

# Ekleme Sıralaması Algoritması

→ Sıralama algoritmalarından biridir ve temel olarak elemanları doğru konumlarına "yerleştirerek" bir dizi veya listeyi sıralamak için kullanılır. Adından da anlaşılacağı gibi, elemanları doğru konumlarına yerleştirmek için sıralama işlemi sırasında bir "yerleştirme" adımı yapar. Insertion Sort(ekleme sıralaması) algoritması, adım adım ilerler ve her adımda bir elemanı doğru konumuna yerleştirir. Başlangıçta, algoritma sıralanmış bir bölge (soldan başlayarak boş bir bölge) ve sıralanmamış bir bölge (sağda kalan elemanlar) olarak düşünebiliriz. Algoritma, sıralanmamış bölgeden bir eleman alır ve bu elemanı sıralanmış bölgeye yerleştirir, böylece sıralanmış bölgenin boyutu bir birim artar.

## Insertion Sort Algoritmasının Çalışma Adımları Şu Şekildedir:

**1- İlk Adım:** Algoritma sıralanmış bölgenin boyutunu 1 olarak kabul eder ve sıralanmamış bölgedeki ilk elemanı seçer.

**2- İkinci Adım:** Seçilen eleman, sıralanmış bölgedeki elemanlarla karşılaştırılır. Eleman, sıralanmış bölgedeki elemanlardan daha küçük bir değere sahip olduğu sürece, bu elemanların sağa kaydırılması gerekmektedir.

**3- Üçüncü Adım:** Sıralanmış bölgedeki elemanlar sağa kaydırılırken, elemanın doğru konumu bulunur.

**4- Dördüncü Adım:** Elemanın doğru konumu bulunduktan sonra, eleman bu konuma yerleştirilir.

**5- Son Adım:** Sıralanmış bölgenin boyutu bir birim artırılır ve işlem sıralanmamış bölgedeki bir sonraki elemana geçer. Bu adımlar sıralanmamış bölgedeki tüm elemanlar sıralanıncaya kadar tekrarlanır.

**Zaman Karmaşıklığı:** Insertion Sort algoritmasının zaman karmaşıklığı  $O(n^2)$  dir. Bu, n elemanlı bir dizinin sıralanması için ortalama olarak  $n^2$  adımda tamamlanacağı anlamına gelir. En iyi durumda (dizi zaten sıralıysa), zaman karmaşıklığı  $O(n)$  olabilir.

**Not:** Insertion Sort algoritması, daha küçük boyutlu dizilerde (n küçük) ve neredeyse sıralı veya kısmen sıralı dizilerde etkili olabilir. Ancak büyük boyutlu dizilerde diğer, daha verimli sıralama algoritmaları tercih edilebilir.

### Örnek:

Başlangıç Durumu: [5, 2, 4, 6, 1, 3]

#### Adım 1:

Sıralanmış Bölge: [5] / Sıralanmamış Bölge: [2, 4, 6, 1, 3]

#### Adım 2:

Sıralanmış Bölge: [2, 5] / Sıralanmamış Bölge: [4, 6, 1, 3]

#### Adım 3:

Sıralanmış Bölge: [2, 4, 5] / Sıralanmamış Bölge: [6, 1, 3]

#### Adım 4:

Sıralanmış Bölge: [2, 4, 5, 6] / Sıralanmamış Bölge: [1, 3]

#### Adım 5:

Sıralanmış Bölge: [1, 2, 4, 5, 6] / Sıralanmamış Bölge: [3]

#### Adım 6:

Sıralanmış Bölge: [1, 2, 3, 4, 5, 6] / Sıralanmamış Bölge: []

### Pseudocode Kod:

InsertionSort(dizi):

n = dizi.length

for i = 1 to n-1:

deger = dizi[i]

j = i - 1

while j >= 0 and dizi[j] > deger:

dizi[j+1] = dizi[j]

j = j - 1

dizi[j+1] = deger

### Python ile Yazarsak:

```
def insertion_sort(arr):  
  
    n = len(arr)  
  
    for i in range(1, n):  
  
        value = arr[i]  
  
        j = i - 1  
  
        while j >= 0 and arr[j] > value:  
  
            arr[j+1] = arr[j]  
  
            j = j - 1  
  
        arr[j+1] = value
```