



Heaps / Priority Queues



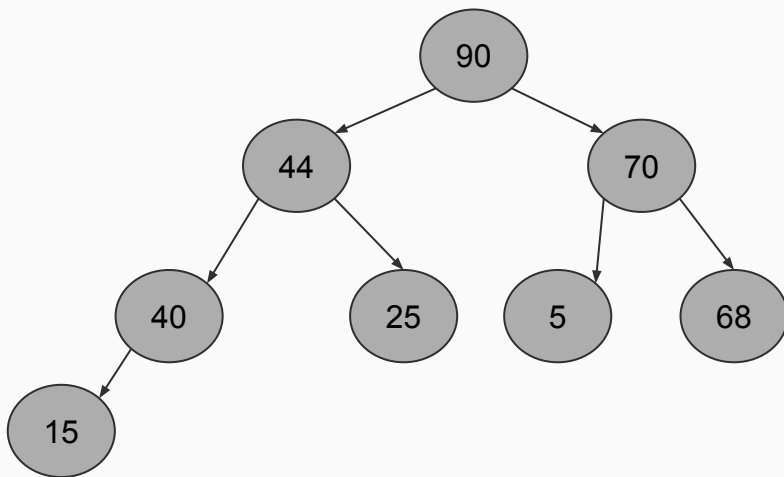
What is a heap / priority queue?

- The terms “heap” and “priority queue” are the interchangeable
- Partially ordered data structure
- Instead of “first in first out”, we want to pop out an element by priority
 - We can prioritize either the min value or the max value
- Properties of min heap
 - Structure
 - Complete binary tree
 - “complete” means that we fill out the tree from top to bottom, left to right
 - Order property
 - Every node’s value is less than its children
 - This makes the root the minimum
- Because a heap is a complete binary tree, we can actually represent it using an array



Heap Implementation

max heap



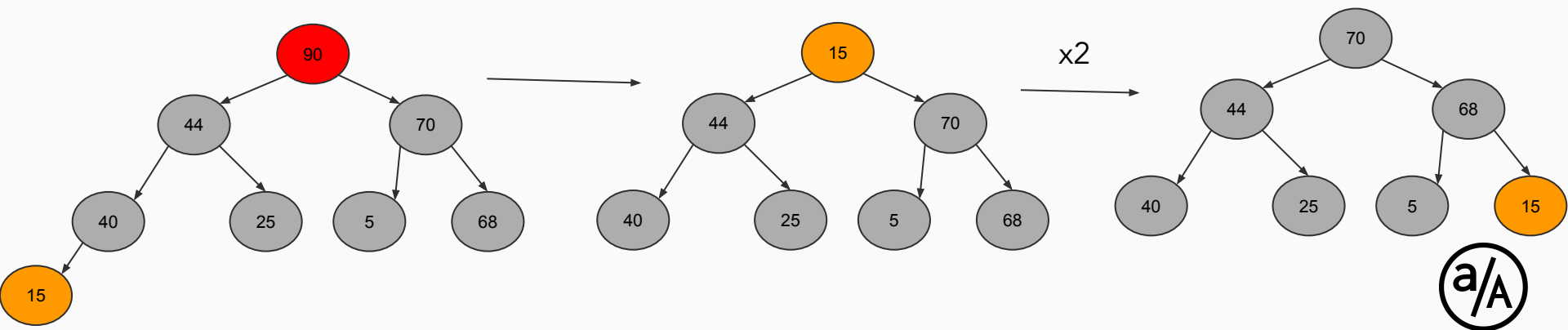
$\text{leftChild} = 2*i + 1$
 $\text{rightChild} = 2*i + 2$
 $\text{parent} = \text{Math.floor}((i-1)/2)$

val	90	44	70	40	25	5	68	15
index	0	1	2	3	4	5	6	7



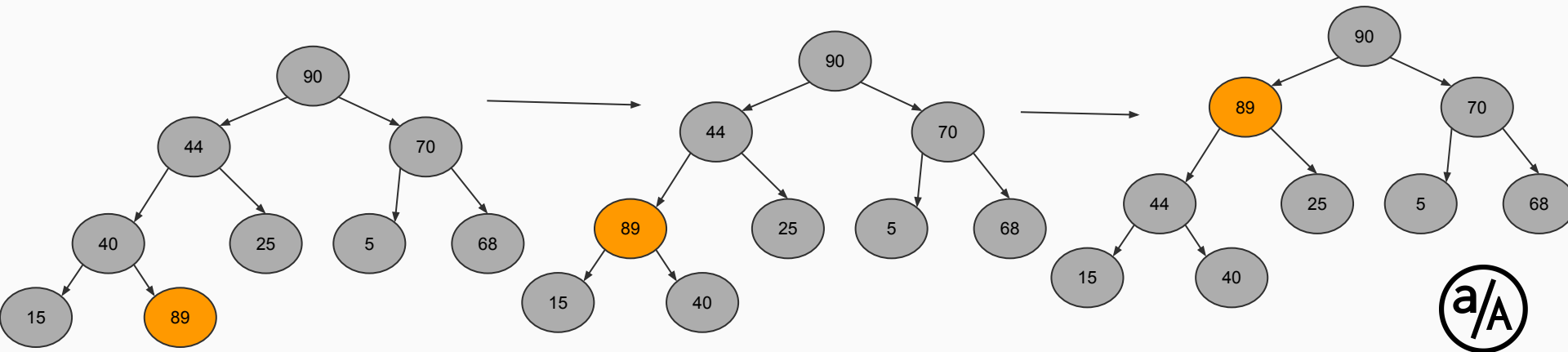
Operations of max heap

- poll
 - pop out top element
 - move last element to top
 - heapify down: recursively compare children, swap with bigger child



Operations of max heap

- push
 - insert element at the end
 - heapify up: recursively it with parent, swap if parent is larger
- heapify
 - In order to convert a set of values to a heap, we would need to push each of the values one by one into the heap



Runtimes

Operations	Big-O Time
Push	$O(\log(n))$
Poll	$O(\log(n))$
Heapify	$O(n)$
Peek	$O(1)$
Search	$O(n)$



Demos

- [Last Stone Weight](#)
- [Seat Reservation Manager](#)



Questions?



Let's practice!

- Review
 - [Kth Largest Element in a Stream](#)
 - [K Closest Points to Origin](#)

