Alexandria University,

Faculty of engineering,

Data structures.

Assignment 1.

Name: Mohamed Mohamed Abdlhakem. (43)

**Problem statement:**

Heap implementation:

It is required to implement the following some basic procedures:

• The MAX-HEAPIFY procedure, which runs in O (log n) time, is the key to maintaining the max-heap property. Its input is a root node. When it is called, it assumes that the binary trees rooted to the left and right of the given node are max-heaps, but that the element at the root node might be smaller than its children, thus violating the max-heap property.

• The BUILD-MAX-HEAP procedure, which runs in linear time, produces a max-heap from an unordered input array.

• The HEAPSORT procedure, which runs in O (n log n) time, sorts an array in place.

• The MAX-HEAP-INSERT, and HEAP-REMOVE-MAX procedures, which run in O (log n) time, allow the heap data structure to implement a priority queue.

Sorting techniques:

• It is required to implement the heap sort algorithm as an application for binary heaps. It is advised to compare the running time of your implementation against:

– An O (n^2) sorting algorithm such as Selection Sort, Bubble Sort, or Insertion sort.

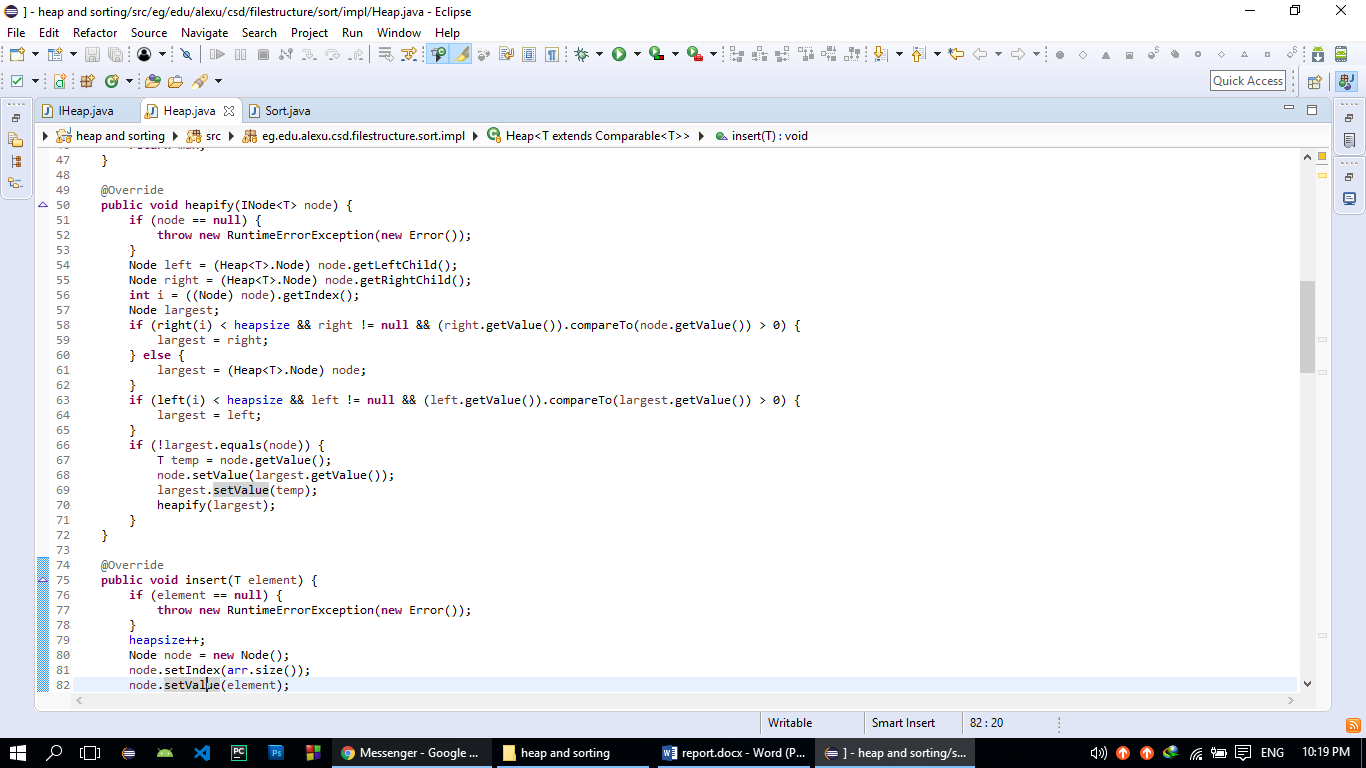
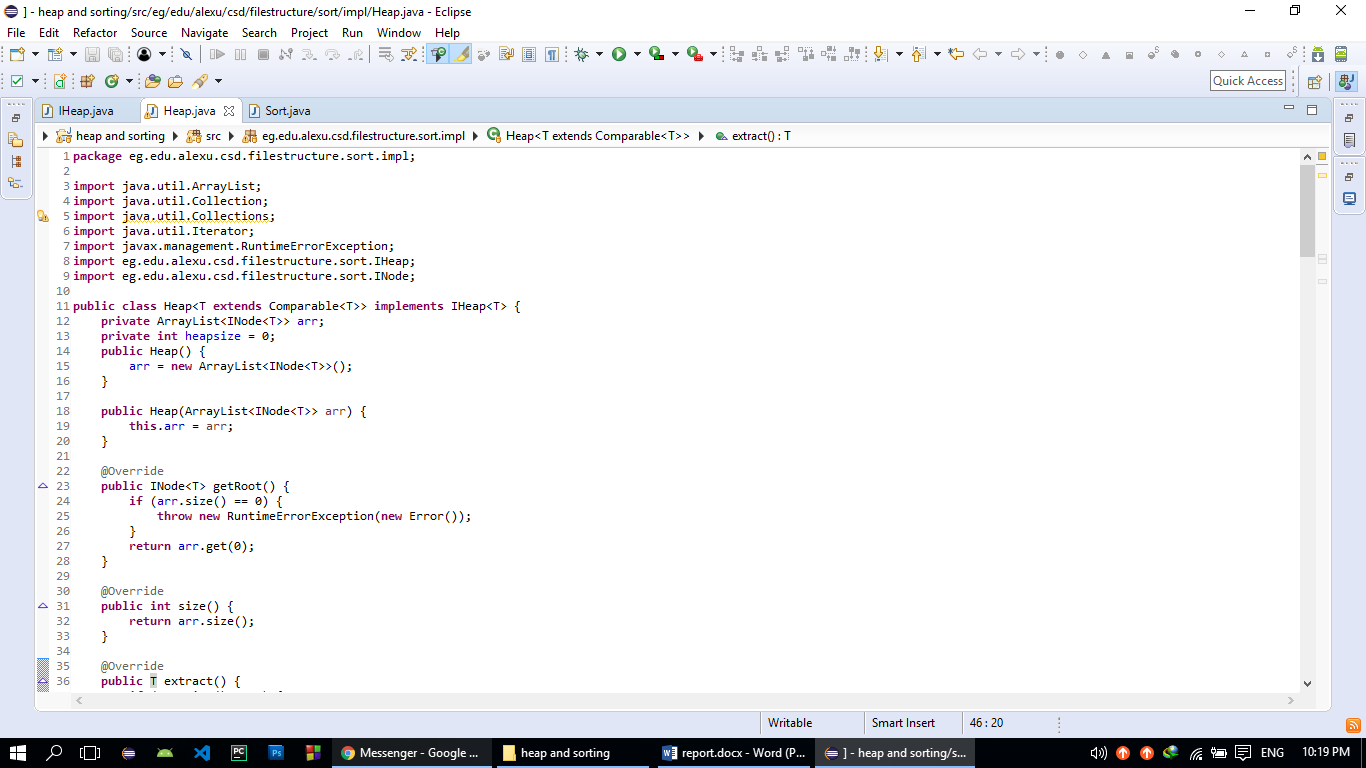
– An O (n log n) sorting algorithm such as Merge Sort or Quick sort algorithm in the average case.

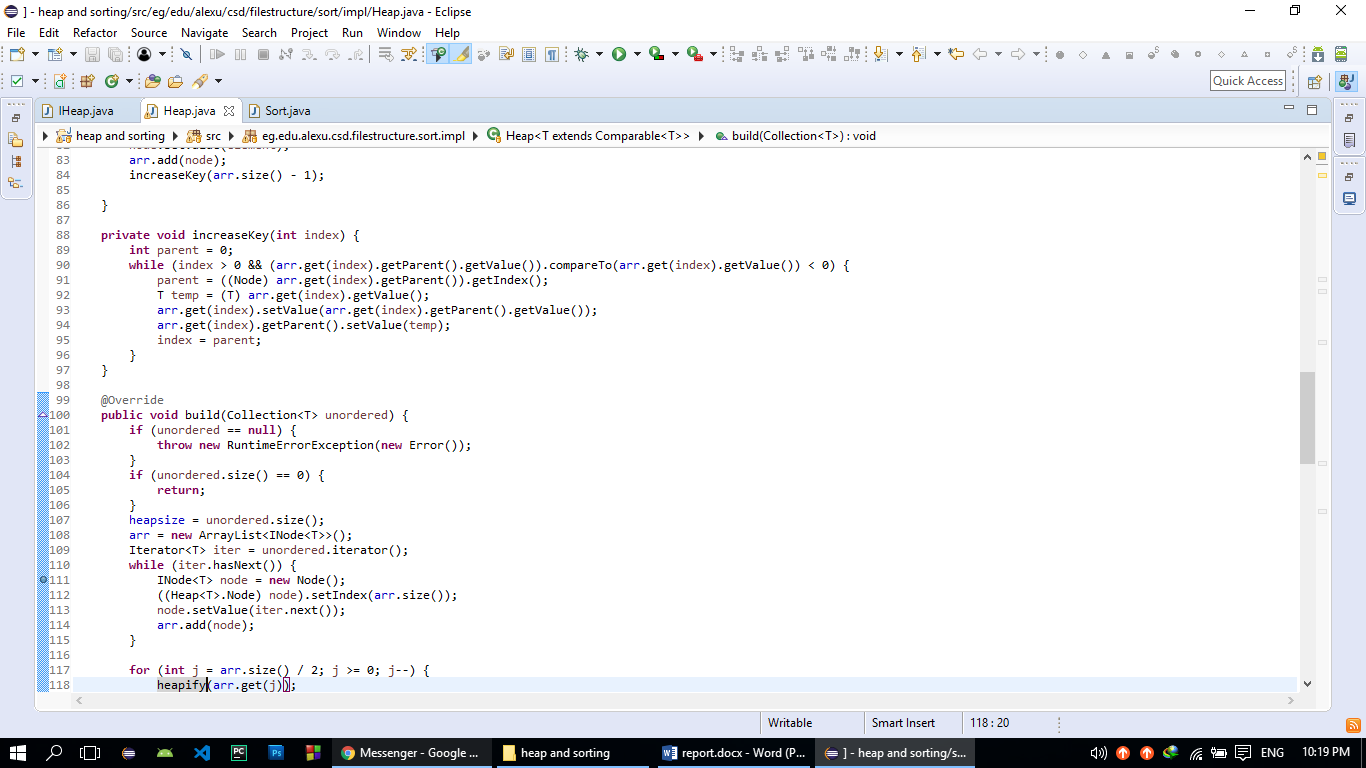
• In addition to heap sort, it is required to implement any of the sorting algorithms from each class mentioned above, O (n log n) and O (n^2). For example, I choose to implement Merge Sort and Bubble Sort.

**Data structures used:**

* Java Array list: to store the nodes of the binary heap. The order of access its element is O (1) because of its random access property.

**Code snippets:**



****

