

Heap

practice 자료구조/Heap · 2024. 9. 20. 18:45

Max-Heap (최대 힙): 부모 노드가 자식 노드보다 크거나 같은 값을 가지는 힙입니다.

루트가 최대값이라 빠르게 최대값 찾을수 있음

Min-Heap (최소 힙): 부모 노드가 자식 노드보다 작거나 같은 값을 가지는 힙입니다.

루트가 최소값이라 빠르게 최소값을 찾을수 있음

힙 정렬은 힙을 사용해 배열을 정렬하는 과정입니다. 배열을 max heap 또는 min heap으로 변환한 후, 힙의 루트에서 가장 큰 값(최소 힙의 경우 가장 작은 값)을 추출하는 과정을 반복하여 정렬합니다. 힙 정렬의 시간 복잡도는 $O(N \log N)$ 입니다.

Heap (Max heap)

1. 완전 이진트리 (Complete Binary Tree)

- 최하단의 leaf node가 좌측에 몰려있을경우 완전 이진 tree(왼쪽부터 채워져서)- 크기상관 없이 왼쪽부터 채워짐

2. 모든 부모 node는 자식 node 보다 크거나 같은 값을 가짐

cf) min heap도 있음(반대경우)

max-heap

=> 루트는 항상 가장 큰 값

루트를 빼낸 다음 재조정함

1. 정렬해야 하는 리스트를 max-heap으로 변환

2. 순서대로 가장 큰 원소부터 꺼냄

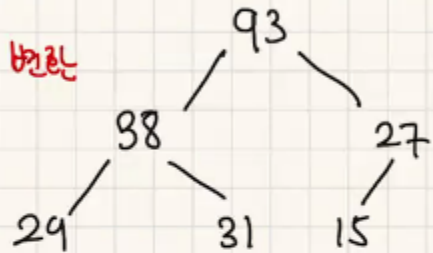
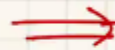
예시:

1.

max-heap으로 변환

정렬해야 할 리스트

$= [29, 38, 31, 93, 27, 15]$



2.

계속해서 Root 꺼내기

새로운 리스트에 넣고

max-heap 재조장 반복

비용: $O(\log n)$

총 n 번 반복 (주어진 예시에서는 $n=6$)

$\Rightarrow [93, 38, 31, 29, 27, 15]$ 정렬 끝

• 주어진 리스트의 원소의 갯수 n 일때

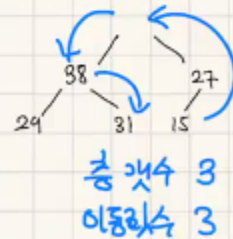
• max-heap 재조장을 n 번 반복

• 재조장은 $O(\log n)$ (총의 갯수 $\log_2 n$)

리스트의 원소의 갯수 \times 재조장 비용

$= n \times \log n$ (max-heap이서 리나 뺀다 재조장 하는 것 n 번 반복)

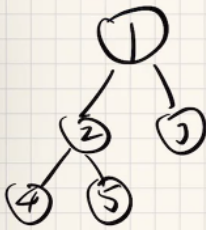
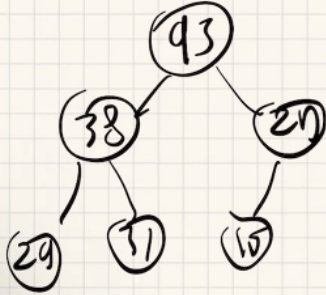
$\Rightarrow O(n \log n)$



Heap (Max heap)

1. 이진 트리

2. 모든 부모 Node는 자식 Node 보다
크거나 같은 값을 가짐

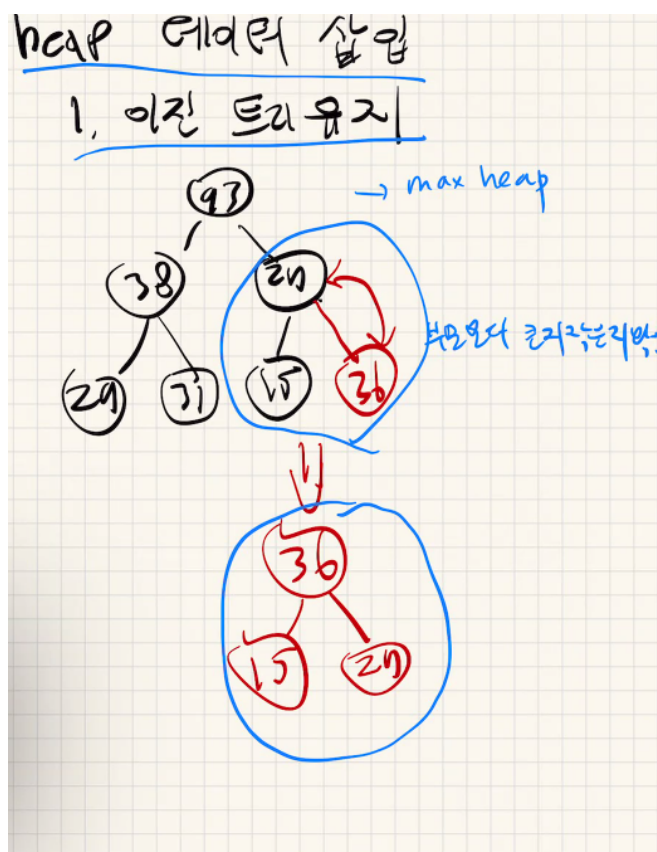


min heap

아래그림

Heap 데이터 삽입

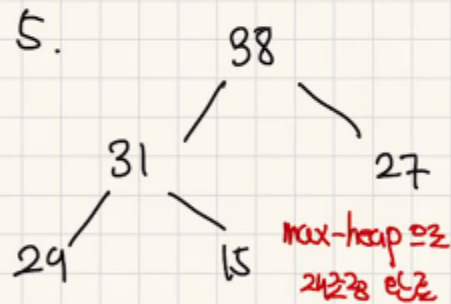
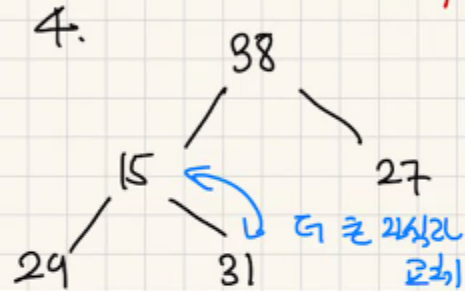
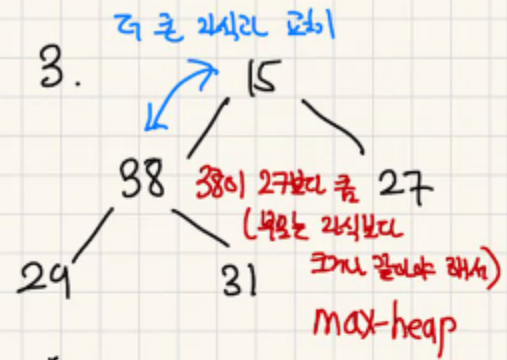
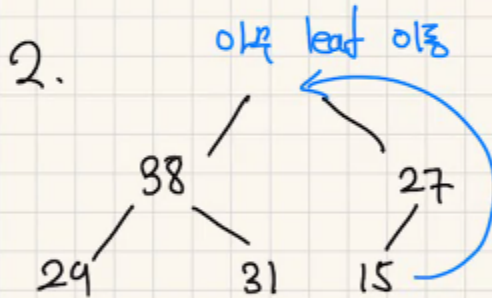
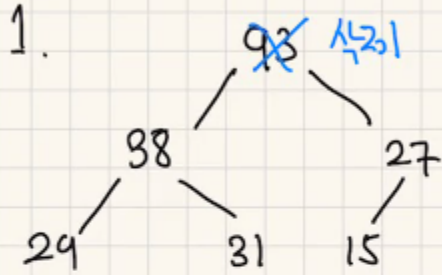
- 이진트리를 유지하면서 새로운데이터는 부모보다 큰지 작은지 확인후 자리바꿈



아래그림) max heap(가장큰값찾은거-root)

(max-heap 이니)

삭제: 아박 leaf 노드인 노드를
올바른 위치에 올리거나
더 큰 자식과 바꾼다.



이런 탐색 트리: 왼쪽에는 부모보다 작음, 오른쪽은 부모보다 큼

max-heap: 부모는 자식보다 크거나 같다

min-heap: 부모는 자식보다 작거나 같다.

왼쪽 자식이 부모보다 작고,
오른쪽이 부모보다 클 필요 X

♡ 공감 📌

구독하기

'practice' 자료구조 > Heap 카테고리의 다른 글

Heap_2. (0)

2024.09.21



자연어(NLP)

네이쳐2024 님의 블로그입니다.



구독하기 +

댓글 0



이름

비밀번호

내용을 입력하세요.



등록