

Word Occurrence Counter

Heap Sort algorithm

Key Features

- Counts word occurrences in input text
- Uses Heap Sort
- Fastmail inspired UI elements

Heap Sort Algorithm

Time Complexity: $O(n \log n)$

Key steps:

1. Build max heap from input array
2. Keep taking out the largest number and rebuild the heap.
3. Result: Sorted array in descending order

Heap Sort Implementation

```
function heapSort(arr) {  
  // Build max heap  
  for (let i = Math.floor(arr.length / 2) - 1; i >= 0; i--)  
    heapify(arr, arr.length, i);  
  
  // Extract elements from heap one by one  
  for (let i = arr.length - 1; i > 0; i--) {  
    [arr[0], arr[i]] = [arr[i], arr[0]];  
    heapify(arr, i, 0);  
  }  
  return arr.reverse(); // For descending order  
}
```

Fastmail Inspired Design

- Background color: Starts dark blue, slowly changes to light blue
- Text style: Uses Open Sans, a clear and easy-to-read letter type
- Look and feel: Simple design with soft, curved edges
- Screen fit: Adjusts to look good on big and small screens