Team PickledApricot - Joan Chirinos, Mohtasim Howlader, Kaitlin Wan

APCS2 pd8

HW48 – Heap o’Trouble

2018-05-14

Left child index = 2N + 1

Right child index = 2N + 2

N is the parent of these children.

Algorithm for adding to a Min Heap:

add (int element)

1. If the heap is empty, then element is the root of a new heap
2. If not, add element to the end of the heap arraylist
3. While element’s index is not 0
    a. Compare element to its parent. Its parent index is ((index of element – 1) / 2).
    b. If element is smaller than its parent, then swap places.
    c. If element is greater than or equal its parent, then leave element there. Break while loop and element is at its final position.

remove (int index)

1. If element at index is a leaf (meaning it has no right child or left child, which is determined if both equations results in a number that’s greater than the size of the heap), simply remove from the heap with remove(index).
2. If element at index has only 1 child, then remove the element at index and promote the child to its place.
3. If element at index has 2 children
    a. Keep on traversing through children until a leaf is found
    b. Remove element at index and promote the leaf to its place. This node’s name is “promoted”.
    c. While “promoted” is not a leaf
        i. If larger than one of its children, swap places with that children
        ii. If less than or equal to its children, make no swaps. Break while loop.