Binary Heap

Showvik Biswas
Muhammad Ehsanul Kader

Department of CSE, BUET

August 29, 2022

The 14th Week of BUET

- One of the most gruesome experiences an undergraduate student at BUET can ever have.

The 14th Week of BUET

- One of the most gruesome experiences an undergraduate student at BUET can ever have.
- So much to be done, with such limited time!

To-Do List

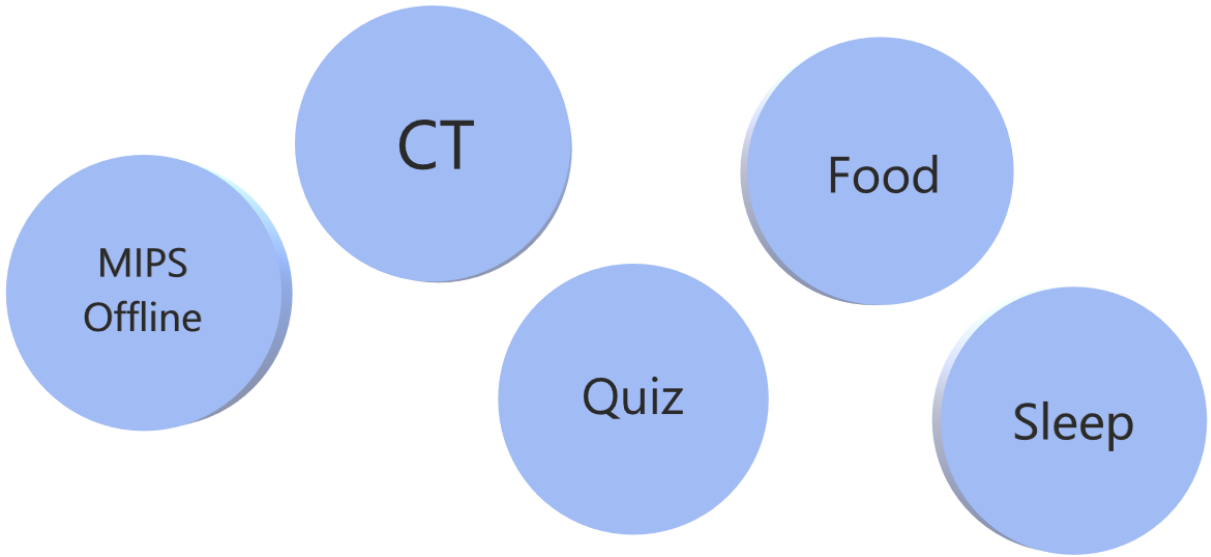

Figure: A few things to be done

Possible Solution

- Sort by priority?

Possible Solution

- Sort by priority?
- Most prioritized task available.

Possible Solution

- Sort by priority?
- Most prioritized task available.
- Results in something called a **priority queue**.

Possible Solution

- Sort by priority?
- Most prioritized task available.
- Results in something called a **priority queue**.

Definition

A priority queue is an abstract data-type in which each element has a priority associated with it.

The Priority Queue

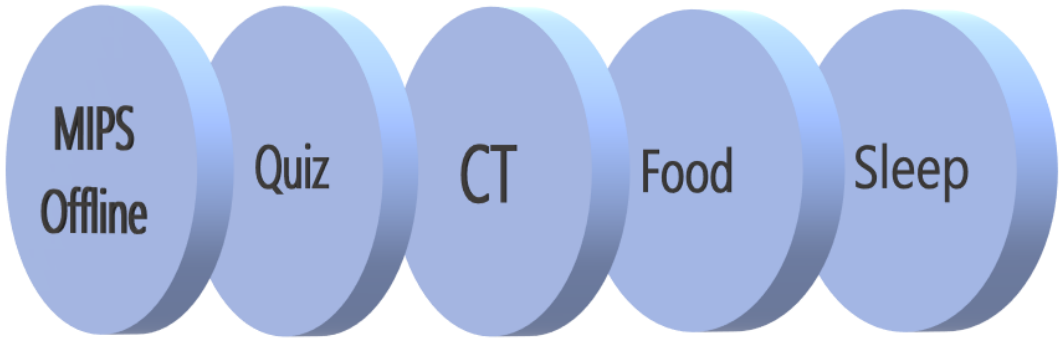

Figure: Sorting the tasks according to their respective priorities...

The Priority Queue

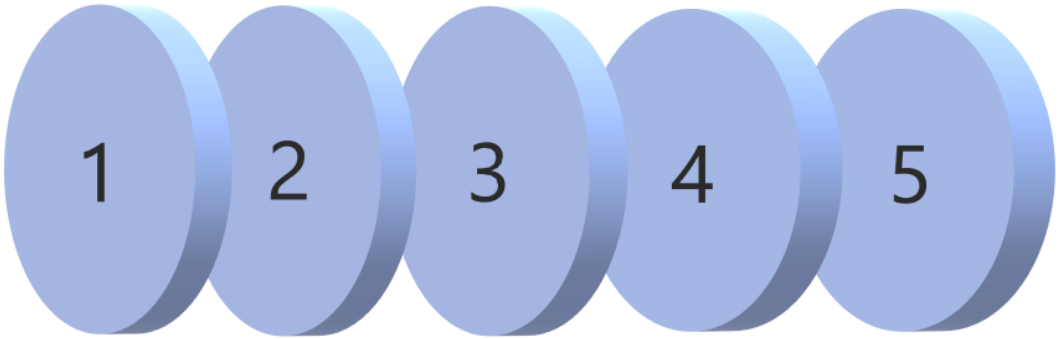

Figure: ...and assigning them priority numbers

Implementing the Priority Queue

- Most plausible option seems an array

Implementing the Priority Queue

- Most plausible option seems an array

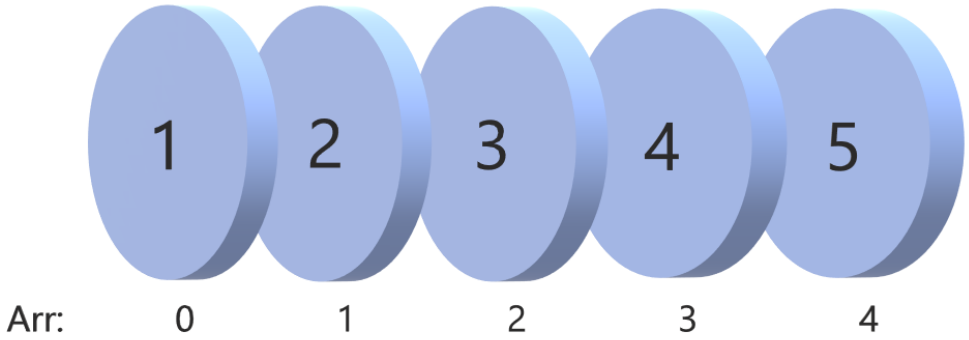

Figure: Array Implementation of PQ

The Catch

- What if a more important task comes along?

The Catch

- What if a more important task comes along?

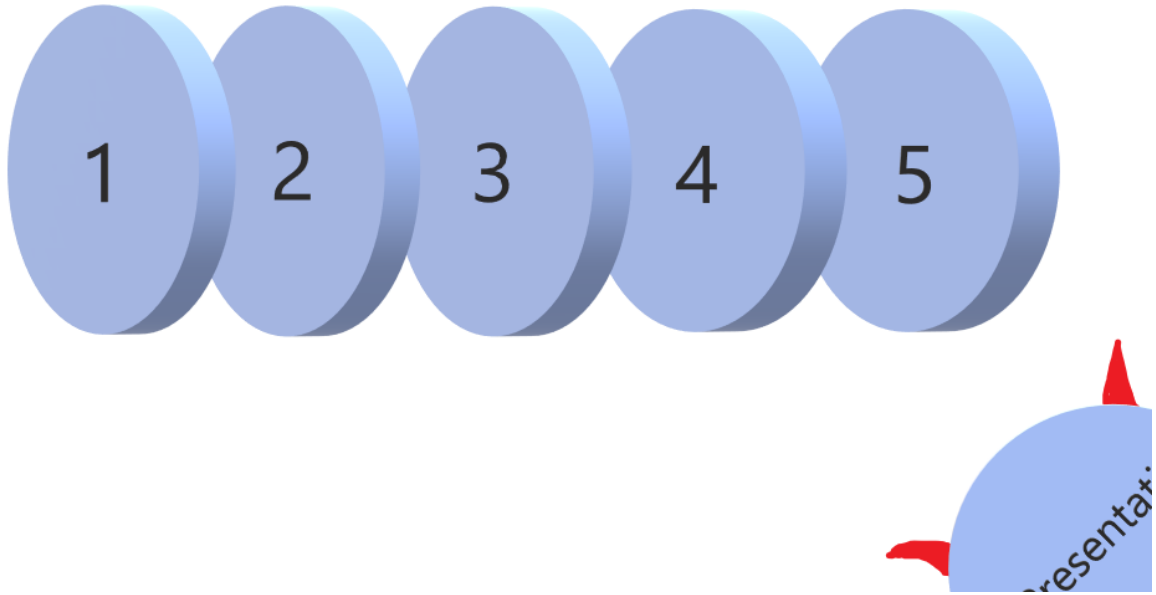

Figure: The LaTeXpresentation is obviously the most prioritized

An Alternative

- The array will have to be resorted, and this will be a trouble with limited storage and time.
- A better alternative: **binary heaps**.

Binary Heap: Definition

A binary heap is a binary tree with **two invariants**.

1. Every node is smaller (for min-heaps), or bigger (for max-heaps) than its descendant nodes.
2. The left sub-tree of a node must have an equal number of nodes, or have one more node than the right sub-tree.

A binary heap

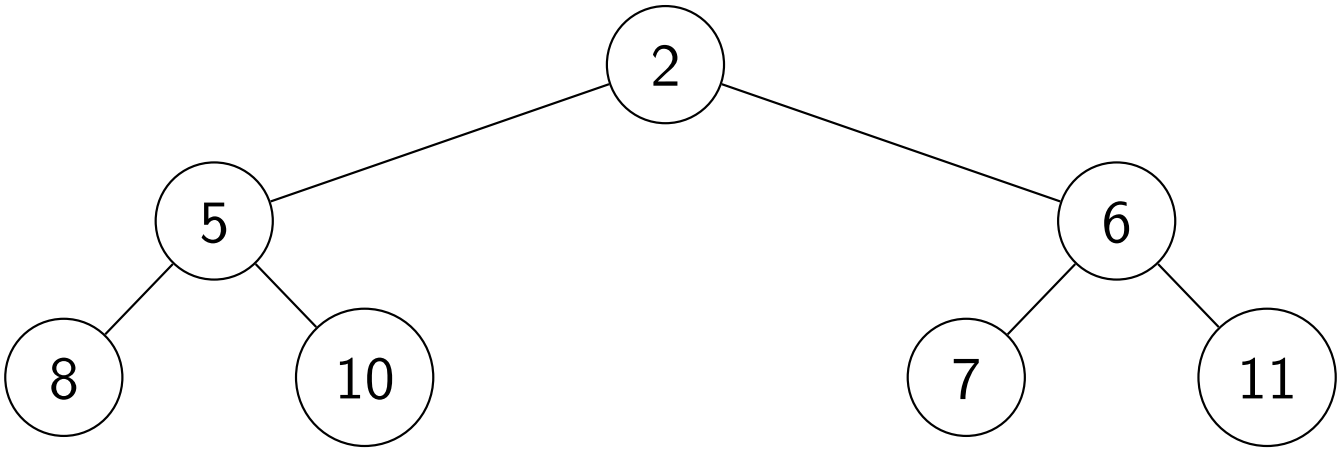

Figure: Binary Heap

Binary Heap: Operations

- Insertion
- Extract Min
- Find Min
- Deletion
- Decrease or increase key

Binary Heap: Operations

- Insertion
- Extract Min
- Find Min
- Deletion
- Decrease or increase key

Insertion

Insertion Steps

- Place the new node in the next available spot
- Push the new node upwards until heap property is satisfied.

Insertion

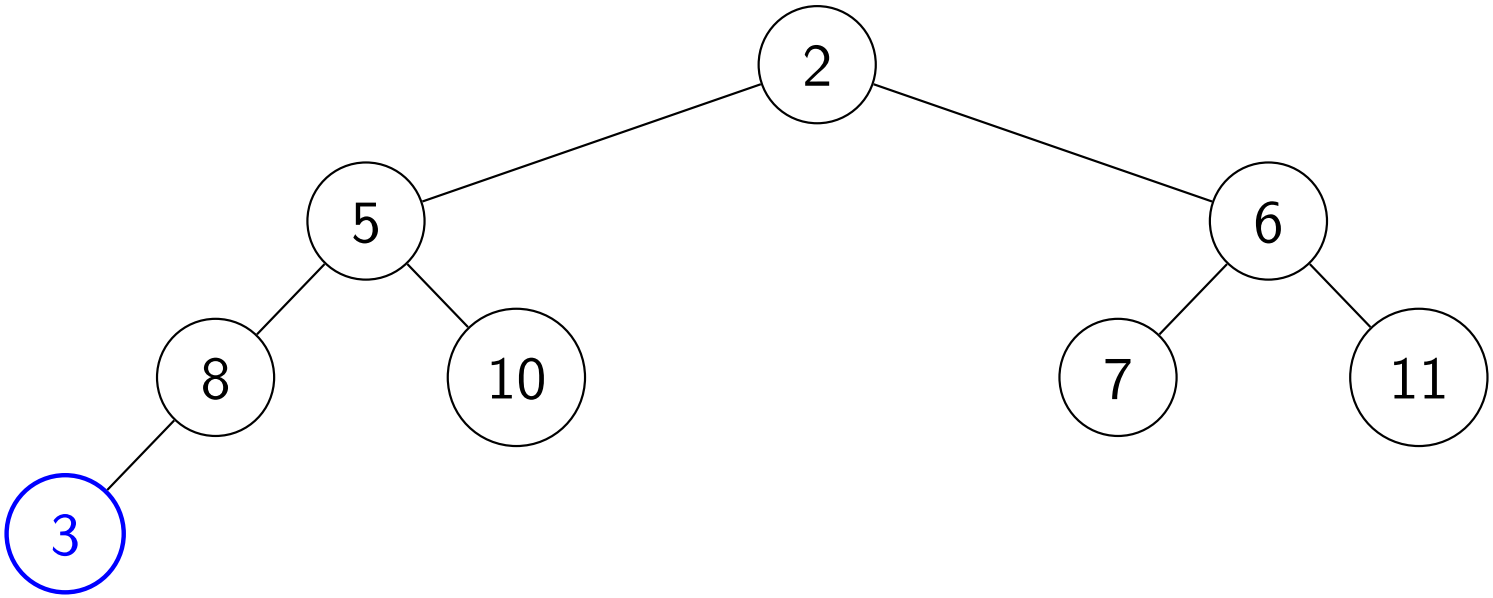

Figure: Inserting 3

Insertion

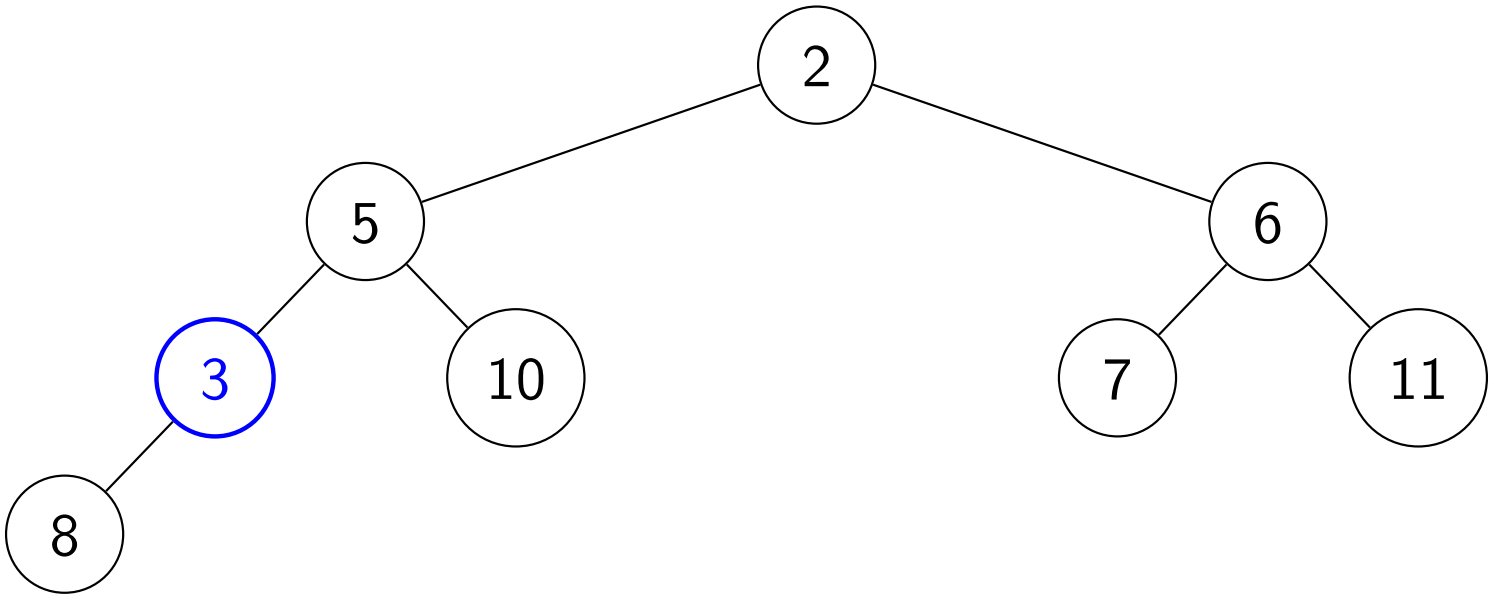

Figure: Restoring heap property

Insertion

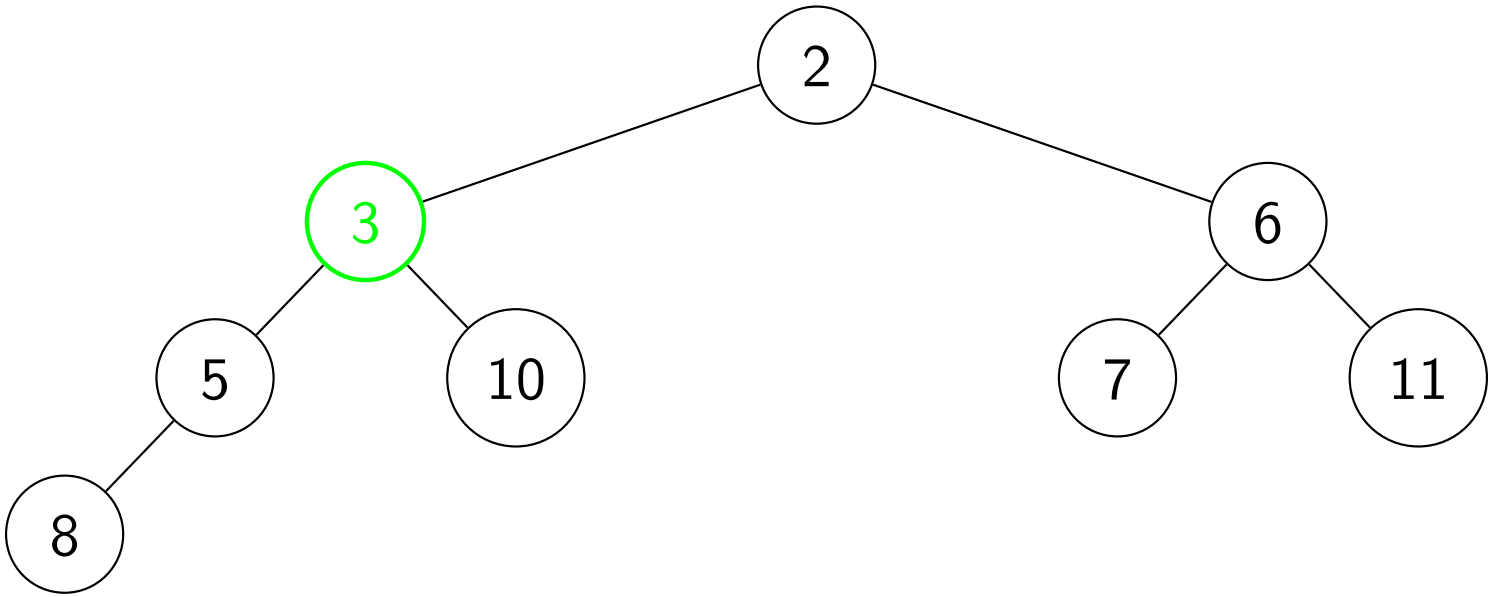

Figure: Restoring heap property

Extract Min

Extract Min

- Place the last node on the root.
- Push the out of place node downwards until heap property is satisfied (Heapify)

Extract Min

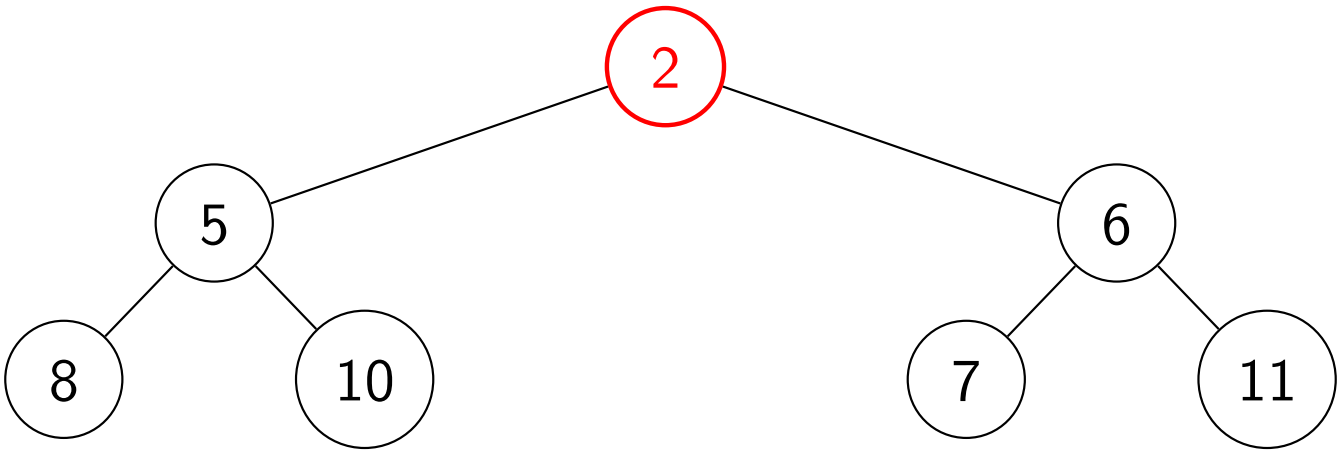

Figure: Place last element on root

Extract Min

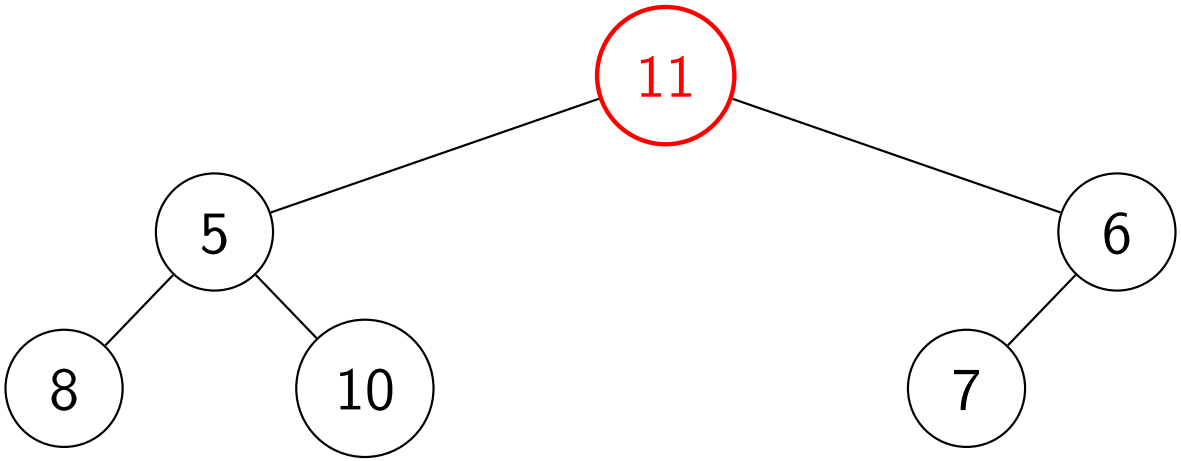

Figure: Heapify after extracting

Extract Min

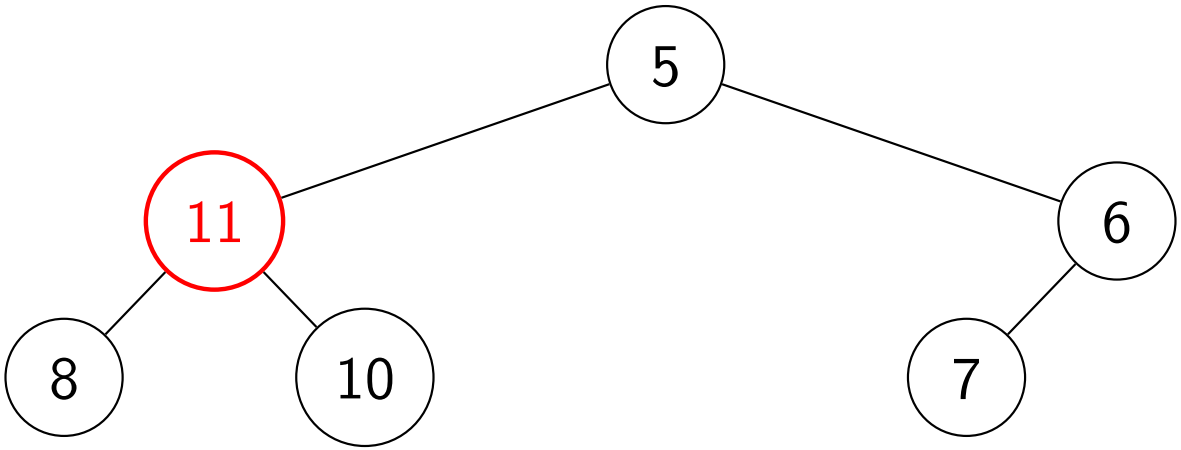

Figure: Heapify after extracting

Extract Min

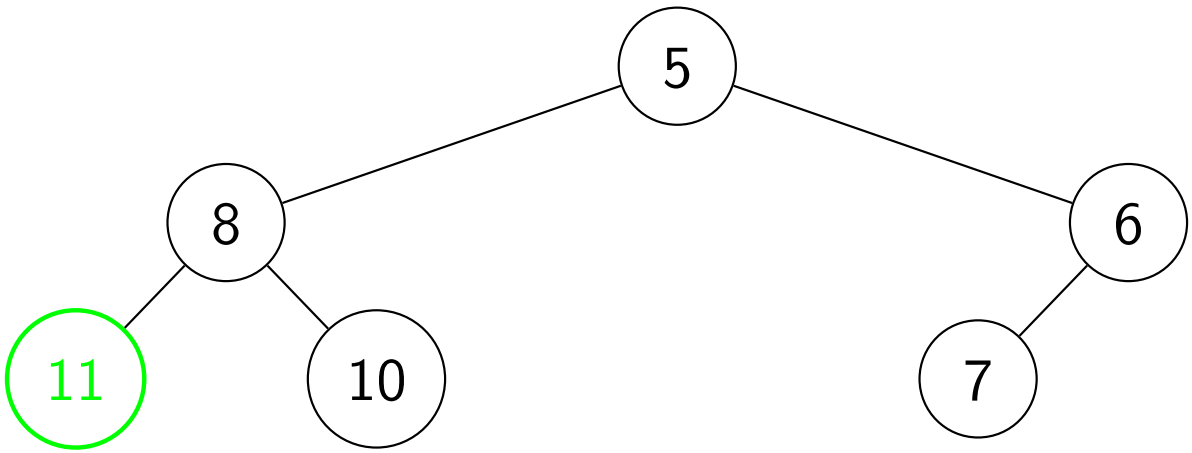

Figure: Heapify after extracting

Time Complexity

Operation	Time Complexity
Insert Value	$O(\log n)$
Extract Min	$O(\log n)$

Thank You!