ReadMe

- I implemented the comparison of sorting algorithms using python as programming language on jupyter notebook
- Open the file “Hemanth_Gaddipati_Project1.ipynb” in the jupyter notebook.

I executed the sorting algorithms for 3 different types of inputs:

- Shuffled input
- Sorted input
- Reversely sorted input

1) Shuffled input:

Input array is: [2, 4, 3, 1, 10, 5, 9, 7, 6, 8]

Time taken to execute insertion_sort is : 0.0

Insertion Sort:1 2 3 4 5 6 7 8 9 10

Time taken to execute mergeSort is : 0.0

Merge Sort:1 2 3 4 5 6 7 8 9 10

Time taken to execute heapSort is : 0.0

Heap Sort:1 2 3 4 5 6 7 8 9 10

Time taken to execute quicksort is : 0.0

Quick Sort:1 2 3 4 5 6 7 8 9 10

Time taken to execute modified_quicksort is : 0.0

Modified Quick Sort:1 2 3 4 5 6 7 8 9 10

2) Sorted input:

Input array is: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Time taken to execute insertion_sort is : 0.0

Insertion Sort for sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute mergeSort is : 0.0

Merge Sort for sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute heapSort is : 0.0

Heap Sort for sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute quicksort is : 0.0010004043579101562

Quick Sort for sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute modified_quicksort is : 0.0

Modified Quick Sort for sorted array:1 2 3 4 5 6 7 8 9 10

3) Reversely Sorted input:

Input array is: [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

Time taken to execute insertion_sort is : 0.0

Insertion Sort for reversely sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute mergeSort is : 0.0

Merge Sort for reversely sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute heapSort is : 0.0009942054748535156

Heap Sort for reversely sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute quicksort is : 0.0

Quick Sort for reversely sorted array:1 2 3 4 5 6 7 8 9 10

Time taken to execute modified_quicksort is : 0.0

Modified Quick Sort for reversely sorted array:1 2 3 4 5 6 7 8 9 10