

BMÜ-111 Algoritma ve Programlama

Bölüm 5

Tek Boyutlu Diziler

Yrd. Doç. Dr. İlhan AYDIN

Problem

100 adet sayı okumak istediğimizi düşünelim. Bu sayıların ortalaması hesaplanacak ve sayıların kaç tanesinin ortalamadan büyük olduğu bulunacaktır.

Özet

Dizi: belirli sayıda ve aynı veri türünden değişkenlere aynı adla erişilmesini sağlayan bir yapıdır.

Dizilerde döngü işlemleri sıklıkla kullanılır. Özellikle for döngüleri dizilerle kullanılmaya çok uygundur.

Diziler tek boyutlu olabileceği gibi iki veya daha çok boyutlu da olabilirler. Matrisler iki boyutlu dizilere örnektir.

Dizi isimleri ile [] kullanımı

1. Yeni bir tip adı oluşturmak için, örneğin `int[] degerler;` “int dizisi” türünde bir isim yaratır
 - `int` ve `int dizisi` farklı türlerdir
 - `int dizisi` ismin türünü belirtir, verinin türünü değil
2. Yeni bir dizi oluşturmak için: `degerler= new int[100];`
`k = new int[3];`
`yt = new float[7];`
`isimler = new String[50];`
3. Dizi içindeki bir elemanı isimlendirmek için
 - indexlenmiş değişken de denir

```
degerler[3] = giris.nextInt();  
System.out.println("Dizinin 3. elemanı" +  
degerler[2]);
```

Dizi terminolojisi

`sicaklik[n + 2]`

Dizi adı

`sicaklik` $\overbrace{[n + 2]}$

Index - altsimge
- int olmalıdır,

`sicaklik[n + 2]`

Indexli değişken – eleman veya altsimgeli değişken

indexli değişkenin değeri
- dizinin bir elemanı olarak da
adlandırılır

`sicaklik[n + 2] = 32;`

Dizilerin uzunluğu

- Bir dizinin uzunluğu `new` kullanılarak oluşturulduğu zaman parantez içinde verilir.
 - dizinin elemanlarına (değerlerine) ne kadar hafıza ayrılacağını belirlemek için ve
 - dizinin en fazla ne kadar eleman alabileceğini belirlemek için kullanılır
 - elemanlara değer atanıp atanmamasına bakılmaksızın hafızada yer ayrılır
- `length`, değişkeni kullanılarak dizinin uzunluğu öğrenilebilir :

```
int[] sicaklik = new int[20];  
System.out.println(sicaklik.length);
```
- Dizinin uzunluğu tanımlama esnasında belirtilir ve daha sonra dizi yeniden tanımlanmadığı sürece değiştirilemez.

Alt-index

Diziler sıfırdan başlayan altindexler kullanırlar

- ilk elemanın indexi 0
- ikinci elemanın indexi 1
- n. elemanın indexi n-1
- son elemanın indexi `length-1`

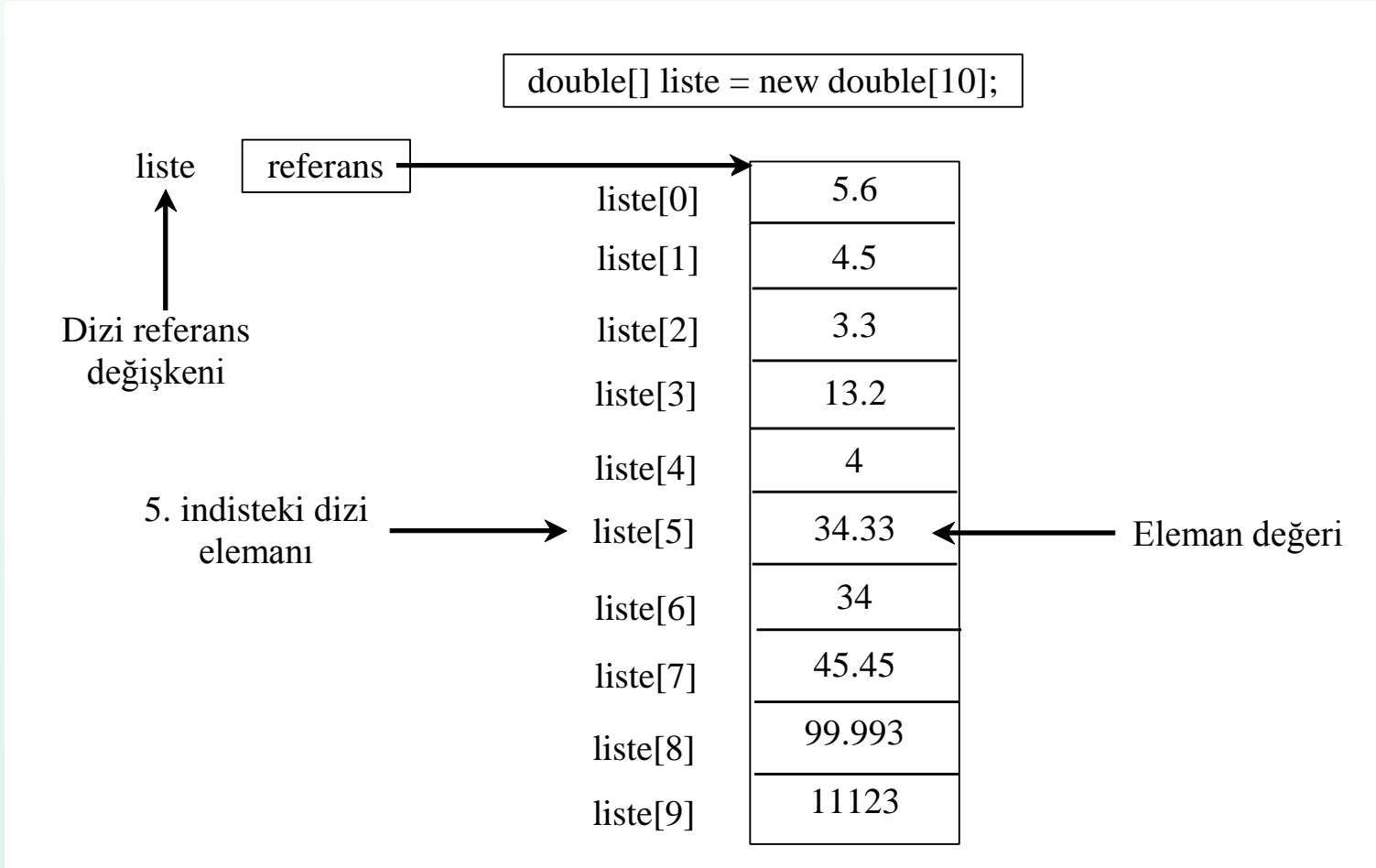
Örnek:

```
int[] scores = {97, 86, 92, 71};
```

index:	0	1	2	3
deger:	97	86	92	71

Dizilerin Tanımlanması

Dizi aynı türden verilerin bir koleksiyonunu tutan bir veri yapısıdır.



Tanımlama ve oluşturmanın tek adımda yapılması

```
veritipi[] diziAdi = new  
veritipi[arraySize];
```

```
double[] dizi = new double[10];
```

```
veritipi diziadi[] = new  
dizitipi[arraySize];
```

```
double liste[] = new double[10];
```

Varsayılan değerler

Bir dizi oluşturulduğunda elemanları varsayılan olarak atanır.

Numerik veri türleri için 0,

Char veri türleri için '\u0000',

Boolean veri türleri için false.

İndislenmiş değişkenler

Dizi elemanlarına indis ile ulaşılır. Dizi indisi 0'dan başlar ve diziadi.length-1'e kadar gider.

Dizideki her eleman aşağıdaki yazım biçimi ile gösterilir :

```
diziAdi[indis];
```

İndisli değişkenlerin kullanımı

Bir dizi oluşturulduktan sonra bir indisli değişken düzenli bir değişken olarak kullanılabilir. Örneğin, aşağıdaki kod liste[0] ve liste[1]'i toplayıp sonucu liste[2]'ye yazar.

```
liste[2] = liste1[0] + liste[1];
```

Dizi Başlatmaları

Tanımlama ve oluşturmanın tek adımda yapılması:

```
double[] listem = {1.9, 2.9, 3.4, 3.5};
```

Yukarıdaki tanımlama aşağıdaki işlemi yapar.

```
double[] listem = new double[4];
```

```
listem[0] = 1.9;
```

```
listem[1] = 2.9;
```

```
listem[2] = 3.4;
```

```
listem[3] = 3.5;
```

Dikkat

Tek adımda tanımlama ve değer atama yapılabilir.
Fakat tanımladıktan sonra aşağıdaki gibi atama ifadesi yanlıştır.

```
double[] listem;
```


```
listem = {1.9, 2.9, 3.4, 3.5};
```

Diziler ile ilgili örnek

Degerler adında bir dizi tanımlar

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Dizi oluşturulduktan sonra



0	0
1	0
2	0
3	0
4	0

Diziler ile ilgili örnek

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

i = 1

Dizi oluşturulduktan sonra

0	0
1	0
2	0
3	0
4	0

Diziler ile ilgili örnek

i (=1) 5'ten küçük mu? true

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Dizi oluşturulduktan sonra

0	0
1	0
2	0
3	0
4	0

Diziler ile ilgili örnek

Bu satır çalıştıktan sonra değerler[1]=1

```
public class Test {  
    public static void main(String[] arg.  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Bu adımdan sonra

0	0
1	1
2	0
3	0
4	0

Diziler ile ilgili örnek

i++ dan sonra, i =2

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Bu adımdan sonra

0	0
1	1
2	0
3	0
4	0

Diziler ile ilgili örnek

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] +  
            degerler[4];  
    }  
}
```

i (= 2) ve 5'ten küçük mu? true

Bu adımdan sonra

0	0
1	1
2	0
3	0
4	0

Diziler ile ilgili örnek

Bu adım çalıştıktan sonra,
 $\text{degerler}[2] = 3 \ (2 + 1)$

```
public class Test {  
    public static void main(String[] arg.  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

İkinci adımdan sonra

0	0
1	1
2	3
3	0
4	0

Diziler ile ilgili örnek

Bu adımdan sonra $i=3$

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Bu adımdan sonra

0	0
1	1
2	3
3	0
4	0

Diziler ile ilgili örnek

i (=3) 3<5 true

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

İkinci adımdan sonra

0	0
1	1
2	3
3	0
4	0

Diziler ile ilgili örnek

Bu satırdan sonra, $\text{degerler}[3] = 6$ ($3 + 3$)

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Üçüncü adımdan sonra

0	0
1	1
2	3
3	6
4	0

Diziler ile ilgili örnek

Bu adımdan sonra $i=4$

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Üçüncü adımdan sonra

0	0
1	1
2	3
3	6
4	0

Diziler ile ilgili örnek

i (=4) 5'ten küçük mu?true

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Ucuncu adımdan sonra

0	0
1	1
2	3
3	6
4	0

Diziler ile ilgili örnek

Bu adımdan sonra, $\text{degerler}[4] = 10$ ($4 + 6$)

```
public class Test {  
    public static void main(String[] args,  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Dördüncü adımdan sonra

0	0
1	1
2	3
3	6
4	10

Diziler ile ilgili örnek

i++ işleminden sonra i=5

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

Dördüncü adımdan sonra

0	0
1	1
2	3
3	6
4	10

Diziler ile ilgili örnek

$i (=5) < 5$? false . Donguden cik

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```

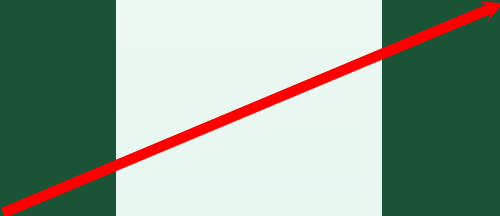
Dorduncu adımdan sonra

0	0
1	1
2	3
3	6
4	10

Diziler ile ilgili örnek

Bu satırdan sonra, `degerler[0] = 11 (1 + 10)`

```
public class Test {  
    public static void main(String[] args) {  
        int[] degerler = new int[5];  
        for (int i = 1; i < 5; i++) {  
            degerler[i] = i + degerler[i-1];  
        }  
        degerler[0] = degerler[1] + degerler[4];  
    }  
}
```



0	11
1	1
2	3
3	6
4	10

Dizilerin işlenmesi

1. (Değerler dizisine klavyeden giriş)
2. (Rastgele değerler ile değerler dizisini başlatma)
3. (Dizi elemanlarını ekrana yazdırma)
4. (Elemanların toplamı)
5. (En büyük elemanı bulma)
6. (En büyük elemanın en düşük indisini bulma)
7. (*Rastgele değiştirme*)
8. (*Elemanları kaydırma*)

Degerler dizisine klavyeden giriş

```
java.util.Scanner input = new java.util.Scanner(System.in);  
System.out.print(" listem dizisine" + listem.length + " eleman gir ");  
for (int i = 0; i < listem.length; i++)  
    listem[i] = input.nextDouble();
```


Rastgele değerler ile listem dizisini başlatma

```
for (int i = 0; i < listem.length; i++) {  
    listem[i] = Math.random() * 100;  
}
```

Dizi elemanlarını ekrana yazdırma

```
for (int i = 0; i < listem.length; i++) {  
    System.out.print(listem[i] + " ");  
}
```

Elemanların toplamı

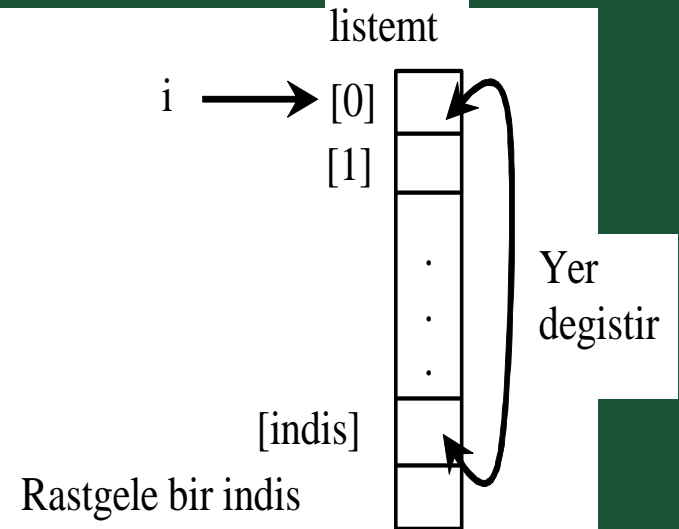
```
double top = 0;  
for (int i = 0; i < listem.length; i++) {  
    top += listem[i];  
}
```

En büyük elemanı bulma

```
double maks = listem[0];  
for (int i = 1; i < listem.length; i++) {  
    if (listem[i] > maks) maks = listem[i];  
}
```

Rastgele deęiřtirme

```
for (int i = 0; i < listem.length; i++) {  
    // Rastgele indis uret  
    int index = (int) (Math.random()  
        * listem.length);  
  
    // listem[i] ile listem[j] I yer deęiřtir  
    double temp = listem[i];  
    listem[i] = listem[index];  
    listem[index] = temp;  
}
```



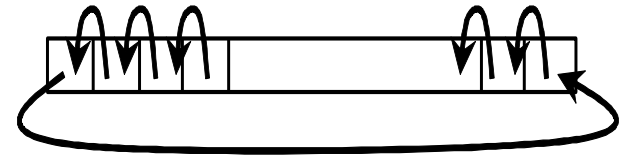
Elemanların kaydırılması

```
double temp = listem[0]; // ilk elemanı ata

// Elemanları sola kaydır
for (int i = 1; i < listem.length; i++) {
    listem[i - 1] = listem[i];
}

// ilk elemanı sona taşı
listem[listem.length - 1] = temp;
```

listem



Genişletilmiş for döngüsü (for-each Döngüsü)

JDK 1.5 ile yeni bir for döngüsü geliştirilmiştir. Bu döngü indis değişkeni kullanmadan bütün elemanlara ardışık olarak ulaşmanızı sağlar. :

```
for (double deger: listem)
    System.out.println(deger);
```

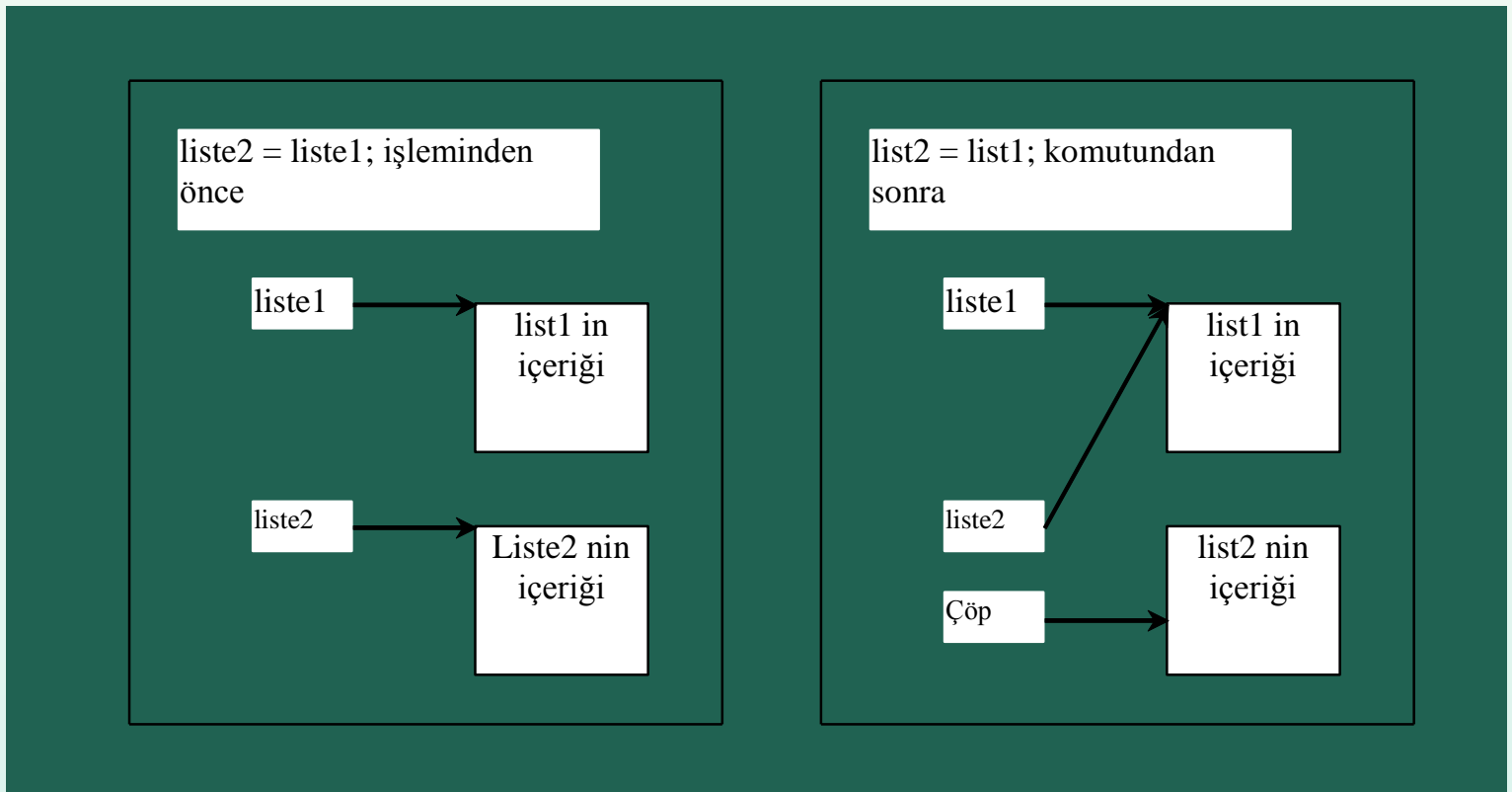
Genel yazım bicimi

```
for (tip degsikenadi: dizi adi) {
    // değişkeni isle
}
```

Dizilerin kopyalanması

Bir programda sık sık bir dizinin tamamını veya bir bölümünü kopyalamanız gerekebilir. Bu durumda = operatörünü kullanabilirsiniz

```
liste2 = liste1;
```



Dizilerin kopyalanması

Döngü ile:

```
int[] liste1= {2, 3, 1, 5, 10};  
int[] liste2= new int[liste1.length];  
  
for (int i = 0; i < liste1.length; i++)  
    liste2[i] = liste1[i];
```

Diziyi ters çevirme örneği

```
int[] liste1 = {1, 2, 3, 4, 5, 6};
```

```
int[] sonuc = reverse(liste1);
```

Sonuc dekkisenini tanimla

```
int[] sonuc = new int[list1.length];
```

```
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}
```

```
}
```

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	0
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
         i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

i = 0 ve j = 5

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	0
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[list1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < list1.length; i++, j--) {  
    sonuc[j] = list1[i];  
}  
  
return sonuc;  
}
```

i (= 0) < 6 mı? true

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	0
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
  
}
```

i = 0 and j = 5
sonuc[5]=liste1 [0]

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	1
---	---	---	---	---	---

Diziyi ters çevirme örneği.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};  
int[] sonuc;
```

```
int[] sonuc = new int[list1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < list1.length; i++, j--) {  
    sonuc[j] = list1[i];  
}  
  
return sonuc;  
}
```

Bu adımdan sonra, $i=1$ ve $j=4$

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};  
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
  
return sonuc;  
}
```

i (=1) < 6? true

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	0	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
}
```

i = 1 ve j = 4
sonuc[4] = liste1[1]

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
  
}
```

i=2 ve j=3

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list1.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
        i < liste1.length; i++, j--) {  
        sonuc[j] = liste1[i];  
    }  
  
    return sonuc;  
}
```

i (=2) < 6? true

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	0	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
        i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

i = 2 and j = 3
sonuc[j]=liste[i]

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
  
}
```

i=3 ve j=2

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};  
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
    i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}
```

i (=3) < 6 ? true

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	0	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
}
```

i = 3 ve j = 2
sonuc[j]=liste1[i]

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
         i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

Bu adımdan sonra i=4 ve
j=1

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
        i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

i (=4) < 6? true

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	0	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
        i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

i = 4 ve j = 1
sonuc[j]=liste1[i]

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};  
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
}
```

Bu adımdan sonra, i=5 ve j=0

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};  
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}
```

i (=5) < 6? true

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

0	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
}
```

i = 5 ve j = 0
sonuc[j]=liste1[i]

list

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

6	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
int[] sonuc = new int[liste1.length];  
  
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}  
  
}
```

i=6 ve j=-1 olur

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

6	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

```
public static int[] reverse(int[] list) {  
    int[] sonuc = new int[list.length];  
  
    for (int i = 0, j = sonuc.length - 1;  
        i < list.length; i++, j--) {  
        sonuc[j] = list[i];  
    }  
  
    return sonuc;  
}
```

i (=6) < 6 ? false.
Döngüden çık

liste1

1	2	3	4	5	6
---	---	---	---	---	---

sonuc

6	5	4	3	2	1
---	---	---	---	---	---

Diziyi ters çevirme örneği, cont.

```
int[] liste1 = new int[]{1, 2, 3, 4, 5, 6};
```

```
int[] sonuc;
```

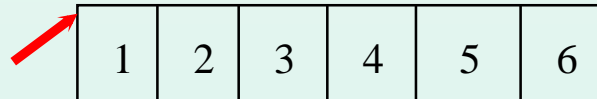
```
int[] sonuc = new int[liste1.length];
```

```
for (int i = 0, j = sonuc.length - 1;  
     i < liste1.length; i++, j--) {  
    sonuc[j] = liste1[i];  
}
```

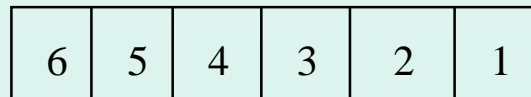


Döngü sonu

liste1



sonuc



Dizilerde Arama

Arama verilen bir elemanın dizide var olup olmadığına bakılması işlemidir

Bu işlem bilgisayar programlamada ortak bir görevdir.

Burada iki yöntemden bahsedeceğiz:

Ardışık arama

İkili arama

Ardışık arama

Aranan

Dizi

3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8
3	6	4	1	9	7	3	2	8

Ardışık arama

```
1  package Uygulama1;
2  import java.util.Scanner;
3  public class Uygulama1 {
4      public static void main(String[] args) {
5          int []dizi={1,2,10, 15, 25,3,-1};
6          for(int eleman:dizi)
7              System.out.print(eleman+" ");
8          int aranan,indis=-1;
9          System.out.println("Aranacak elemanı gir");
10         Scanner klavye=new Scanner(System.in);
11         aranan=klavye.nextInt();
12         for (int i = 0; i < dizi.length; i++) {
13             if(dizi[i]==aranan) {
14                 indis=i; break;
15             }
16         }
17         if(indis!=-1) System.out.println(aranan+" dizide "+
18             indis+" pozisyonunda bulundu");
19         else System.out.println("aranan bulunamadı");
20     }
21 }
22 }
```

İkili Arama

İkili arama sıralı diziler üzerinde çalışır.

Dizi küçükten veya büyükten küçüğe büyüğe sıralıdır..

örneğin, 2 4 7 10 11 45 50 59 60 66 69 70 79

İkili arama ilk olarak aranan elemanı dizinin ortasındaki eleman ile karşılaştırılır.

Eğer aranan ortadan küçük ise, dizinin ilk yarısından arama devam eder..

Eğer eşit ise arama eşleşme ile sonlanır.

Eğer eleman orta elemandan büyükse, dizinin ikinci yarısında elemanı ara.

İkili Arama

Aranan

Liste1

8	1	2	3	4	6	7	8	9
8	1	2	3	4	6	7	8	9
8	1	2	3	4	6	7	8	9

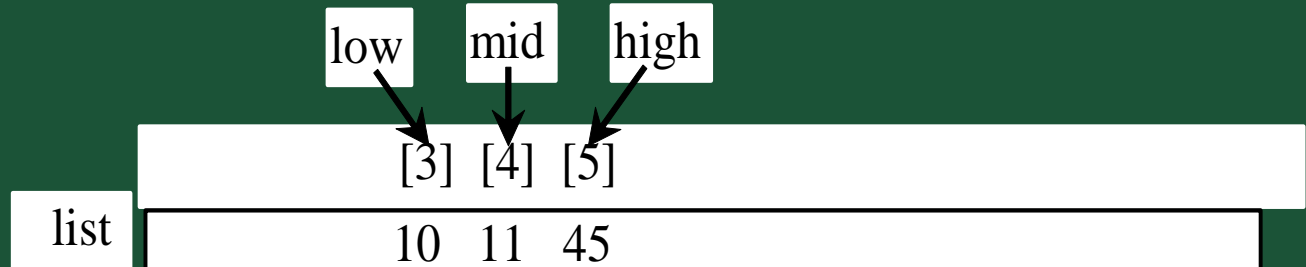
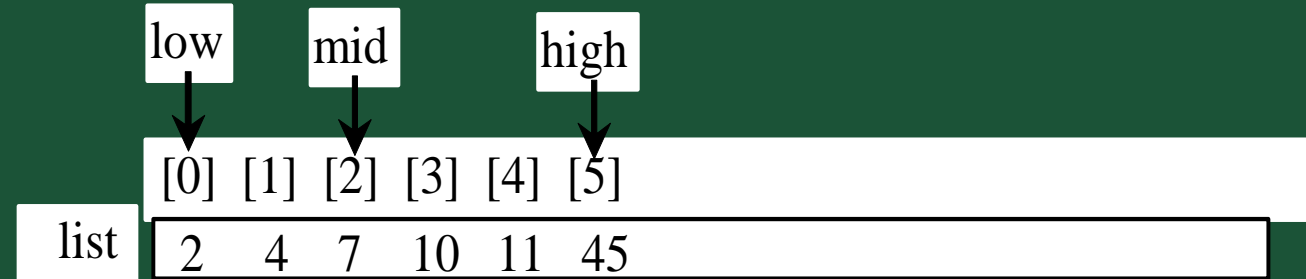
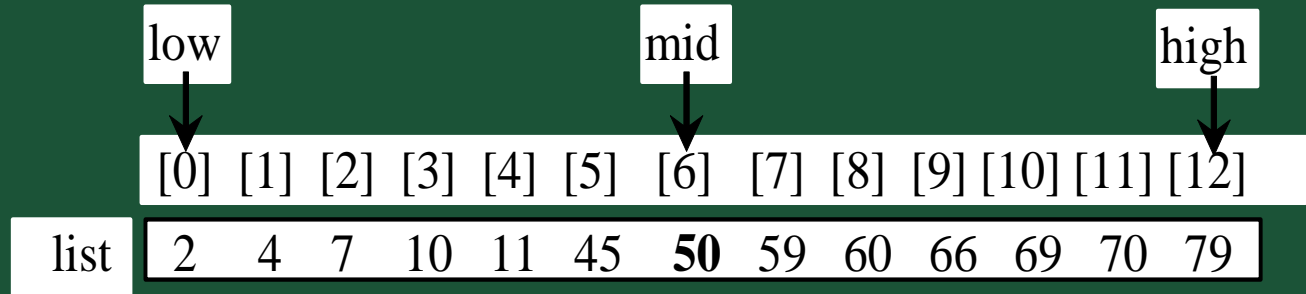
İkili Arama

key = 11

key < 50

key > 7

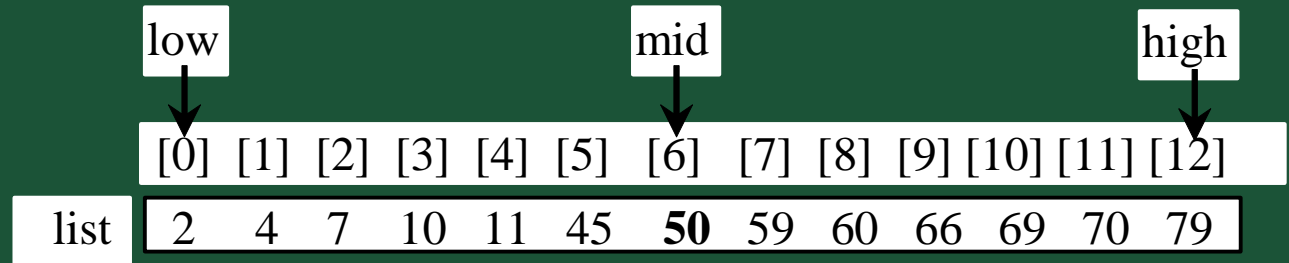
key == 11



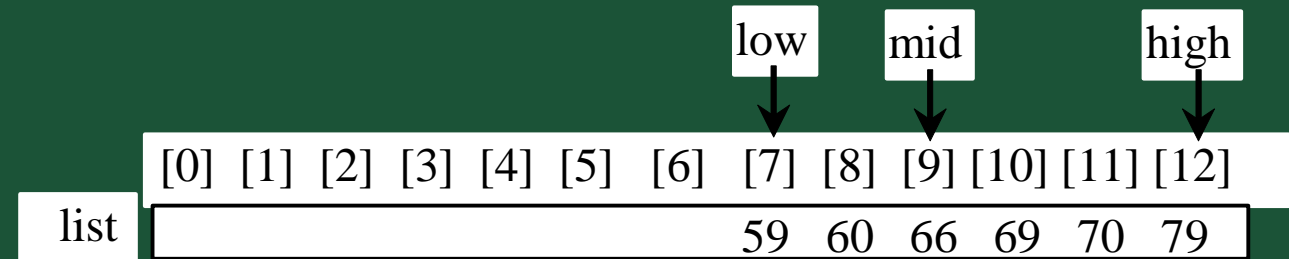
İkili Arama

key = 54

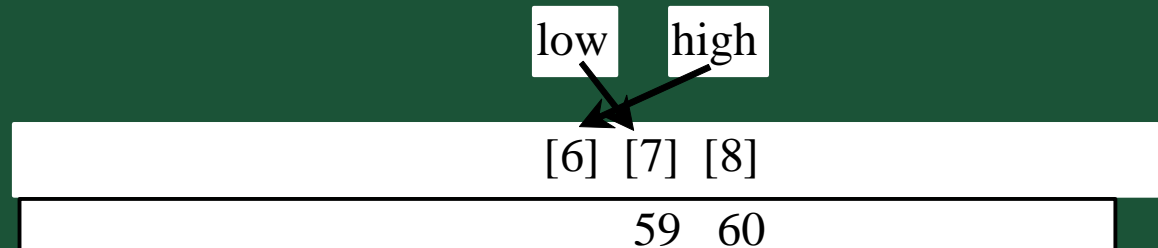
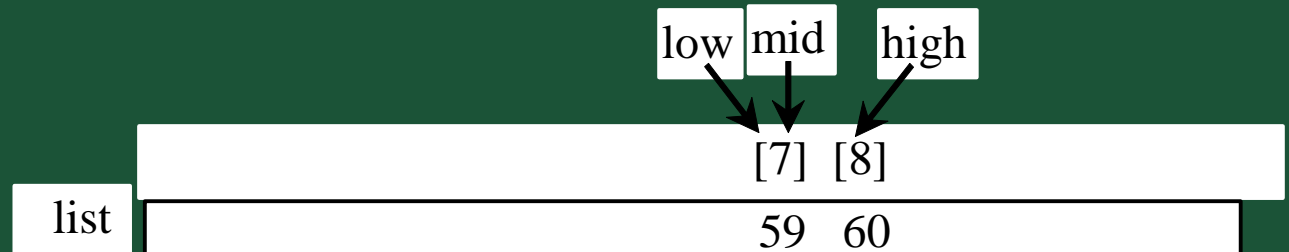
key > 50



key < 66



key < 59



İkili Arama

```
3 public class Uygulama1 {
4     public static void main(String[] args) {
5         int []dizi={1,2,10,15,25,30,41};
6         for(int eleman:dizi)
7             System.out.print(eleman+" ");
8         int aranan,indis=-1,bas=0, orta,son=dizi.length-1;
9         System.out.println("Aranacak elemanı gir:");
10        Scanner klavye=new Scanner(System.in);
11        aranan=klavye.nextInt();
12        while(son>=bas)
13        {
14            orta=(son+bas)/2;
15            if(aranan<dizi[orta]) son=orta-1;
16            else if(aranan>dizi[orta]) bas=orta+1;
17            else if(aranan==dizi[orta]){
18                indis=orta; break; }
19        }
20        if(indis!=-1) System.out.println(aranan+" dizide "+
21            indis+" pozisyonunda bulundu");
22        else System.out.println("aranan bulunamadı");
23    }
24 }
```