

# Sort\_Insertion

🕒 작성일시	@2022년 8월 3일 오후 5:42
▼ 강의 번호	
▼ 유형	
▼ 강사	
🔗 자료	
☑ 복습	<input type="checkbox"/>
📅 날짜	@2022년 7월 28일



## 단순 삽입 정렬 Insertion Sort

기본 삽입법 , 삽입법, 삽입 정렬

요소들을 하나씩 차례대로 올바른 순서가 되도록 데이터를 삽입하여 최종적으로 전체 데이터를 오름차순 또는 내림차순정렬을 하는 알고리즘이다.

우리가 실제생활에서 물건을 정렬할때 사용 하는 방법에 가깝다.

기본적인 속도는 빠르지 않지만 데이터가 어느정도 정렬 이 되어 있는 상태라면 빠른 처리를 기대할수도 있는 정렬 알고리즘이다.

### ▼ 단순 이해법

## Insertion Sort

5	3	4	1	2
0	1	2	3	4

5	3	4	1	2
0	1	2	3	4

첫번째 칸의 공을 정렬된 공이라고 가정하고 나머지 공들은 정렬되지 않은 공으로 나누어 생각한다.

5	3	4	1	2
0	1	2	3	4

정렬이 완료된 (가정) 다음공의 위치를 정한다.

3	5	4	1	2
0	1	2	3	4

결과적으로 2개의 공의 정렬이 완료(가정) 되었다.

3	5	4	1	2
0	1	2	3	4

정렬이 되지 않은 첫번째 (4번공)를 올바른 위치로 삽입

3	4	5	1	2
0	1	2	3	4

결과적으로 3개의 공의 정렬이 완료(가정) 되었다.

3	4	5	1	2
0	1	2	3	4

정렬이 되지 않은 첫번째 (1번공)를 올바른 위치로 삽입

1	3	4	5	2
0	1	2	3	4

결과적으로 4개의 공의 정렬이 완료(가정) 되었다.

1	3	4	5	2
0	1	2	3	4

정렬이 되지 않은 첫번째 (2번공)를 올바른 위치로 삽입

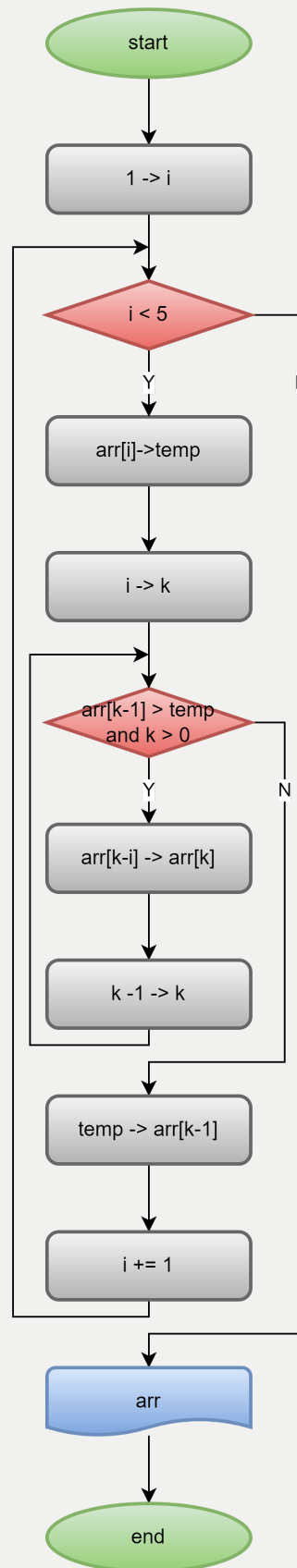
1	2	3	4	5
0	1	2	3	4

결과적으로 5개의 모든 공 정렬이 완료되었다.

### ▼ Flow Chart 알고리즘

1. 삽입하고 싶은 데이터는 정렬되었다고 가정한 데이터들과 순서대로 비교한다.

2. 변수의 데이터를 이미 정렬 되었다라고 가정한 데이터들과 순서대로 비교한다.
3. 변수의 데이터보다 작은 데이터가 발견되면 바로 그 뒤의 요소에 변수의 데이터를 입력한다.



## ▼ Java Code 자바코드

```
package algo;

public class InsertionSort {

    public static void main(String[] args) {
        int[] arr = {5,3,4,1,2};

        for(int i = 1 ; i < 5 ; i++) {
            int k = i;
            int temp = arr[i];
            while (k-1 >= 0 && temp < arr[k-1]) {
                arr[k] = arr[k-1];
                k--;
            }
            arr[k] = temp;
        }

        for (int each : arr) {
            System.out.print(each + " ");
        }

    }

}
```