

# İkili Arama

→ İkili Arama (Binary Search) Algoritması, bir sıralı dizide belirli bir elemanın bulunması için kullanılan bir algoritmadır. Bu algoritma, dizinin ortasındaki elemanı hedeflenen elemanla karşılaştırarak arama işlemini gerçekleştirir ve böylece dizinin yarısını elemine ederek arama işlemini hızlandırır. İkili arama algoritması şu adımlarla gerçekleşir:

**1- İlk Adım:** Başlangıçta, arama yapılacak olan sıralı dizi soldan başlayarak sıralanmıştır.

**2- İkinci Adım:** Dizinin ortasındaki eleman bulunur ve hedeflenen eleman ile karşılaştırılır.

**3- Üçüncü Adım:** Hedeflenen eleman ortadaki elemana eşit ise, arama işlemi başarıyla sonuçlanmıştır ve ortadaki elemanın dizindeki konumu (indisi) döndürülür.

**4- Dördüncü Adım:** Hedeflenen eleman ortadaki elemandan küçük ise, arama işlemi soldaki yarı dizide tekrarlanır.

**5- Beşinci Adım:** Hedeflenen eleman ortadaki elemandan büyük ise, arama işlemi sağdaki yarı dizide tekrarlanır.

**6- Son Adım:** Bu işlem, hedeflenen eleman bulunana kadar tekrarlanır veya dizinin tüm elemanları taranana kadar devam eder. Eğer hedef eleman bulunmaz ise “-1” döndürülerek işlem bitirilir.

**Zaman Karmaşıklığı:** İkili Arama Algoritması' nın zaman karmaşıklığı  $O(\log n)$  'dir. Bu, algoritmanın, verilen dizinin boyutuna bağlı olarak, arama işlemini gerçekleştirmek için logaritmik bir zaman karmaşıklığına sahip olduğu anlamına gelir.

**Not:** İkili Arama Algoritması, sıralı bir dizide hedeflenen elemanı bulmak için kullanılan etkili bir algoritmadır ve zaman karmaşıklığı logaritmik olduğundan, büyük veri kümelerinde hızlı bir şekilde arama yapmak için tercih edilir.

### Örnek:

Dizi: 4 8 10 15 18 21 24 27 29 33 34 37 39 41 43

Hedef Değer: 18

### Başlangıç:

Dizi:	4	8	10	15	18	21	24	27	29	33	34	37	39	41	43
	^							^							^
	Başlangıç							Orta							Son

Başlangıç İndisi: 0

Son İndis: 14

Ortanca İndis: 7 (Hedef Değerden Büyük)

### 1. Adım:

Dizi:	4	8	10	15	18	21	24	27	29	33	34	37	39	41	43
	^			^			^								
	Başlangıç			Orta			Son								

Başlangıç İndisi: 0

Son İndis: 6

Ortanca İndis: 3 (Hedef Değerden Küçük)

## 2. Adım:

Dizi: 4 8 10 15 18 21 24 27 29 33 34 37 39 41 43

^ ^ ^

| | |

Baş Ort Son

Başlangıç İndisi: 4

Son İndis: 6

Ortanca İndis: 5 (Hedef Değerden Büyük)

## 3. Adım:

Dizi: 4 8 10 15 18 21 24 27 29 33 34 37 39 41 43

^

|

Baş, Ort , Son

Başlangıç İndisi: 4

Son İndis: 4

Ortanca İndis: 4 (Hedef Değerden ile Eşleşti)

### Pseudocode Kod:

İkili **Arama** (dizi, hedef):

başlangıç = **0**

son = **uzunluk**(dizi) - **1**

**while** başlangıç <= son:

orta = (başlangıç + son) / **2** # orta elemanın indeksini hesapla

**if** dizi[orta] == hedef:

**return** orta # hedef değeri bulunduğunda orta elemanın indeksini döndür

**if** dizi[orta] < hedef:

başlangıç = orta + **1** # hedef değeri ortadan büyükse sağ yarıyı ele al

**else:**

son = orta - **1** # hedef değeri ortadan küçükse sol yarıyı ele al

**return** **-1** # hedef değeri dizide bulunamadığında -1 döndür

### Python ile Yazarsak:

```
def ikili_arama(dizi, hedef):
```

```
    baslangic = 0
```

```
    son = len(dizi) - 1
```

```
    while baslangic <= son:
```

```
        orta = (baslangic + son) // 2 # orta elemanın indeksini hesapla
```

```
        if dizi[orta] == hedef:
```

```
            return orta # hedef değeri bulunduğunda orta elemanın indeksini döndür
```

```
        if dizi[orta] < hedef:
```

```
            baslangic = orta + 1 # hedef değeri ortadan büyükse sağ yarıyı ele al
```

```
        else:
```

```
            son = orta - 1 # hedef değeri ortadan küçükse sol yarıyı ele al
```

```
    return -1 # hedef değeri dizide bulunamadığında -1 döndür
```