

n^2 vs $n\log_2 n$ Sorting Algorithms

Learning Outcomes:

- Modify the existing sorting algorithms to accommodate varied sizes of inputs.
- Plot the result of the n^2 and $n\log_2 n$ on a graph to visualize their performance differences.

Procedure

1. Modify the existing code base for bubble sort, insertion sort, selection sort, merge sort and quicksort so that it would dynamically accommodate an increasing size of input.
2. Test your modification with a randomized and increasing input size according to your preference. E.g. starting input is size is 1000 elements increasing by 1000 every cycle up to 20000 elements.
3. Record in a file (.txt) the time of each algorithm with respect to randomized input and different sizes of input.
4. Create a line graph (preferably use Python programming in Jupyter notebook) with size of input at the x-axis and time on the y-axis.
5. Plot in a graph the performance of the bubble sort, insertion sort, selection sort, merge sort and quicksort in your Python Jupyter notebook
6. Submit the modified sorting algorithm code (in .cpp/c and .ipynb notebook)