**Sorting Algorithms**

1. **Bubble Sort**

Bubble sort is an easy-to-understand algorithm. It goes through a list multiple times, comparing each pair of adjacent numbers and moves the bigger one to the right. It keeps doing this until all elements are in order.

Even though it looks simply, it’s not very efficient because it takes more time, especially for bigger lists.

**Big-O Notation**

Worst Case: O (n^2) => Array is the largest

Average Case: O (n^2) => Random array

Best Case: O (n) => Array is already in order

**metin, yazı tipi, ekran görüntüsü, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

1. **Insertion Sort**

Insertion sort scans the array from left to right and insert each value into correct position. Each new value is compared with the previous value and inserted into correct position.

It’s better than bubble sort. But it’s not efficient for big list like bubble sort.

**Big-O Notation**

Worst Case: O (n^2) => Array is the largest

Average Case: O (n^2) => Random array

Best Case: O (n) => Array is already in order

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

1. **Merge Sort**

Merge sort is a divide and conquer algorithm. It works dividing to array into smaller parts and then merging them. It splits to array into two halves, after that recursively sorts to halves. Finally, it merges the sorted halves to single sorted array.

It can be used for large dataset. But not for small datasets. Because requires to extra space for all time.

**Big-O Notation**

Worst Case: O (n log n) => Array is the largest

Average Case: O (n log n) => Random array

Best Case: O (n log n) => Array is already in order

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

1. **Quick Sort**

Quick Sort is a divide and conquer algorithm same as merge sort. Firstly, select a pivot value. After that divide array into two groups.

* Bigger than pivot
* Smaller than pivot

Recursively apply each sub-array. When elements sorted merge them together. It best for big datasets (faster than merge sort), it does not use extra memory. But if pivot selection is bad may impair performance.

**Big-O Notation**

Worst Case: O (n^2) => Pivot is the largest value in the array

Average Case: O (n log n) => Random array

Best Case: O (n log n) => Array is already in ordermetin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

1. **Bucket Sort**

Bucket sort is a dividing value into a several groups (buckets) and then sorting each group. Each bucket is using another sorting algorithm like insertion sort.

It very fast, but not stable. It can be using large datasets.

**Big-O Notation**

Worst Case: O (n log n) => Array is the largest

Average Case: O (n log n) => Random array

Best Case: O (n log n) => Array is already in order

metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.