Priority Queue

Overview

- A PQ is a queue where each element has a priority ranking
- Elements with a higher priority are dequeued first
- Elements must be comparable/ordered so priority can be assigned
- A PQ uses a heap to determine which item is to be dequeued (based on priority)
- **What is a Heap?**
    - A tree-based DS that satisfies the heap invariant: If A is a parent node of B then A is ordered with respect to B for all nodes, A, B in the heap
        - Max Heap: Root is the largest value
        - Min Heap: Root is the smallest value
    - * ***PQ's ARE NOT HEAPS***
        - PQ is an Abstract Data Type meaning it can be implemented with other DS also (such as an array list)
    - * ***ALL HEAPS MUST BE TREES***
        - A binary heap is a binary tree with the heap invariant. Every node must have two children
        - A complete binary tree is a tree in which at every level, except the last, is completely filled and all the nodes are as far left as possible
- **Implementation**
    - A heap is conveniently represented using an array
        - Let **i** be the parent node index
            - Left Child Index: **2i + 1**
            - Right Child Index: **2i + 2**
            - Parent Index: (i - 1) / 2

- **Adding Elements**
  - Elements are inserted at the bottom left position of the heap and are "bubbled up" to adhere to the heap property
- **Removing Elements**
  - The element to be removed is swapped with the bottom right element (leaf node) and the root node is "bubbled down" to adhere to the heap property
  - Polling - Remove root element (no searching required, O(log n))
    - If removeElement is not at root, element must be linearly searched to find the position of the element. Therefore, removal takes O(n).
    - A HashMap can be used to map each element to an index for instant search O(1)

Why Use It?

- Dijkstra's Shortest Path Algo
- When you need to dynamically fetch the next best or next worst element
- Used in Huffman coding
- BFS algos such as A* use PQs to grab the next most promising node
- Used by Minimum Spanning Tree algo

Big O Analysis (PQ with binary heap)

Aa Operation	Big O Notation	Explanation
Binary Heap construction	O(n)	
Adding	O(log(n))	Restoring the heap invariant
Deletion (Polling)	O(log(n))	Restoring the heap invariant
Deletion (RemoveAt)	O(n)	Linear Search (no hashmap) + Restoring the heap invariant

Operation	Big O Notation	Explanation
Peeking	O(1)	
Contains	O(n)	Linear Search

Code Implementation

- https://www.geeksforgeeks.org/priority-queue-using-binary-heap/