Algorithm Visualization

Visualizing the following well-known algorithms on large input sizes.

Algorithm	Input
Max heap: Extract max with heapify.	Randomly generate a heap with at least 2000 nodes
AVL tree: Search, insertion, and deletion in AVL trees, including balancing the tree	Randomly generate an AVL tree with at least 1000 nodes
Dijkstra's algorithm. Visualize the algorithm progression and the final path	Randomly generate a graph with at least 10,000 nodes

The students can choose GUI framework (e.g. Javascript, Python, etc.) at his/her discretion.

The algorithm implementations must be correct. The UI can use randomly-generated inputs, i.e. it does not have to take input from the user.

The student must demonstrate the visualization of all the algorithms above. The visualization must include both how an algorithm progresses and the final result. Partial submissions are not accepted.

Submission of code and a short project report is required.

The deadlines will be the same as the last project's deadline for the rest of the class.