CS 209
Data Structures and Mathematical Foundations

04 / 05 / 2024

Instructor: Michael Eckmann

Today's Topics

- Questions?/Comments?
- Priority Queues
  - Binary Heaps
  - Let's implement a binary heap class and a priority queue class that stores its data in a binary heap

Next data structures

- Priority Queues
- Heap

Heaps

- Recall that a binary max heap is
  - A complete binary tree
  - And the ordering property is such that every node's data is >= its children node data
  - We will store it in a list (instead of as a tree of connected Nodes like we did with the BST)
    - a node at index i has children at
    - left child: 2*i + 1
    - right child: 2*i + 2
    - And its parent at: (i-1)//2
  - Operations:
    - Add
    - Remove largest (remove from the root)
    - Empty?

Heaps

- Heaps can be used to implement Priority Queues
  - Main operations of a priority queue
    - remove (highest priority item)
      - dequeue
    - add
      - enqueue
- For a Heap implementation of a priority queue, we would remove from the root (and then make sure the heap remains a heap by the process we described earlier.)
- For add, we would place at last slot and upward reheapify.

Heaps

- Let's create a class (named ItemAndPriority) that holds a data item of some type, and a priority value (int).
- Then let's create a class Heap that stores objects of them in a heap (implemented as a list).
- This Heap then can be used as a priority queue based on the priority value in the ItemAndPriority class.