UNFAMILIAR SORTING TECHNIQUES

1. Counting Sort

Counting Sort is a non-comparison sorting algorithm that counts element occurrences, stores cumulative sums, and places elements in order. It works in O(n + max) time, making it efficient for small-range integers.

1. Radix Sort.

Radix sort works by sorting each digit from least significant digit to most significant digit. So in base 10 (the decimal system), radix sort would sort by the digits in the 1's place, then the 10's place, and so on. To do this, radix sort uses counting sort as a subroutine to sort the digits in each place value.

1. [Heap Sort](https://visualgo.net/en/heap)

Heap sort is a sorting algorithm that organizes elements in an array into a binary heap, and then sorts that heap by moving the largest element in the array.

1. [Balanced BST Sort](https://visualgo.net/en/bst).

The idea is to store the elements of the tree in an array using inorder traversal. Inorder traversal of a BST produces a sorted array. Once we have a sorted array, recursively construct a balanced BST by picking the middle element of the array as the root for each subtree.

Why am I not mastering them now?

Adequate time to master the underlying techniques and how the sorting algorithm works is currently an issue. This will be mastered as we progress into its usage in JavaScript