





Quad Layout ▼









Prompt

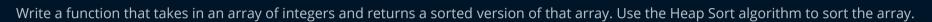
Scratchpad

Our Solution(s)

**Video Explanation** 

**Difficulty:** Category: Sorting Successful Submissions: 6,008+

# Heap Sort ○ ☆



If you're unfamiliar with Heap Sort, we recommend watching the Conceptual Overview section of this question's video explanation before starting to code.

## Sample Input

$$array = [8, 5, 2, 9, 5, 6, 3]$$

#### Sample Output

### Hints

#### Hint 1

Divide the input array into two subarrays in place. The second subarray should be sorted at all times and should start with a length of 0, while the first subarray should be transformed into a max (or min) heap and should satisfy the heap property at all times.

## Hint 2

Note that the largest (or smallest) value of the heap should be at the very beginning of the newly-built heap. Start by swapping this value with the last value in the heap; the largest (or smallest) value in the array should now be in its correct position in the sorted subarray, which should now have a length

of 1; the heap should now be one element smaller, with its first element out of place. Apply the "sift down" method of the heap to re-position this out-of-place value.

#### Hint 3

Repeat the step mentioned in Hint #2 until the heap is left with only one value, at which point the entire array should be sorted.

## **Optimal Space & Time Complexity**

Best: O(nlog(n)) time | O(1) space - where n is the length of the input array Average: O(nlog(n)) time | O(1) space - where n is the length of the input array Worst: O(nlog(n)) time | O(1) space - where n is the length of the input array