Daee kang ass 8 sec 1002

Using min and Max heaps, we were able to find the medians at any given location. To find a median dynamically, we need to separate the data into two halves where all the lower numbers are in one and the all the higher numbers in one. With the heaps, we are able to keep track of this dynamically and also keep tracking of the highest number in the lowest half and the lowest in the highest half making it finding a median very easy.

Big Oh –

Insert : O(log n)

DeleteMin: O(log n)

DeleteMax: O(log n)

reheapUp: O(log n)

reheapDown:O(log n)

heapify: O(n)

printHeap: O(n)

DynamicMedian algorithm: O(n log n)

Resize : O(n)

Difference of insertion and heapify:

When you use heapify, you one need to call it once which it would then reheap down. When you do the insertion method, you are consistently calling the insertion method which in turn uses the reheap up method and it will eventually become O(n log n). With heapify it is just O(n).