[1] Heap-Sort algorithm

The Heap-Sort algorithm relies on the following main steps:

Build-Max-Heap: Convert the unsorted array into a max-heap.

Heapify: Maintain the max-heap property for a subtree, given the root and size.

Heap-Sort: Use the max-heap to sort the array

Time Complexity:

- **Heapify: Operates in O (log n) because it traverses the height of the heap.**
- **Heap-Sort:**
    - **Build-Max-Heap: Executes O (n) because it calls Max-Heapify for n/2 nodes with decreasing levels of work.**
    - **The for loop in Heap Sort runs n−1 times.**
    - **Each iteration involves an O (log n) Heapify call after a swap.**

The overall time complexity of Heap sort algorithm is O (E log E).

Space Complexity (Heap sort is in-place):

The overall space complexity of Heap sort algorithm is O (1)