalarını oluşturma Python'ın listeleri ele alma şekli nedeniyle listelerin kopyalarını oluşturmak karmaşıktır. Bir L listesi ve bu listenin bir kopyasını oluşturup ona M adını verilmek istensin. İleriki bölümlerde ele alınan nedenlerden dolayı  $M=L$  ifadesi işe yaramayacaktır. Bunun yerine,  $M=L$  yerine aşağıdaki yöntem ile kopyalama yapılabilir.
- $M = L[:]$
- **Listeleri Değiştirme** Bir listedeki belirli bir öğeyi değiştirmek, stringlerdeki değiştirmekten daha kolaydır. L listesinin 2. konumundaki değeri 100 olarak değiştirmek için,  $L[2]=100$  dememiz yeterlidir. Mevcut değeri üzerine yazmadan 2. konuma 100 değerini eklemek istiyorsak, insert yöntemini kullanabiliriz. Bir listeden bir öğeyi silmek için del operatörünü kullanabiliriz. Aşağıda bazı örnekler gösterilmiştir. Her işlem için  $L=[6,7,8]$  olduğunu varsayalım.

Operation	New L	Description
$L[1]=9$	$[6, 9, 8]$	replace item at index 1 with 9
$L.insert(1, 9)$	$[6, 9, 7, 8]$	insert a 9 at index 1 without replacing
$del L[1]$	$[6, 8]$	delete second item
$del L[:2]$	$[8]$	delete first two items

# Alıştırmalar

- **Örnek 1:** 1 ile 100 arasında rastgele 50 sayıdan oluşan bir L listesi üreten bir program yazın.

```
from random import randint
L = []
for i in range(50):
    L.append(randint(1,100))
print(L)
```

- Boş liste [] ile başlayarak listeye tek tek öğeler eklemek için `append` yöntemini kullanıyoruz. `append` yöntemine alternatif olarak aşağıdaki yöntem kullanılabilir:

```
L = L + [randint(1,100)]
```

# Alıştırmalar

- **Örnek 2:** L listesindeki her elemanı karesiyle değiştirin.

```
1 L = [3,5,7,3,6,89]
2
3 for i in range(len(L)):
4     L[i] = L[i]**2
5
6 print(L)
```

- **Örnek 3:** L listesindeki 50'den büyük olan öğelerin sayısını sayın.

```
1 L = [3,5,9,70,23,76]
2
3 count = 0
4 for i in range(len(L)):
5     if L[i] > 50:
6         count= count + 1
7 print(count)
```

```
1 L = [3,5,9,70,23,76]
2
3 count = 0
4 for i in L:
5     if i > 50:
6         count= count + 1
7 print(count)
```

# Alıştırmalar

- **Örnek 4:** "Scores" adlı listedeki en büyük iki ve en küçük iki elemanı yazdıran bir program yazın.

```
1  scores = [3,5,9,70,23,9,76]
2
3  scores.sort()
4  print('İki en küçük: ', scores[0], scores[1])
5  print('İki en büyük: ', scores[-1], scores[-2])
```

# Alıştırmalar

- **Örnek 5:** Basit bir bilgi yarışması oyunu oynamak için bir program.

```
1  dogru_sayisi = 0
2
3  # Soru 1
4  print('Fransa\'nin başkenti nedir?', end=' ')
5  tahmin = input()
6
7  if tahmin.lower() == 'paris':
8      print('Doğru!')
9      dogru_sayisi += 1
10 else:
11     print('Yanlış. Cevap Paris.')
12     print('Şu ana kadar', dogru_sayisi, 'doğru var')
13
14 # Soru 2
15 print('Hangi eyaletin sadece bir komşusu vardır?', end=' ')
16 tahmin = input()
17
18 if tahmin.lower() == 'maine':
19     print('Doğru!')
20     dogru_sayisi += 1
21 else:
22     print('Yanlış. Cevap Maine.')
23     print('Şu ana kadar', dogru_sayisi, 'doğru var')
```

```
1  sorular = [
2      'Fransa\'nin başkenti nedir? ',
3      'Hangi eyaletin sadece bir komşusu vardır? '
4  ]
5
6  cevaplar = ['Paris', 'Maine']
7
8  dogru_sayisi = 0
9
10 for i in range(len(sorular)):
11     tahmin = input(sorular[i])
12
13     if tahmin.lower() == cevaplar[i].lower():
14         print('Doğru')
15         dogru_sayisi = dogru_sayisi + 1
16     else:
17         print('Yanlış. Doğru cevap:', cevaplar[i])
18
19     print('Şu ana kadar', dogru_sayisi, 'doğru var')
```

- Alternatif cevap olarak liste kullanımında, bir soruyu değiştirmek, bir soru eklemek veya sırayı değiştirmek için yalnızca sorular ve cevaplar listelerinin değiştirilmesi gerekmesidir. Ayrıca, programda bir değişiklik yapmak istiyorsanız, örneğin kullanıcıya doğru cevabı söylememek gibi, program boyunca yayılmış yirmi satır yerine yalnızca tek bir satırı değiştirmeniz yeterlidir.



# Sorular

- **Soru 1:** Kullanıcıdan tamsayı listesi girmesini isteyen bir program yazın. Aşağıdakileri yapın:
  - (a) Listedeki toplam öğe sayısını yazdırın.
  - (b) Listedeki son öğeyi yazdırın.
  - (c) Listeyi ters sırada yazdırın.
  - (d) Liste 5 içeriyorsa "Evet", aksi takdirde "Hayır" yazdırın.
  - (e) Listedeki 5 sayısını yazdırın.
  - (f) Listeden ilk ve son öğeleri kaldırın, kalan öğeleri sıralayın ve sonucu yazdırın.

# Sorular

## ▪ Soru 1:

```
1  L = []
2  for i in range(5):
3      num = int(input("Bir tamsayı girin: "))
4      L.append(num)
5  print("Toplam öğe sayısı:", len(L))
6  print("Son öğe:", L[-1])
7  print("Liste ters sırada:", L[::-1])
8  if 5 in L:
9      print("Evet")
10     print("5 sayısı:", L.count(5))
11 else:
12     print("Hayir")
13 # Listeden ilk ve son öğeleri kaldırın, kalan öğeleri sıralayın ve sonucu yazdırın.
14 if len(L) > 2:
15     L.pop(0) # İlk öğeyi kaldır
16     L.pop(-1) # Son öğeyi kaldır
17     L.sort() # Kalan öğeleri sırala
18     print("Kalan öğeler sıralı:", L)
19 else:
20     print("Liste çok kısa")
```



# Sorular

- **Soru 2:** 1 ile 100 arasında rastgele 20 sayı üreten bir program yazın.
  - (a) Listeyi yazdırın.
  - (b) Listedeki elemanların ortalamasını yazdırın.
  - (c) Listedeki en büyük ve en küçük değerleri yazdırın.
  - (d) Listedeki ikinci en büyük ve ikinci en küçük elemanları yazdırın.
  - (e) Listedeki çift sayıların sayısını yazdırın.

# Sorular

## ▪ Soru 2:

```
1 import random
2
3 # 1 ile 100 arasında 20 rastgele sayı üret
4 liste = []
5 for i in range(20):
6     sayi = random.randint(1, 100)
7     liste.append(sayi)
8
9 # (a) Listeyi yazdır
10 print("Liste:", liste)
11
12 # (b) Ortalama
13 toplam = 0
14 for x in liste:
15     toplam += x
16 ortalama = toplam / len(liste)
17 print("Ortalama:", ortalama)
18
19 # (c) En büyük ve en küçük
20 en_buyuk = max(liste)
21 en_kucuk = min(liste)
22 print("En büyük:", en_buyuk)
23 print("En küçük:", en_kucuk)
24
25 # (d) İkinci en büyük ve ikinci en küçük
26 sirali = sorted(liste)
27 print("İkinci en küçük:", sirali[1])
28 print("İkinci en büyük:", sirali[-2])
29
30 # (e) Çift sayıların sayısı
31 cift_sayisi = 0
32 for x in liste:
33     if x % 2 == 0:
34         cift_sayisi += 1
35
36 print("Çift sayı adedi:", cift_sayisi)
```



# Sorular

- **Soru 3:** [8,9,10] listesiyle başlayın. Aşağıdakileri yapınız:
- (a) İkinci elemanı (indeks 1) 17 olarak ayarlayın
- (b) Listenin sonuna 4, 5 ve 6 ekleyin
- (c) Listedeki ilk elemanı kaldırın
- (d) Listeyi sıralayın
- (e) Listeyi ikiye katlayın
- (f) 3. indekse 25 ekleyin

# Sorular

## ▪ Soru 3:

```
1  # Başlangıç listesi
2  L = [8, 9, 10]
3  print("Başlangıç:", L)
4
5  # (a) İkinci elemanı 17 yap
6  L[1] = 17
7  print("(a):", L)
8
9  # (b) Sonuna 4, 5, 6 ekle (append ile)
10 L.append(4)
11 L.append(5)
12 L.append(6)
13 print("(b):", L)
14
15 # (c) İlk elemanı kaldır
16 L.pop(0)
17 print("(c):", L)
18
19 # (d) Listeyi sırala
20 L.sort()
21 print("(d):", L)
22
23 # (e) Listeyi ikiye katla
24 L = L * 2
25 print("(e):", L)
26
27 # (f) 3. indekse 25 ekle
28 L = L[:3] + [25] + L[3:]
29 print("(f):", L)
```