

힉과 우선순위큐

2019.04.16

15 신준호





INDEX



001/

힙

002/

우선순위큐

003/

0x00

004/

null

005/

'\0'



알고가자 알고리즘

힙(Heap)

- 잘 알아보도록 합시다

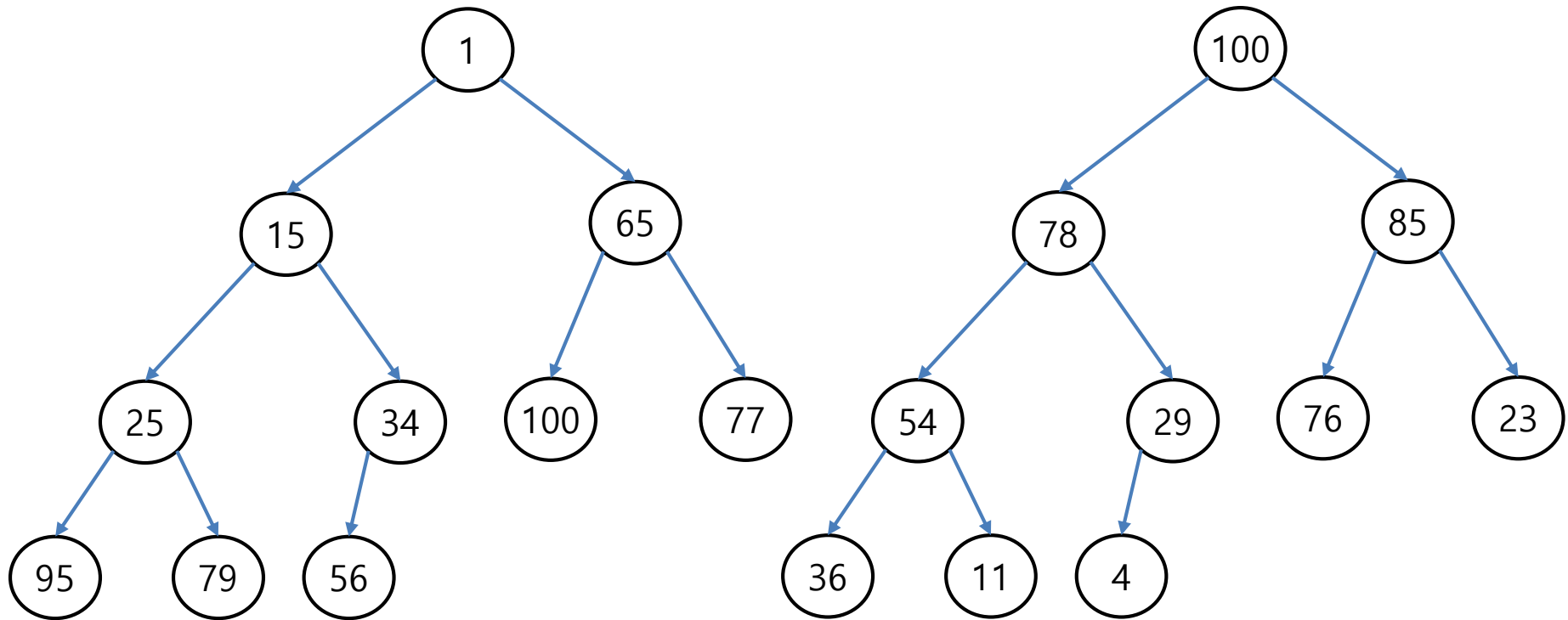




힙

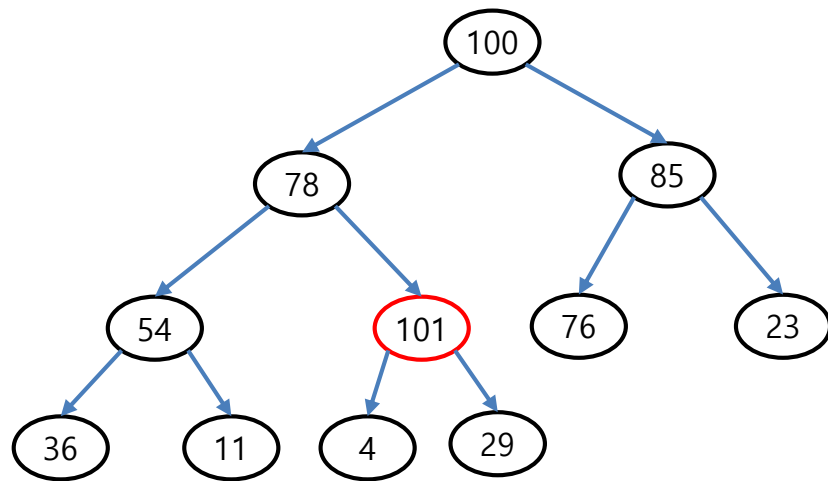
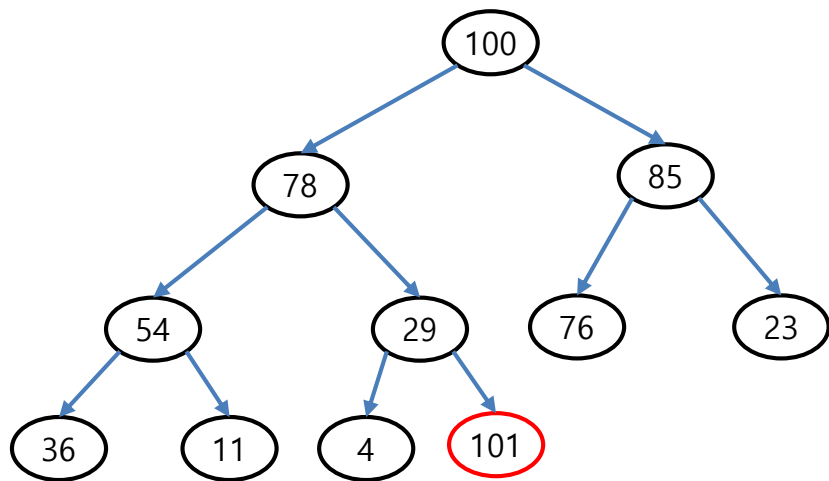
- 완전이진트리를 기본으로 한 자료구조
- 최소힙 : 부모노드.value \leq 자식노드.value
- 최대힙 : 부모노드.value \geq 자식노드.value
- 최댓값 및 최솟값을 찾아내는 연산을 빠르게 하기 위해 고안

최소 힙과 최대 힙



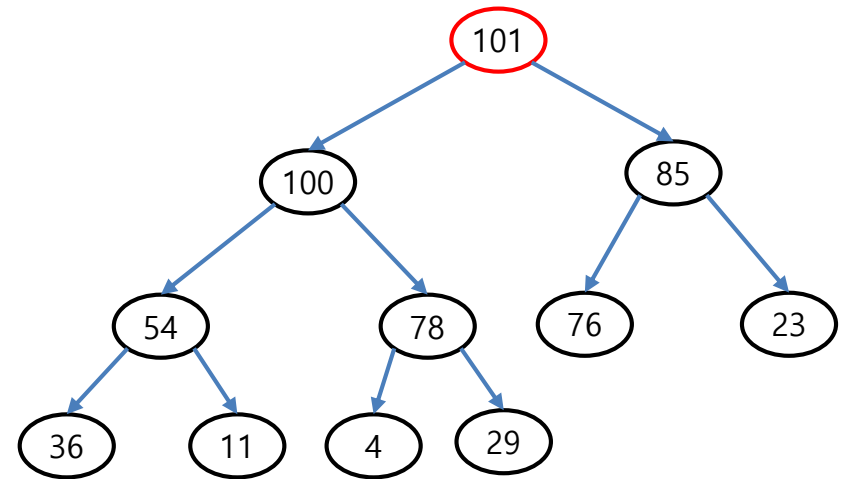
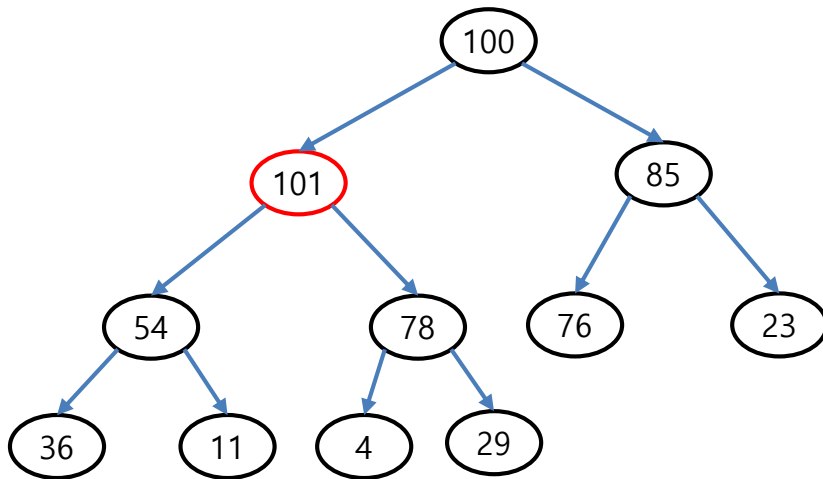
최대 힙의 삽입

1. 트리의 마지막 원소로 값을 삽입한다.
2. 부모가 나보다 작다면 두 값을 교환
3. root까지 가거나, 부모가 자식보다 크면 끝



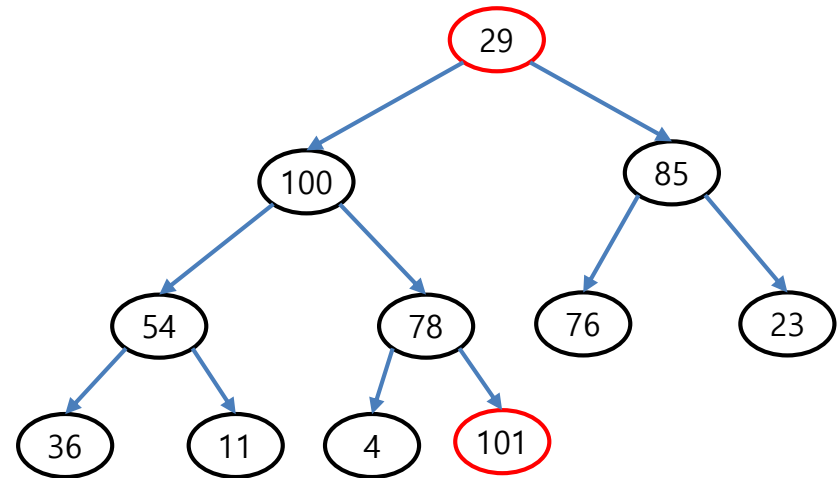
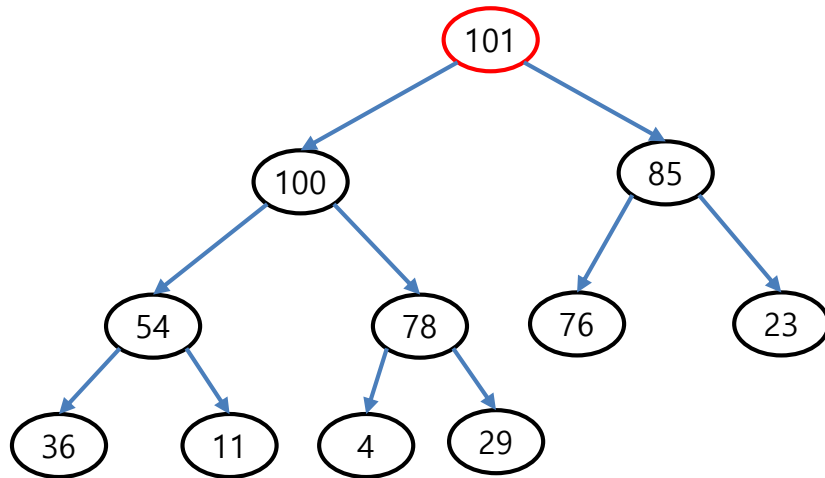
최대 힙의 삽입

1. 트리의 마지막 원소로 값을 삽입한다.
2. 부모가 나보다 작다면 두 값을 교환
3. root까지 가거나, 부모가 자식보다 크면 끝



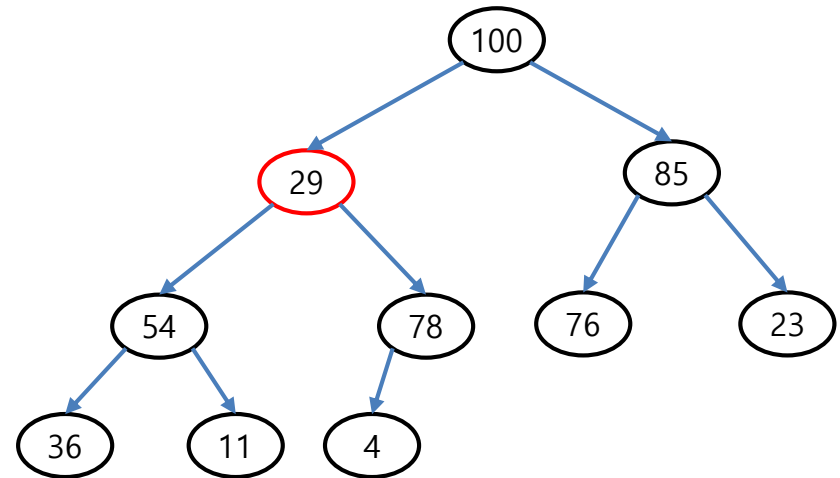
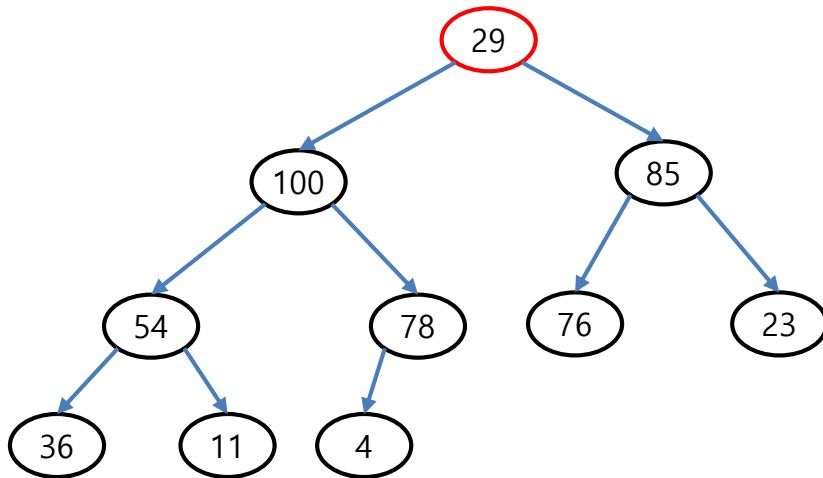
최대 힙의 삭제

1. root노드를 heap의 마지막 노드와 교환
2. 교환된 마지막 노드를 삭제
3. 새로운 root노드를 적합한 위치로 내려보낸다.
 - 두 자식 모두 자신보다 크면 더 큰 값의 자식을 선택해서 교환



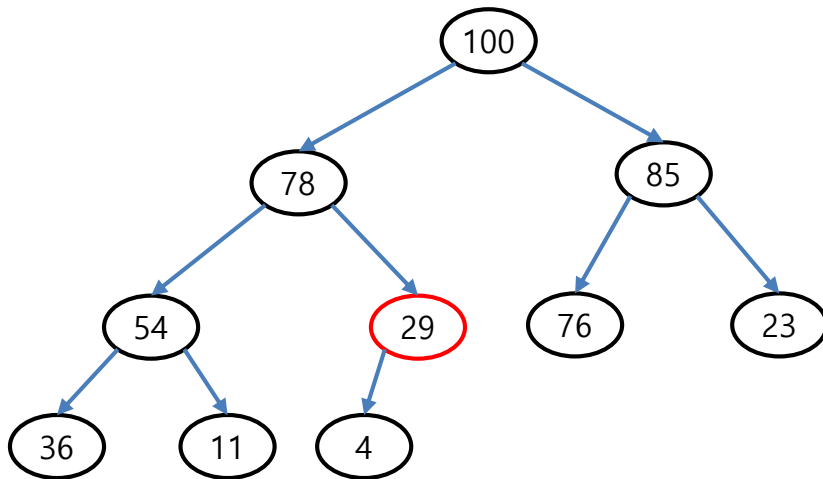
최대 힙의 삭제

1. root노드를 heap의 마지막 노드와 교환
2. 교환된 마지막 노드를 삭제
3. 새로운 root노드를 적합한 위치로 내려보낸다.
 - 두 자식 모두 자신보다 크면 더 큰 값의 자식을 선택해서 교환



최대 힙의 삭제

1. root노드를 heap의 마지막 노드와 교환
2. 교환된 마지막 노드를 삭제
3. 새로운 root노드를 적합한 위치로 내려보낸다.
 - 두 자식 모두 자신보다 크면 더 큰 값의 자식을 선택해서 교환



알고가자 알고리즘

우선순위 큐

- 잘 알아보도록 합시다



우선순위 큐

- 큐 : 들어간 데이터가 먼저 나오는 자료구조
- 우선순위 큐 : 들어간 순서에 상관없이 우선순위가 높은 데이터가 먼저 나옴





1. 배열을 기반으로 구현

- 데이터 삽입 및 삭제과정에서 데이터를 한 칸씩 당기거나 밀어야 하는 연산이 계속됨
- 삽입의 위치를 찾기 위해 배열에 저장된 모든 데이터와 우선순위를 비교

2. 연결리스트를 기반으로 구현

- 삽입의 위치를 찾기 위해 첫번째 노드에서부터 시작해 마지막 노드에 저장된 데이터와 우선순위를 비교할 지도 모른다.

3. 힙을 이용해 구현

- 일반적으로 사용하는 방식
- 삽입과 삭제에 있어서 일정한 $\log n$ 시간이 걸림

알고가자 알고리즘

우선순위 큐



간단하게 시간복잡도로 확인하자

우선순위큐 표현방식	삽입	삭제
순서 없는 배열	$O(1)$	$O(n)$
순서 없는 연결 리스트	$O(1)$	$O(n)$
정렬된 배열	$O(n)$	$O(1)$
정렬된 연결 리스트	$O(n)$	$O(1)$
힙	$O(\log n)$	$O(\log n)$

Q & A

Thank You for Listening

