

Insertion Sort (Sokuşturma Sıralaması)

Bu sıralama Bubble Sort algoritmasının iyileştirilmiş biçimidir. Zaman karmaşası (time complexity) $O(n^2)$ dir.

Bu algoritmayı açıklayan basit bir örnek verebiliriz.

Masada bir deste oyun kağıdı, sırasız ve sırtları yukarıya doğru duruyor olsun.

- Desteden en üstteki kartı alalım. Onu masaya yüzü görünür biçimde koyalım. Tek kart olduğu için sıralı bir kümedir.
- Sırasız destenin üstünden bir kart daha çekelim. Masadaki ilk çektiğimiz kart ile karşılaştıralım. Gerekirse yerlerini değiştirerek, çektiğimiz iki kartı küçükten büyüğe doğru sıralayalım.
- Sırasız destenin üstünden bir kart daha çekelim. Masaya sıralı dizilen iki kart ile karşılaştıralım. Gerekirse yerlerini değiştirerek çekilen üç kartı küçükten büyüğe doğru sıralayalım.
- Bu işleme sırasız deste bitene kadar devam edelim. Sonunda, oyun kağıtlarını sıralamış oluruz.

Herhangi bir adımda, kağıt destesi *sıralı altdeste* ve *sırasız altdeste* olmak üzere iki altkümeye ayrılmış durumdadır. Her adımda, sırasız desteden bir kart çekip, sıralı destedeki doğru yerine yerleştiriyoruz. Bu iş, sırasız deste bitene kadar devam ediyor.

Bunu başka bir örnek üzerinde düşünelim.

$a[] = \{5,2,4,6,1,3\}$ arrayinin öğelerini küçükten büyüğe doğru sıralayacağız. Array'in öğelerini 6 gözü olan bir kutuya aşağıdaki veriliş sırasıyla aşağıdaki gibi yerleştirelim. Sıralama işlemi boyunca kutunun 6 gözü sabit kalacaktır. Biz sadece gözdeki bir sayıyı alıp elimizde tutatbiliriz veya başka bir göze taşıyabiliriz. Ancak, bir sayıyı başka bir göze taşıyınca, orada var olan sayı silinecek, onun yerine taşınan sayı görünür olacaktır.

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

- Array'in en küçük indisli öğesi $a[0] = 5$ dir. İlk önce bunu seçelim. Tek öğeli $\{a[0]\} = \{5\}$ kümesi (altdizim) sıralıdır. Array'in geri kalan $\{2,4,6,1,3\}$ altdizimi ise sırasızdır.

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

- İkinci adımda sırasız altdizimin ilk öğesi olan 2 yi seçip, sıralı altdizimde olması gereken yere taşıyacağız. Buun için 2 sayısını sol elimize alalım. Sağ elimizle 5 sayısını ikinci göze taşıyalım. Sonra sol elimizdeki 2 sayısını birinci göze koyalım. Bu işlem sonunda sıralı altdizim $\{2,5\}$ olur, sırasız altdizim ise $\{4,6,1,3\}$ olacaktır.

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

3. Üçüncü adımda, sırasız altdizimin ilk ögesi olan 4 sayısını sıralı altdizimde olması gereken yere taşıyacağız. $2 < 4 < 5$ olduğundan, 4 sayısını 2 ile 5 arasına yerleştirmeliyiz (insertion). Bunun için 2 ile 5 in arasını açmalıyız. 4 sayısını sol elimizde tutarken, sağ elimizle 5 sayısını üçüncü göze taşıyalım. Sonra sol elimizdeki 4 sayısını ikinci göze koyalım. Bu işlem sonunda sıralı altdizim {2,4,5}, sırasız altdizim ise {6,1,3} olacaktır.

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

4. Dördüncü adımda, sırasız altdizimin ilk ögesi olan 6 sayısını sıralı altdizimde olması gereken yere taşıyacağız. $5 < 6$ olduğundan, 6 sayısını sıralı dizimin sonuna yerleştirmemiz gerekiyor. O halde sıralı dizimin öğeleri yerlerini koruyacaktır. Bu işlem sonunda sıralı altdizim {2,4,5,6}, sırasız altdizim ise {1,3} olacaktır.

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

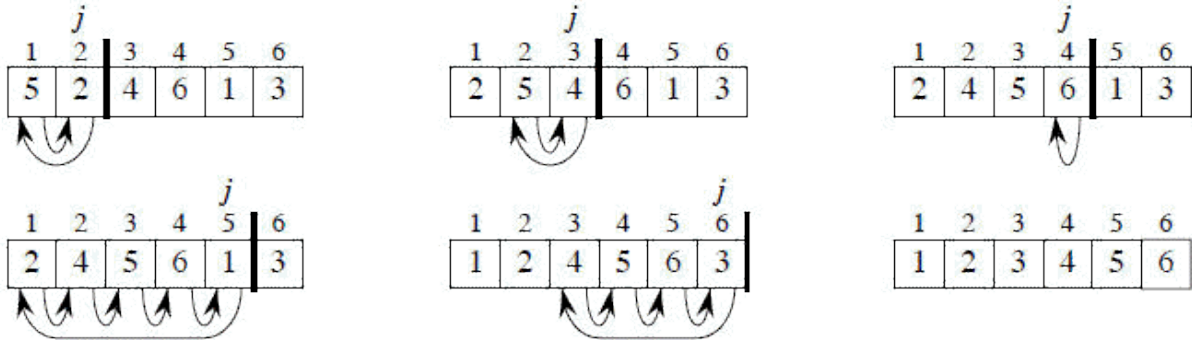
5. Beşinci adımda, sırasız altdizimin ilk ögesi olan 1 sayısını sıralı altdizimde olması gereken yere taşıyacağız. $6 > 1$, $5 > 1$, $4 > 1$ ve $2 > 1$ olduğundan, 1 sayısını sıralı dizimde 2 sayısının önüne kadar götürmeliyiz. 1 sayısını sol elimizde tutarken, sağ elimizle sırasıyla şu işlemleri yapalım: 6 sayısını beşinci göze, 5 sayısını dördüncü göze, 4 sayısını üçüncü göze, 2 sayısını ikinci göze taşıyalım. En sonunda sol elimizdeki 1 sayısını ilk göze koyalım. Bu işlemler bitince, sıralı altdizim {1,2,4,5,6}, sırasız altdizim ise tek ögeli {3} kümesi olacaktır.

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

1. Altıncı adımda, sırasız altdizimin tek ögesi olan 3 sayısını sıralı altdizimde olması gereken yere taşıyacağız. $1 < 2 < \mathbf{3} < 4 < 5 < 6$ olduğundan, 3 sayısını sıralı dizimde 2 ile 4 arasına yerleştirmeliyiz. 3 sayısını sol elimizde tutarken, sağ elimizle sırasıyla şu işlemleri yapalım: 6 sayısını altıncı göze, 5 sayısını beşinci göze, 4 sayısını dördüncü göze taşıyalım. Sonra sol elimizdeki 3 sayısını üçüncü göze koyalım. Bu işlemler bitince, arrayin bütün öğeleri $\{1,2,4,5,6\}$ biçiminde sıralanmış olur.

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

Yukarıda yapılan işlemleri, aşağıdaki şekillerdeki oklarla da izleyebiliriz.



Insertion Sort

Bu sıralamayı yapan bir java metodu yazabiliriz:

```
void insertionSort(int[] arr) {
    int i, j, Taşınan;
    for (i = 1; i < arr.length; i++) {
        Taşınan = arr[i];
        j = i;
        while (j > 0 && arr[j - 1] > Taşınan) {
            arr[j] = arr[j - 1];
            j--;
        }
        arr[j] = Taşınan;
    }
}
```

Bu fonksiyonu çağıran bir java programı yazalım:

```
package Sıralama;

public class InsertionSort {
    int[] a = {5,2,4,6,1,3};

    void insertionSıralama(int[] arr) {
        int i, j, Taşınan;
        for (i = 1; i < arr.length; i++) {
            Taşınan = arr[i];
            j = i;
            while (j > 0 && arr[j - 1] > Taşınan) {
                arr[j] = arr[j - 1];
                j--;
            }
            arr[j] = Taşınan;
        }
    }

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

    public static void main(String[] args) {
        InsertionSort insrt = new InsertionSort();

        insrt.insertionSıralama(insrt.a);

        insrt.arrayYaz(insrt.a);
    }
}
```

```

public class InsertionSort{
    public static void main(String a[]){
        int i;
        int array[] = {12,9,4,99,120,1,3,10};
        System.out.println("\n\n        SIRALAMA\n\n");
        System.out.println("        Selection Sort\n\n");
        System.out.println("Sıralamadan önceki durum:\n");
        for(i = 0; i < array.length; i++)
            System.out.print( array[i]+" ");
        System.out.println();
        insertion_srt(array, array.length);
        System.out.print("Sıralamadan sonraki durum:\n");
        for(i = 0; i < array.length; i++)
            System.out.print(array[i]+" ");
        System.out.println();
        System.out.println("PAUSE");
    }

    public static void insertion_srt(int array[], int n){
        for (int i = 1; i < n; i++){
            int j = i;
            int B = array[i];
            while ((j > 0) && (array[j-1] > B)){
                array[j] = array[j-1];
                j--;
            }
            array[j] = B;
        }
    }
}

```

```
package Siralama;

import java.util.Arrays;

public class InsertionSort02 {

    int[] arr = { 3, 17, 86, -9, 7, -11, 38 };

    static void insertionSort(Comparable[] a, int left, int right) {
        // Sort a [left...right].
        for (int r = left + 1; r <= right; r++) {
            Comparable val = a[r];
            int p = r;
            while (p > left && val.compareTo(a[p - 1]) < 0) {
                a[p] = a[p - 1];
                p--;
            }
            a[p] = val;
        }
    }

    public static void main(String[] args) {
        InsertionSort02 iSort = new InsertionSort02();
        System.out.println("\nSıralamadan önce:");
        System.out.println(Arrays.toString(iSort.arr));
        insertionSort(iSort.arr, 0, 6);
        System.out.println("\n\nSıralamadan sonra:");
        System.out.println(Arrays.toString(iSort.arr));
    }
}
```

```
import java.util.*;
import java.util.Arrays;

public class InsertionSort03 {

    int[] arr = { 3, 17, 86, -9, 7, -11, 38 };

    static void insertionSort(int[] a, int left, int right) {
        // Sort a [left...right].
        for (int r = left + 1; r <= right; r++) {
            int val = a[r];
            int p = r;
            while (p > left && val < a[p - 1] ) {
                a[p] = a[p - 1];
                p--;
            }
            a[p] = val;
        }
    }

    public static void main(String[] args) {
        InsertionSort03 iSort = new InsertionSort03();
        System.out.println("\nSıralamadan önce:");
        System.out.println(Arrays.toString(iSort.arr));
        iSort.insertionSort(iSort.arr, 0, 6);
        System.out.println("\n\nSıralamadan sonra:");
        System.out.println(Arrays.toString(iSort.arr));
    }
}
```