


# Sorting

# Index	2
📅 CreatedAt	@September 28, 2022
👤 Person	<span>A</span> Ally Hyeseong Kim
☰ Status	In Progress
☰ Tags	Java Python Sorting
📅 UpdatedAt	@September 28, 2022

## References

### 파이썬 알고리즘 인터뷰

2021 세종도서 학술부문 선정작. 현업과 실무에 유용한 주요 알고리즘 이론을 깊숙이 이해하고, 파이썬의 핵심 기능과 문법까지 상세하게 이해할 수 있는 취업용 코딩 테스트를 위한 완벽

 <https://www.aladin.co.kr/shop/wproduct.aspx?ItemId=245495826>



### References

- [1. Sorting Algorithm](#)
- [2. Bubble Sort Algorithm](#)
- [3. Merge Sort Algorithm](#)
- [4. Quick Sort Algorithm](#)
- [5. Stable Sort vs Unstable Sort](#)

## 1. Sorting Algorithm

**Sorting Algorithm**은 요소를 특정 순서대로 넣는 알고리즘이다. 주로 *Numerical Order*, *Lexicographical Order*로 정렬한다.

## 2. Bubble Sort Algorithm

- 시간 복잡도:  $O(n^2)$

```
Bubblesort(A):
    for i from 1 to A.length
        for j from 0 to A.length - 1
            if A[j] > A[j + 1]
                swap a[j] with a[j + 1]
```

```
def bubblesort(A):
    for i in range(len(A)):
        for j in range(0, len(A) - 1):
            if A[j] > A[j + 1]:
                A[j], A[j + 1] = A[j + 1], A[j]
```

### 3. Merge Sort Algorithm

**Merge Sort Algorithm**은 **Divide and Conquer**를 사용하여 정렬한다.

- 시간 복잡도:  $O(n \log n)$

#### 3.1. Divide and Conquer

### 4. Quick Sort Algorithm

**Quick Sort Algorithm**은 **Divide and Conquer**를 사용하여 정렬한다. Pivot을 기준으로 좌우를 나누어 정렬하므로 **Partition-Exchange Sort Algorithm**이라고도 한다.

- 시간 복잡도: Worst  $O(n^2)$ , Best  $O(n \log n)$

```
partition(A, lo, hi):
    pivot := A[hi]
    i := lo
    for j := lo to hi do
        if A[j] < pivot then
            swap A[i] with A[j]
            i := i + 1
    swap A[i] with A[hi]
    return i

Quicksort(A, lo, hi):
    if lo < hi then
        pivot := partition(A, lo, hi)
        Quicksort(A, lo, pivot - 1)
        Quicksort(A, pivot + 1, hi)
```

```
def Quicksort(A, lo, hi):
    def partition(lo, hi):
        pivot = A[hi]
        left = lo
        for right in range(lo, hi):
            if A[right] < pivot:
                A[left], A[right] = A[right], A[left]
                left += 1
        A[left], A[hi] = A[hi], A[left]
        return left
    if lo < hi:
        pivot = partition(lo, hi)
        Quicksort(A, lo, pivot - 1)
        Quicksort(A, pivot + 1, hi)
```

## 5. Stable Sort vs Unstable Sort

**Stable Sort Algorithm**은 기존 입력값을 유지하며 정렬한다.

- Merge Sort, Bulbble Sort Algorithm

**Unstable Sort Algorithm**은 기존 입력값을 유지하지 않고 정렬한다.

- Quick Sort Algorithm