

Assignment 3 – Write Up

Mike Kim

November 27th, 2022

For PQ.h file:

Functions that I have additionally implemented are operations within AVL tree (the whole header file), `percolateUp(private)`, `updatePrio_percolateUp(private)`, `percolateDown(private)`, `displayHeap(private)`, and `printSize(private)`

`PercolateUp` and `percolateDown`:

These functions are there because it allows us to have a proper priority queue, by percolating up and down through the data structure, ensuring that the heap and tree is in proper order at all times.

`updatePrio_percolateUp`

This function was specifically designed to percolate up after updating the priority of the element. It ensures that the priority queue and heap are in correct order by percolating up and swapping if necessary.

`displayHeap`

This private function was designed to have a visual reference while testing out the functions – it helps tremendously while debugging the codes

`printSize`

This private function is simply used to get the size of the heap. It is used for debugging purposes after testing out the functions such as `deleteMin()`.

Part 2: Each class other than PQ class (struct `AvlNode` and class `AvlTree`)

`AvlNode`

It is used to create binary heap structure – author-supplied code.

`AvlTree`

It is used to perform AVL-tree like operations – author-supplied code as well.

Both classes were nested within the PQ class for easier access when performing operations

