

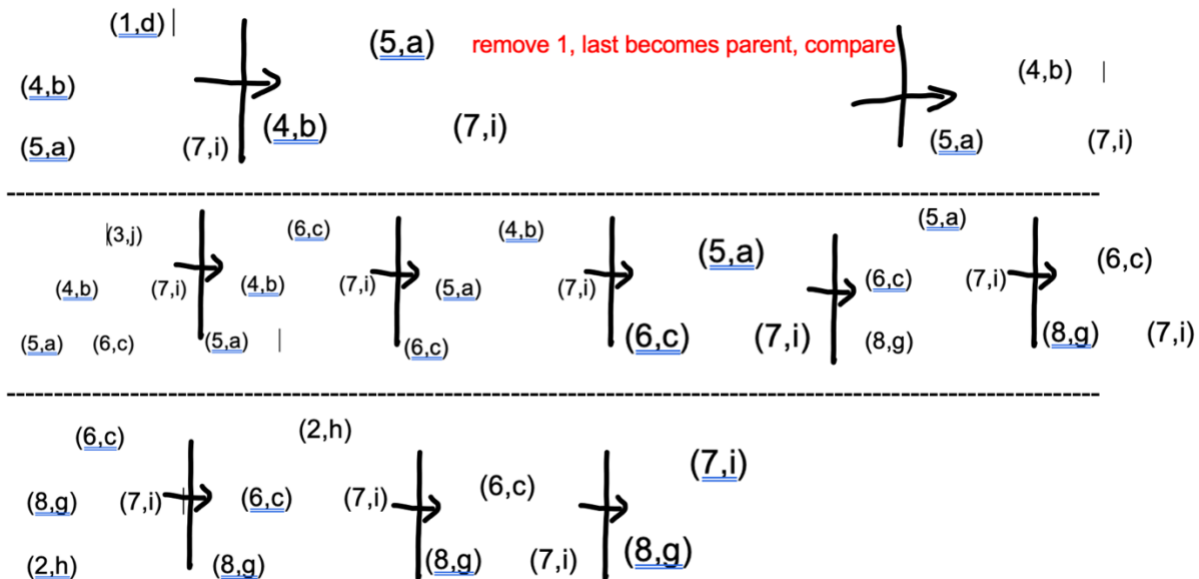
Homework 4

CMSC 341 - Section 2 - Spring 2021 Due Date April 4, 2021, 11:59 PM

1 - What is the output of the following sequence of priority queue ADT operations. The entries are key-value pairs. Only show the values, there is no need to write keys in the answer. The keys are integer numbers and the values are alphabet characters. (10 points)

insert(5,a), insert(4,b), insert(7,i), insert(1,d), removeMin(), insert(3,j), insert(6,c), removeMin(), removeMin(), insert(8,g), removeMin(), insert(2,h), removeMin(), removeMin().

Min key value is the priority: End result is (7,i), (8,g), If output is in-order traversal- g, i



2 - Explain why we do not need to consider the case where the right child of a node r is an internal node and its left child is a leaf node, when implementing the down heap operation in a heap data structure? (10 points)

There wouldn't be a case of just a right internal node and a left child to that node that's a leaf. In insertion, the tree is filled from the left to the right which is then checked to preserve the trees priority property. In removal, the tree is almost always complete with the way its inserted so there will never be a case described above.

3 - Implement the function `void Heap::upHeap(int anIndex)` . Using your implementation test the enqueue function in the Heap class. Make sure after an insertion the heap property is restored in the binary tree. Your algorithm should be correct, and your code should compile. (40 points)

4 - Implement the function `void Heap::downHeap(int anIndex)` . Using your implementation test the dequeue function in the Heap class. Make sure after an insertion the heap property is restored in the binary tree. Your algorithm should be correct, and your code should compile. (40 points)

Submissions:

Questions 1 and 2 - You need to submit your answers in the PDF file to Blackboard.

Questions 3 and 4 - You need to submit your heap.cpp file including the implementation of upHeap and downHeap functions to Blackboard.