

- 1- Extract() extracts the data from heap with minimum key value. Decrease() decreases the key value and moves it to the root. Insert() inserts a new element to the heap.

The asymptotic upper bound for extract operation =  $O(\log n)$

The asymptotic upper bound for decrease operation =  $O(\log n)$ .

The asymptotic upper bound for insert operation =  $O(\log n)$ .

As we see, upperbound of all functions are the same and  $O(\log n)$  since the number of operation each iteration is similar to each and they connected to Heapify() function

- 2- The asymptotic upper bounds of the all functions are  $O(\log n)$ , and we make calls like  $O(n)$ , but in these calls not every element is the same depth, so the tight limit goes to  $\theta(n)$ . Normally the asymptotic upper bound should intersect with  $n * \log n$  but as we know  $\theta(n)$  because not every node is at maximum depth. For example, some nodes are closer to the root, while others are further away. Our hypothesis has been corrected in the table as there is a straight line showing that the asymptotic upper bound is  $O(n)$ .

