



Pervasive Elastic MetaLearning Laboratory
Department of Computer Engineering
Hongik University

HW11 실습

Heaps

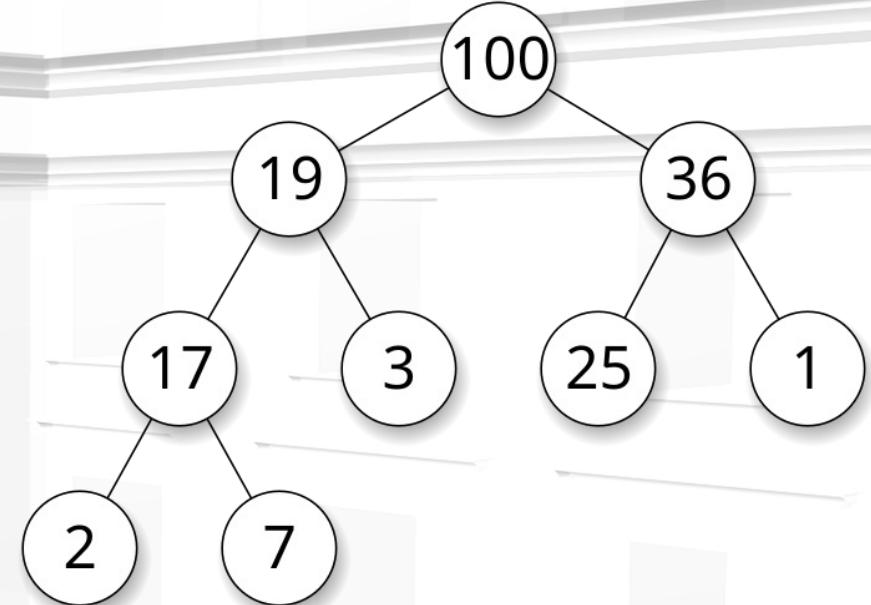
Max Heap

Definition

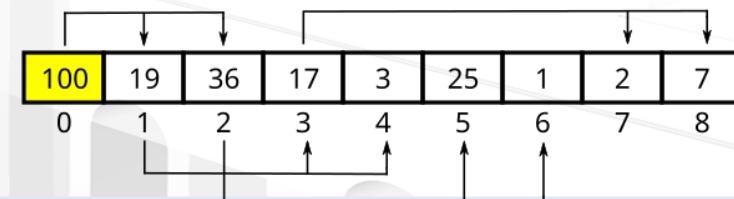
■ Complete Binary Tree && Max Tree

- Complete Binary tree:
 - A binary tree with n nodes and depth k is *complete* iff its nodes correspond to the nodes numbered from 1 to n in the full binary tree of depth k
- Max Tree:
 - A tree which the key value in each node is no smaller than the key values in its children(if any)

Tree representation



Array representation



Max Heap

Operation-Creation

- Complete Binary Tree
→ Array Representation

```
template <class T>
Maxheap<T>::Maxheap(int _capacity = 10) : heapSize(0) {
    if (_capacity < 1) throw "Must be > 0";
    capacity = _capacity;
    heap = new T[capacity + 1];
}
```

```
template <class T>
class Maxheap {
private:
    void ChangeSize1D(int);
    T *heap;
    int heapSize;
    int capacity;

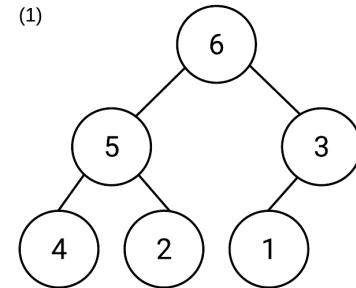
public:
    Maxheap(int);
    void Push(const T &);
    void Pop();
    bool IsEmpty() { return heapSize == 0; }
    T Top() { return heap[1]; }
    template <class T2>
    friend ostream &operator<<(ostream &, Maxheap<T2> &);
};
```

Max Heap

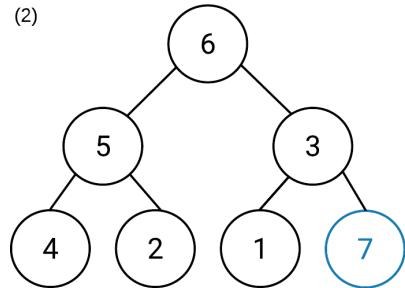
Operation-Insertion

- 일단 끝에 넣고(Complete Binary Tree),
- 아래에서 위로 비교하면서 있어야 될
곳으로 이동시킨다(Max Tree)
- $O(\log n)$ (트리의 높이 만큼 비교)

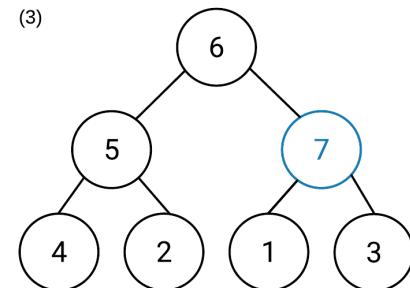
Inserting 7 into this heap



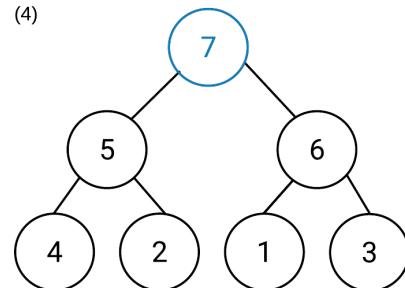
Starting with this max heap



Step 1: 7 is inserted at the bottom most, right most position



Step 2: Because 7 is bigger than its parent, the 3 node, it gets swapped



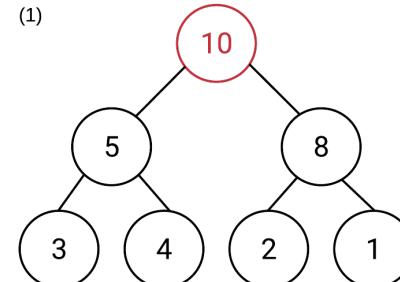
Step 3: Once again, 7 is bigger than its parent, the 6 node, so it gets swapped

Max Heap

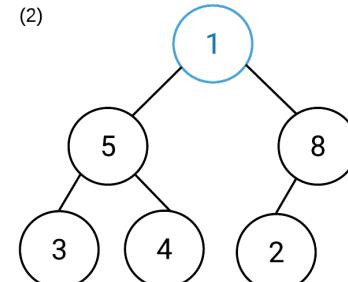
Operation-Delete

- root를 지우고 일단 맨 마지막 노드를 root에 둔다(Complete Binary Tree)
- 위에서 아래로 비교해가면서 있어야 할 곳에 둔다(Max Tree)
- $O(\log n)$ (트리의 높이 만큼 비교)

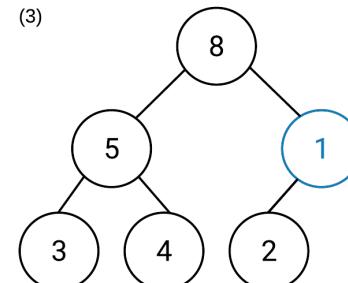
Deleting from this heap



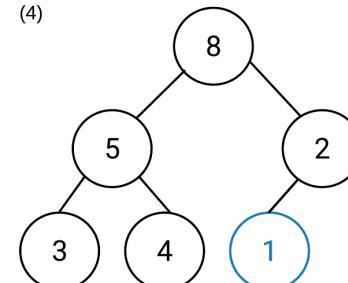
Starting with this max heap



Step 1: the bottom most, left most node, the 1 node, gets placed at the root



Step 2: Because 1 is less than both of its children, it swaps with the larger element, the 8 node



Step 3: Once again, 7 is bigger than its parent, the 6 node, so it gets swapped

Max Heap

Priority Queues

- Priority Queue는 빈번하게 Heap을 사용해서 구현됨.
- 큐의 일종

Max Heap Visualization

- 21,14,20,2,10,15

질문

- pemds81718@gmail.com
- 간단한 구글링으로 알 수 있는 내용은 답변하지 않습니다.

