



데이터 구조

12주차(continue): 정렬

힙 정렬 (Heap Sort)

● Heap 정의

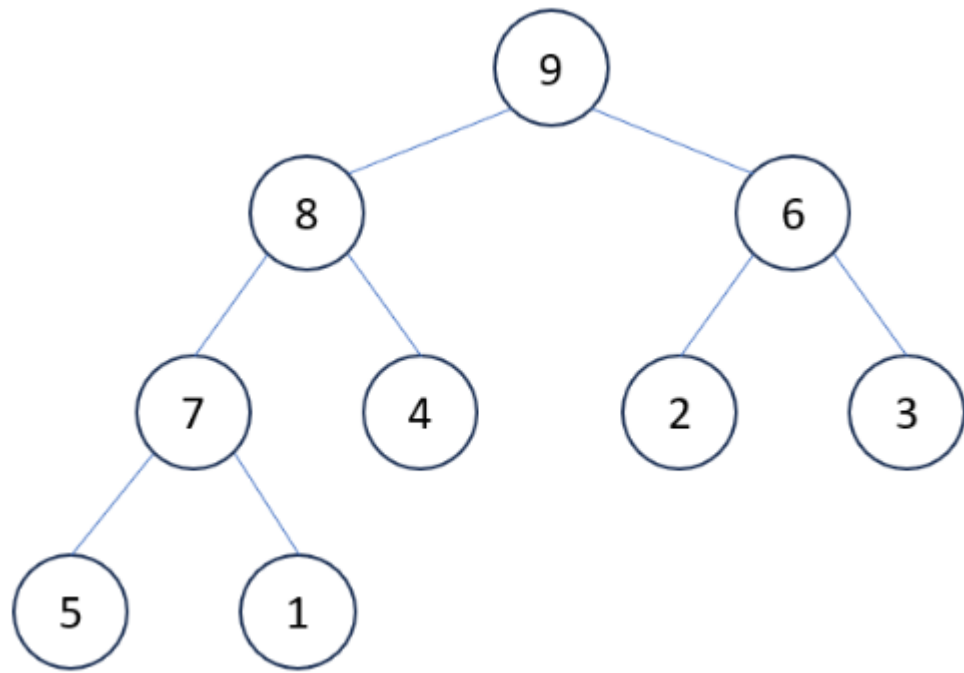
- 완전이진트리 일종으로, 부모노드와 자식 노드간에 특정 조건을 만족하는 자료구조
- Heap 에는 최대 Heap과 최소 Heap이 있으며,
- Heap 정렬을 하기 위해서는 최대 Heap 또는 최소 Heap이 되도록 구성

● Heap 종류

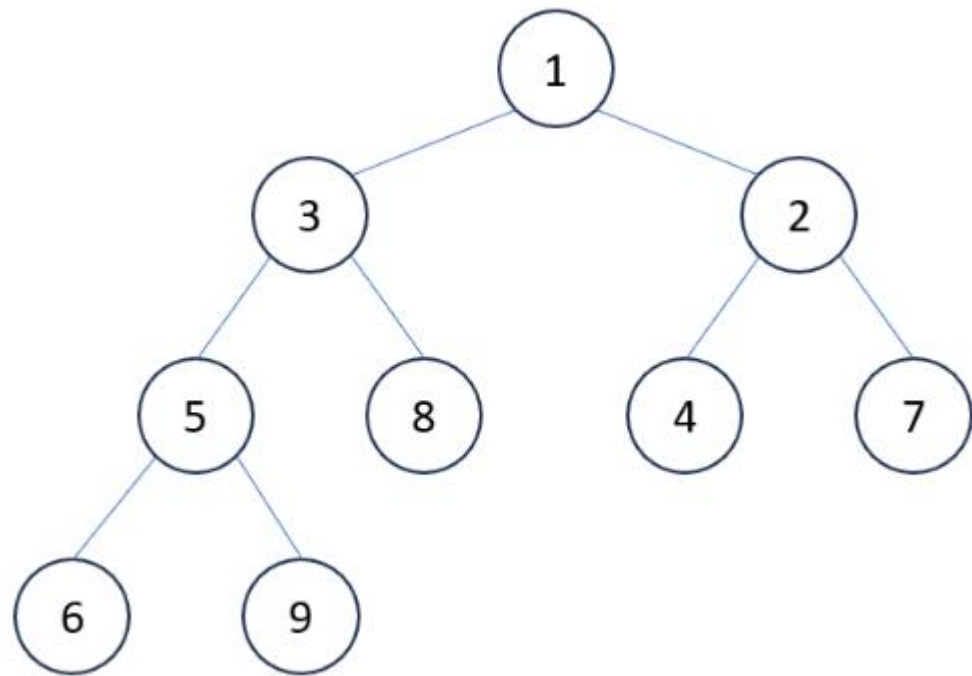
- 최대 Heap
 - 모든 부모 노드가 그 자식 노드보다 큰 값을 갖는 특성
- 최소 Heap
 - 모든 부모 노드가 그 자식 노드보다 작은 값을 갖는 특성

힙 정렬 (Heap Sort)

최대 Heap 예시



최소 Heap 예시



힙 정렬 (Heap Sort)

● Heap 구현 방법

- 힙은 보통 배열을 이용해서 구현

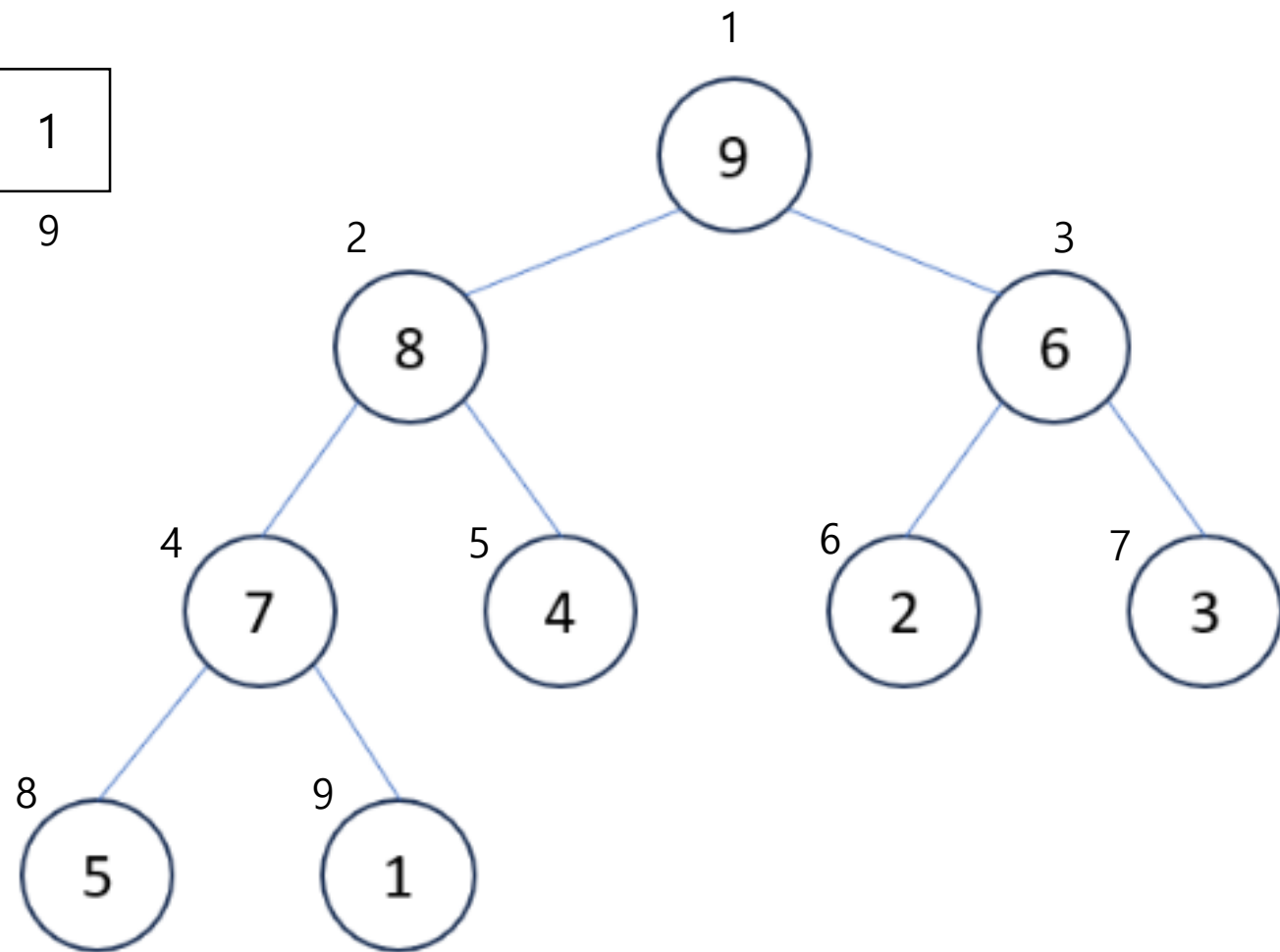
※ 배열로 구현된 힙 자료구조에서 자식 노드 구분을 위한 공식

- 왼쪽 자식 노드: $2 * \text{index}$
- 오른쪽 자식 노드: $2 * \text{index} + 1$
- 부모 노드 = $\text{index} // 2$

힙 정렬 (Heap Sort)

● Heap 자료구조에 대한 노드와 배열 자료형 표현 상관관계

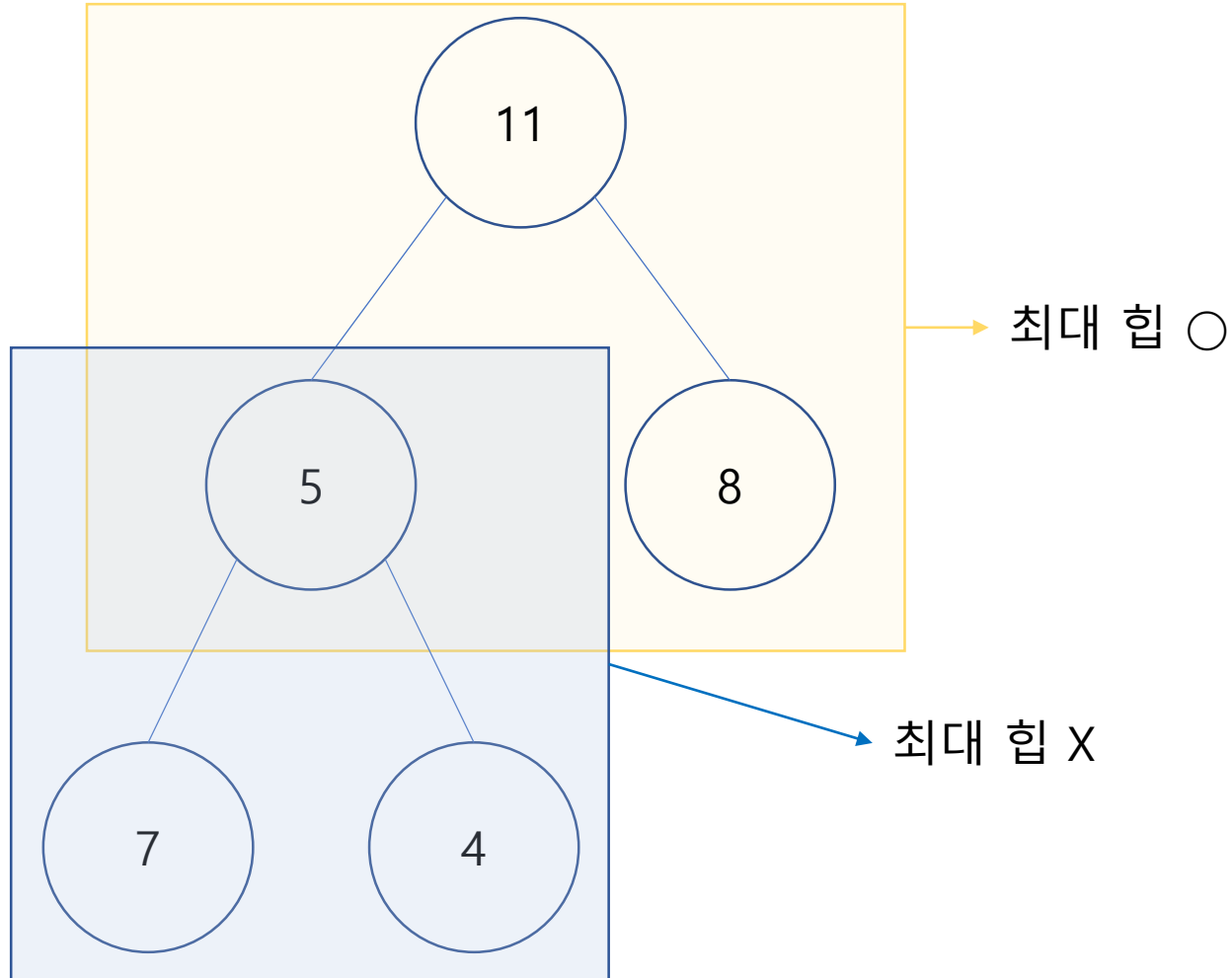
	9	8	6	7	4	2	3	5	1
0	1	2	3	4	5	6	7	8	9



힙 정렬 (Heap Sort)

● 데이터를 Heap 구성으로 재배열하는 방법 (최대 힙 기준으로 설명)

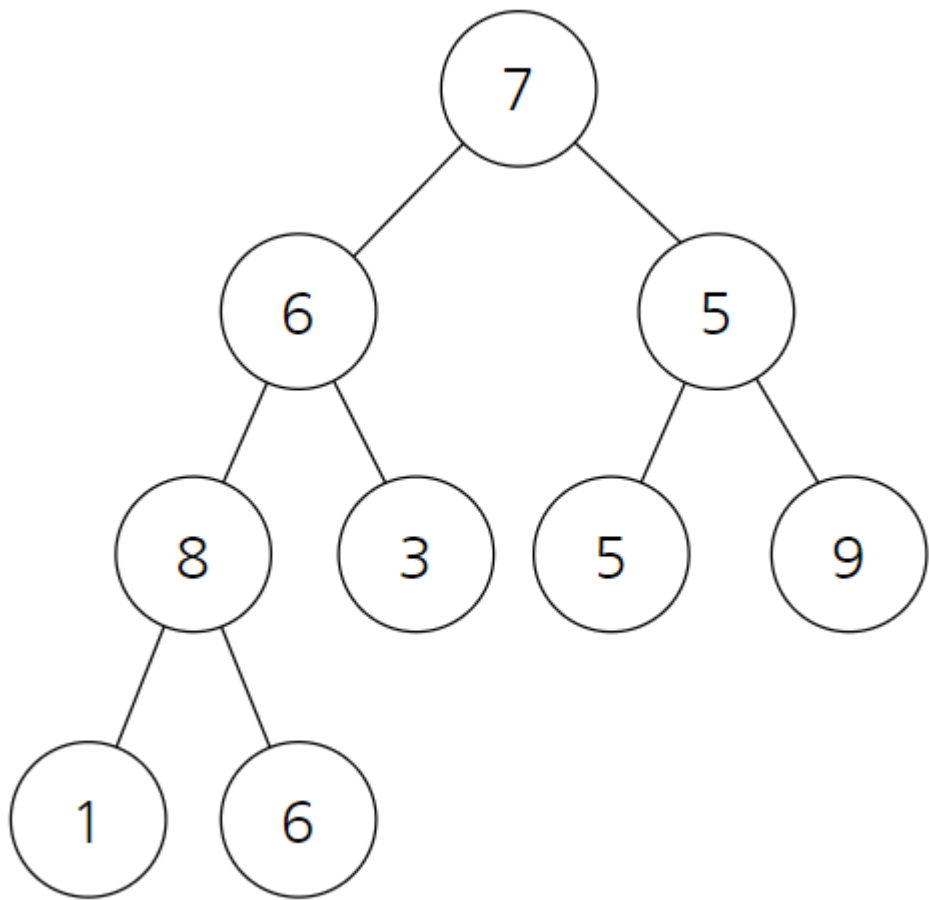
- 힙 생성 알고리즘 (Heapify Algorithm)



- 힙 정렬을 수행하기 위해서는 힙 생성알고리즘을 사용
- 힙 생성 알고리즘은 특정한 하나의 노드에 대해 수행
- 특정한 노드의 두 자식 노드 중에서 더 큰 자식과 자신의 위치를 교환하는 알고리즘
- 위치를 바꾼 뒤에도 여전히 자식이 존재하는 경우 반복해서 수행하며, 자식 노드가 더 이상 존재하지 않을 때 까지 반복

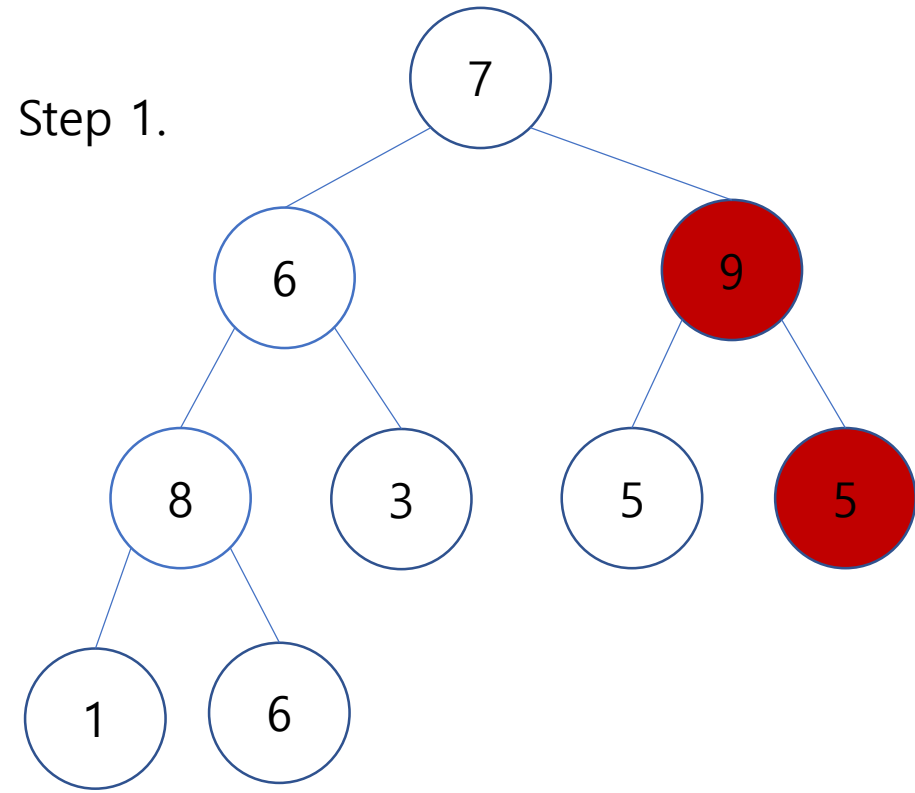
힙 정렬 (Heap Sort) - 예시

7	6	5	8	3	5	9	1	6
0	1	2	3	4	5	6	7	8



1. 주어진 데이터를 Heap 구조로 재배열

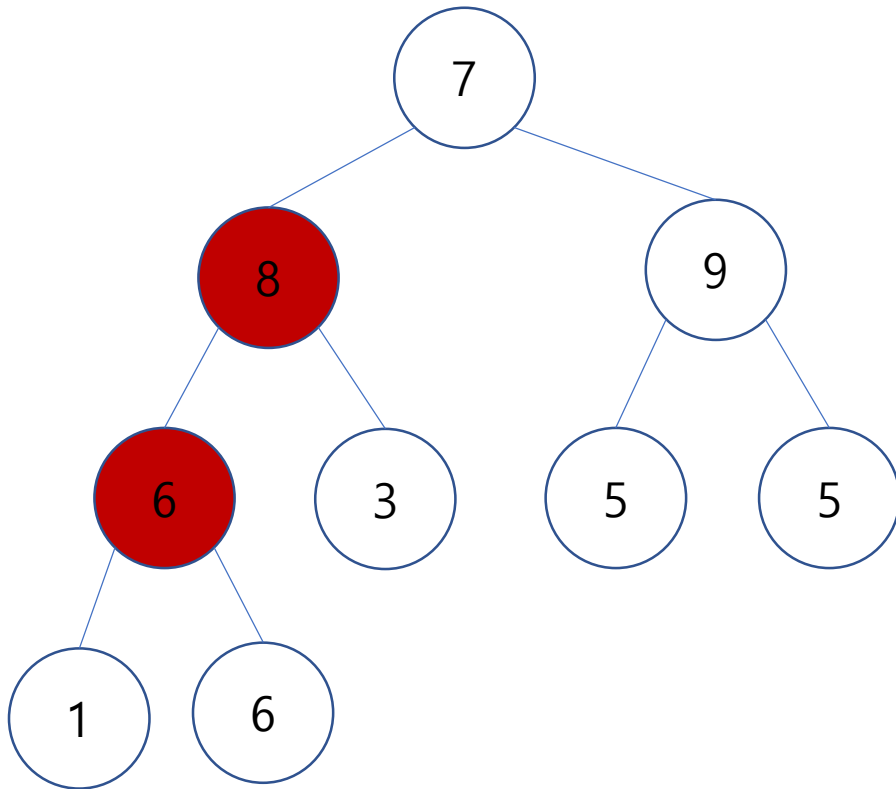
7	6	9	8	3	5	9	1	6
---	---	---	---	---	---	---	---	---



힉 정렬 (Heap Sort) - 예시

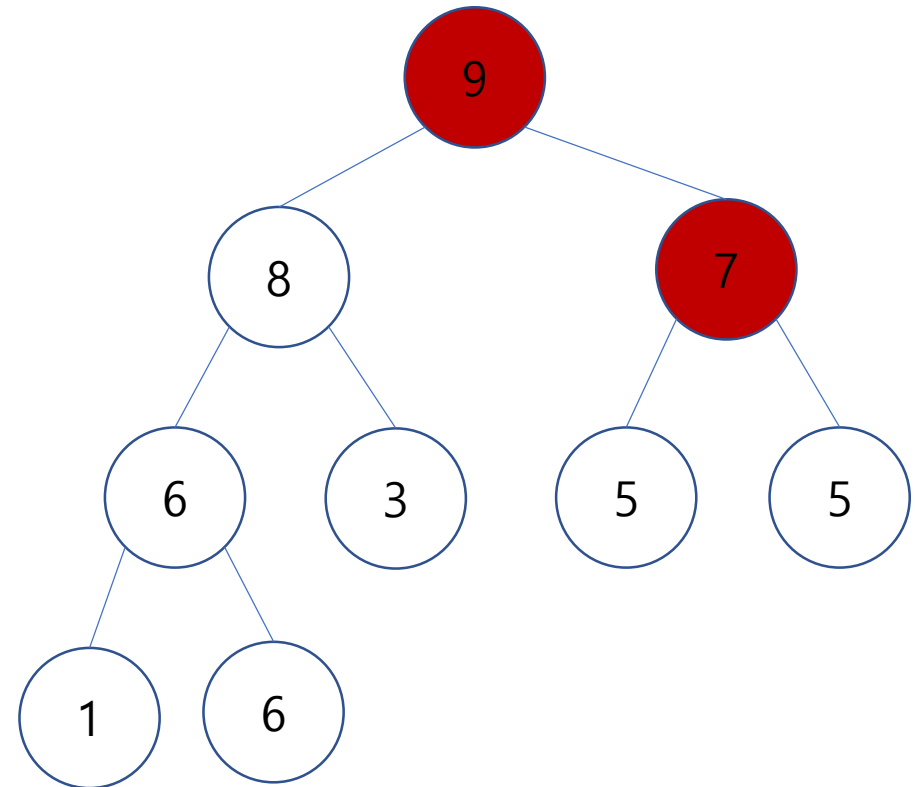
Step 2.

7	8	9	6	3	5	9	1	6
---	---	---	---	---	---	---	---	---



Step 3.

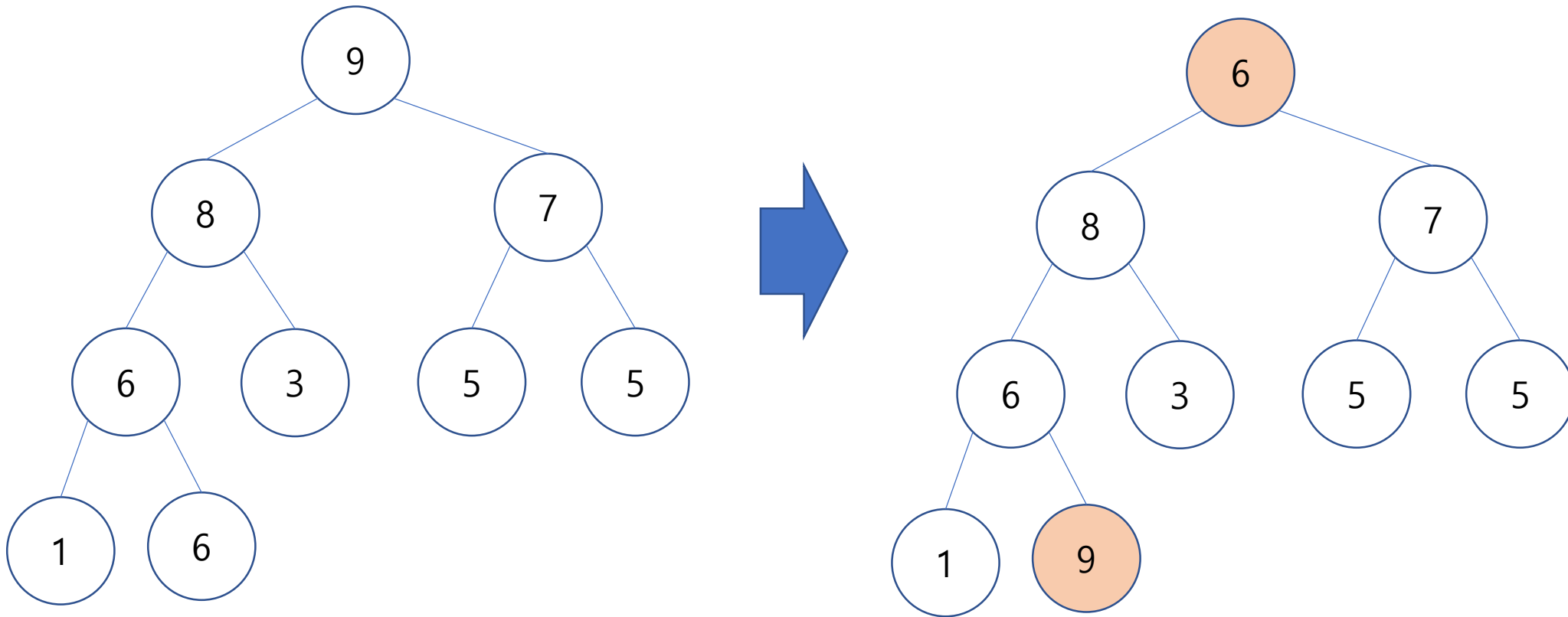
9	8	7	6	3	5	9	1	6
---	---	---	---	---	---	---	---	---



힙 정렬 (Heap Sort) - 예시

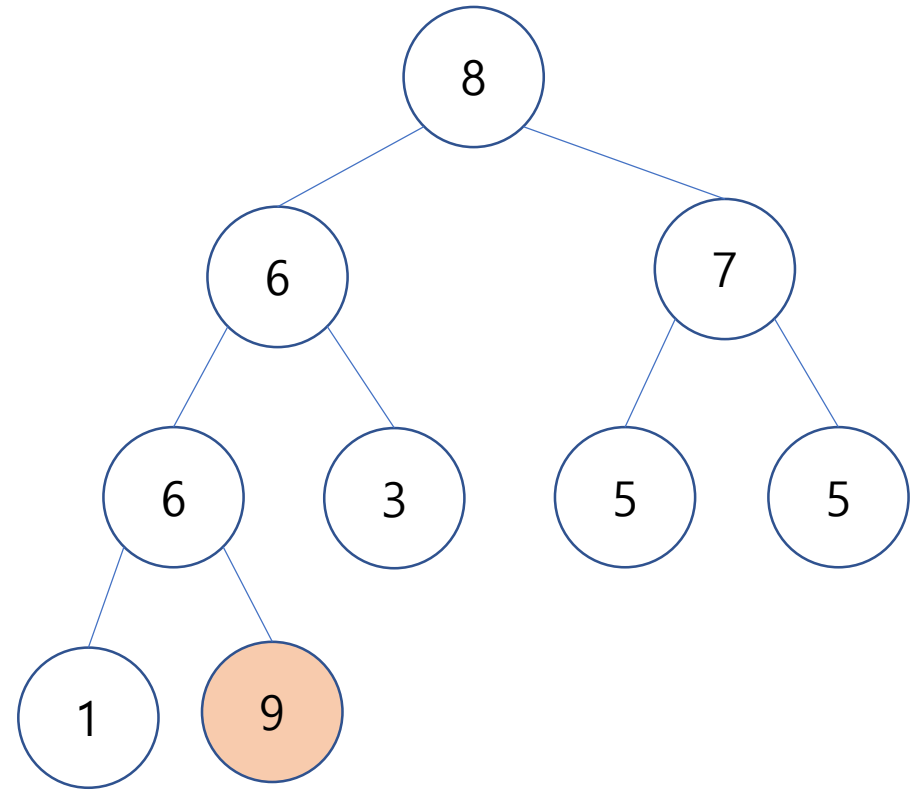
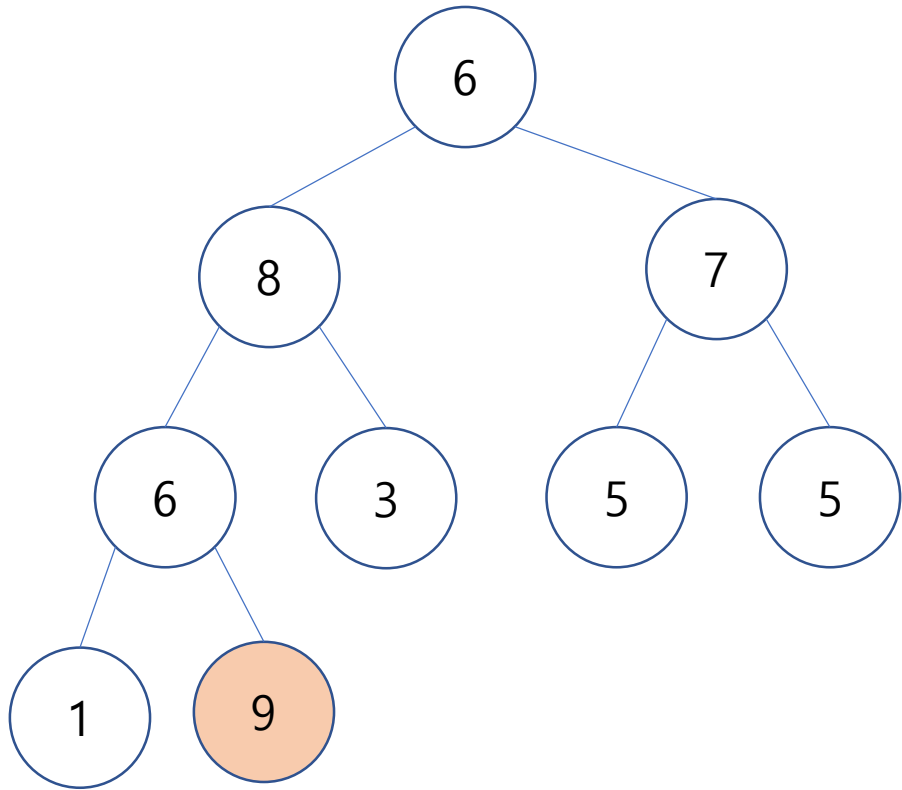
2. 루트(Root)에 있는 값을 가장 뒤쪽으로 보내면서 힙 트리의 크기를 1감소

- 제일 마지막으로 보내진 데이터("9")는 정렬 완료된 데이터이며, 마지막 원소("9")를 제외한 나머지 데이터로 다시 힙 생성 알고리즘 수행



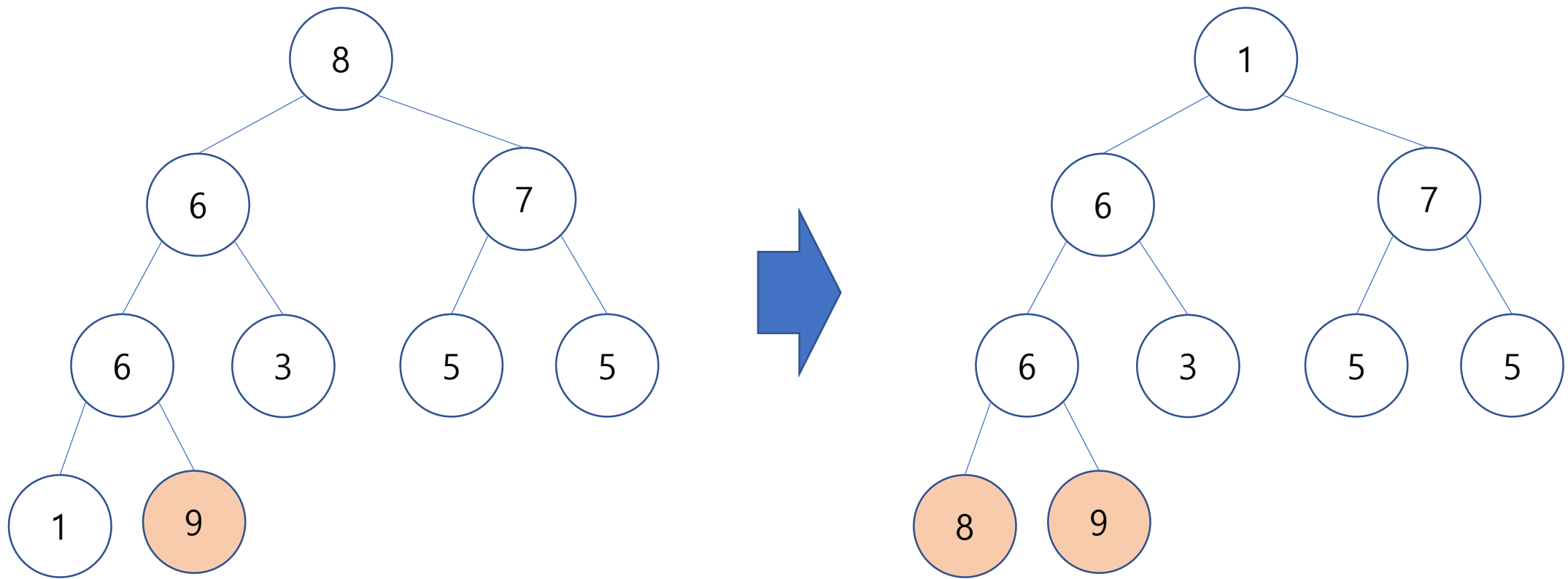
힙 정렬 (Heap Sort) - 예시

3. 제일 마지막에 있는 데이터("9")를 제외하고 다시 힙 생성 알고리즘 수행



힉 정렬 (Heap Sort) - 예시

4. 루트(Root)에 있는 값을 가장 뒤쪽으로 보내면서 힉 트리의 크기를 1감소
정렬 완료된 데이터를 제외한 나머지 데이터로 다시 힉 생성알고리즘 수행



합병 정렬 (Merge Sort)

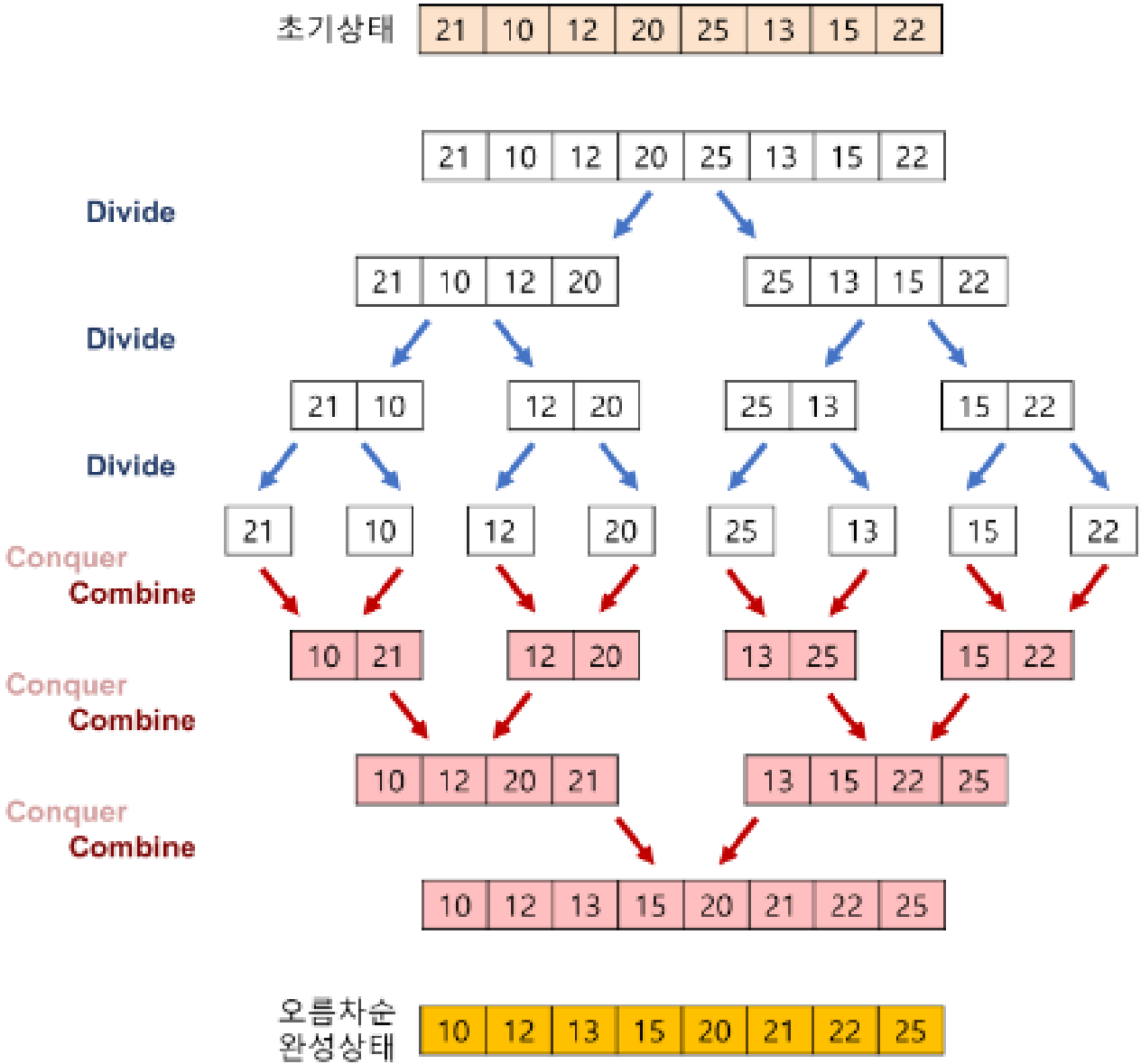
● Merge Sort 정의

- 병합 정렬은 분할정복(Divide and Conquer)기법과 재귀 알고리즘을 이용하는 정렬 알고리즘
- 주어진 배열을 원소가 하나 밖에 남지 않을 때까지 계속 둘로 쪼갠 후에 다시 크기 순으로 재배열 하면서 원래 크기의 배열로 합침

● Merge Sort 특징

- 분할(Split) 단계와 합병(Merge) 단계로 나누며, 분할 비용보다 모든 값들을 비교해야 하는 합병 비용이 큼

합병 정렬 (Merge Sort)



퀵 정렬 (Quick Sort)

● Quick Sort 정의

- 합병 정렬과 마찬가지로 분할기법과 재귀 알고리즘을 이용하는 정렬 알고리즘
- 피벗(pivot)이라는 임의의 기준값을 사용하여 분할
- Pivot 을 기준으로 더 작은 값과 큰 값으로 반복 분할한 후 합침

● Quick Sort 특징

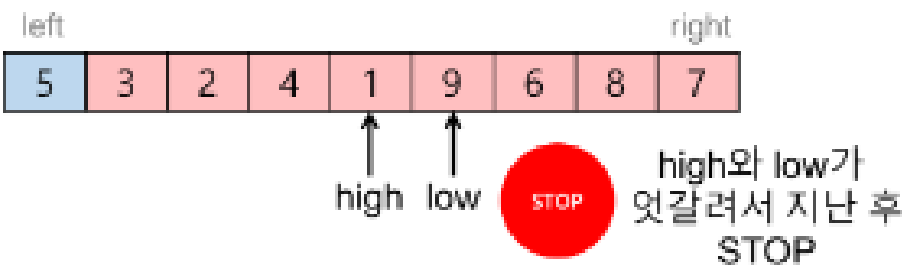
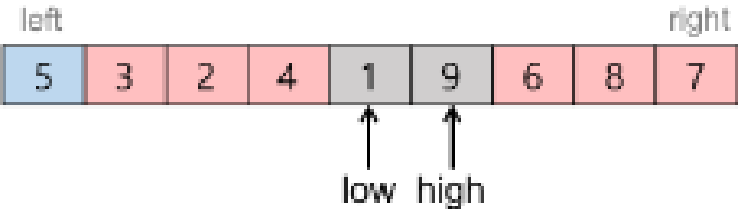
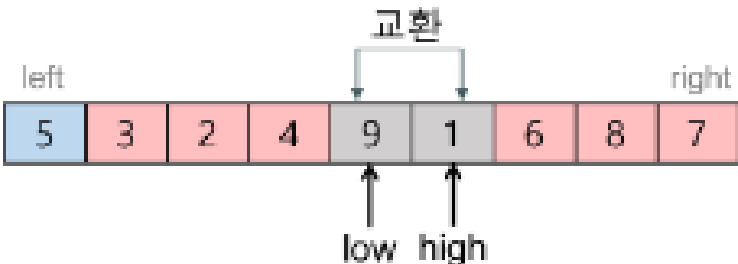
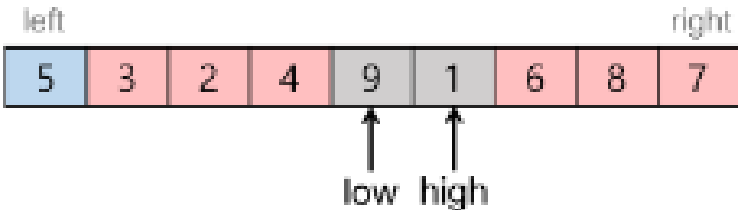
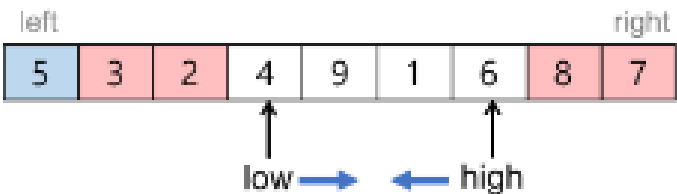
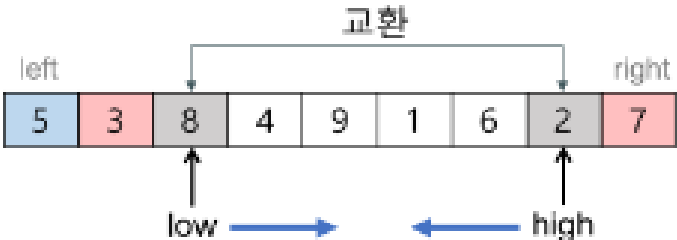
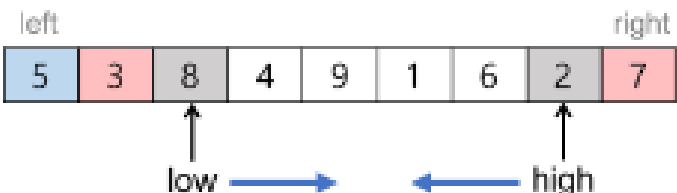
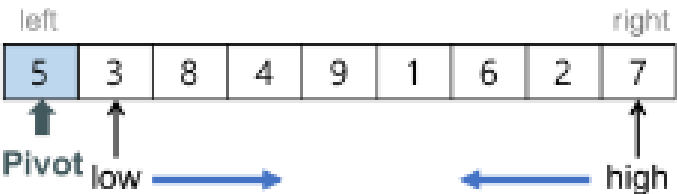
- 일반적으로 원소의 개수가 적어질수록 나쁜 중간값이 선택될 확률이 높아지기 때문에, 원소의 개수에 따라 퀵 정렬에 다른 정렬을 혼합해서 쓰는 경우 많음
- 합병 정렬은 항상 정 중앙을 기준으로 단순 분할 후 병합시점에서 값의 비교 연산이 발생하는 반면, 퀵 정렬은 분할시점부터 비교연산이 발생하기 때문에 그 이후 병합에 들어가는 비용이 매우 적음

퀵 정렬 (Quick Sort)

초기상태

5	3	8	4	9	1	6	2	7
---	---	---	---	---	---	---	---	---

1



퀵 정렬 (Quick Sort)

