Heaps and Tries

Abhinav Ashar
CS 61B: Data Structures and Algorithms

March 25, 2019

1 Heaps

Heaps are a form of binary trees that maintain some unique property. One popular heap is the min-heap, which cleverly organizes data such that it is very easy to access the smallest elements in the heap. A max heap does that same for accessing the largest elements. A property of a heap is that it must be complete or nearly complete such that every row of the heap is filled except for the last, which has nodes in the row that are **not** separated by an empty space.

For a min heap, every node needs to be smaller than its children. For a max heap, every node needs to be larger than its children. If the heap is currently set up so it does not follow these rules, then it must be heapified so that it does follow the rules. These links provide a great explanation of heapification:

a) Josh Hug's Slides

b) Video on Heapification

Heapification can only happen if you traverse in level order and bubble up or traverse in reverse level order and bubble down. The former takes $\mathcal{O}(N \log N)$ and the latter takes $\mathcal{O}(N)$ time.

Get Smallest/Biggest:
The smallest number for a min heap or the biggest number for a max heap is always the root node at the top, so this takes constant time.

Add
Add the number to the rightmost position in the last row, and heapify.

Remove

Remove the top root node, replace it with the rightmost node in the last row, and heapify.

2 Tries

Tries are a tree structure that are often used to represent words. The top root node is a sentinel, or dummy node. It's children are the first letters of the words in a trie. From there, if you traverse down one path of the tree to a leaf, you will get one word in the trie. Thus, all the words in a trie are simply all the paths from root to leaves. If you are checking if a certain word exists in the trie, check if the first letter of this certain word is a child of the sentinel. If not, then the word doesn't exist. Else, check the rest of the sequence of letters to see if they exist. The former takes constant time and the latter takes linear time.