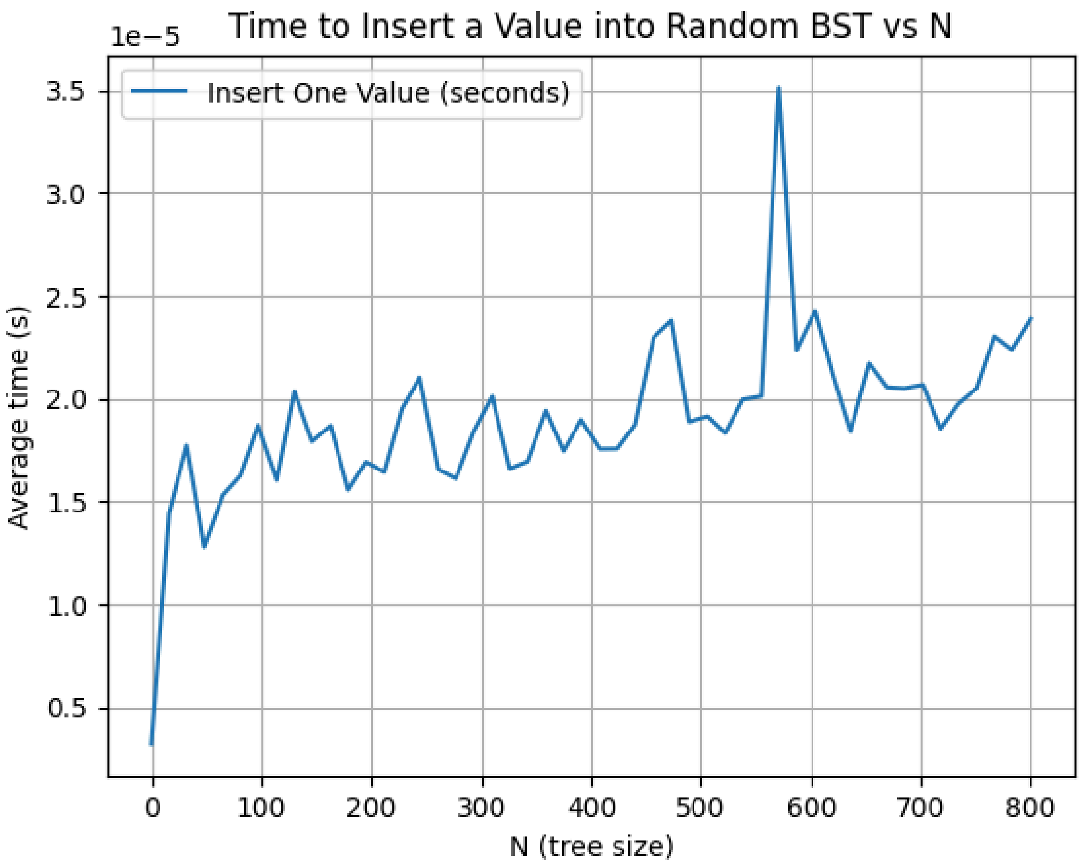
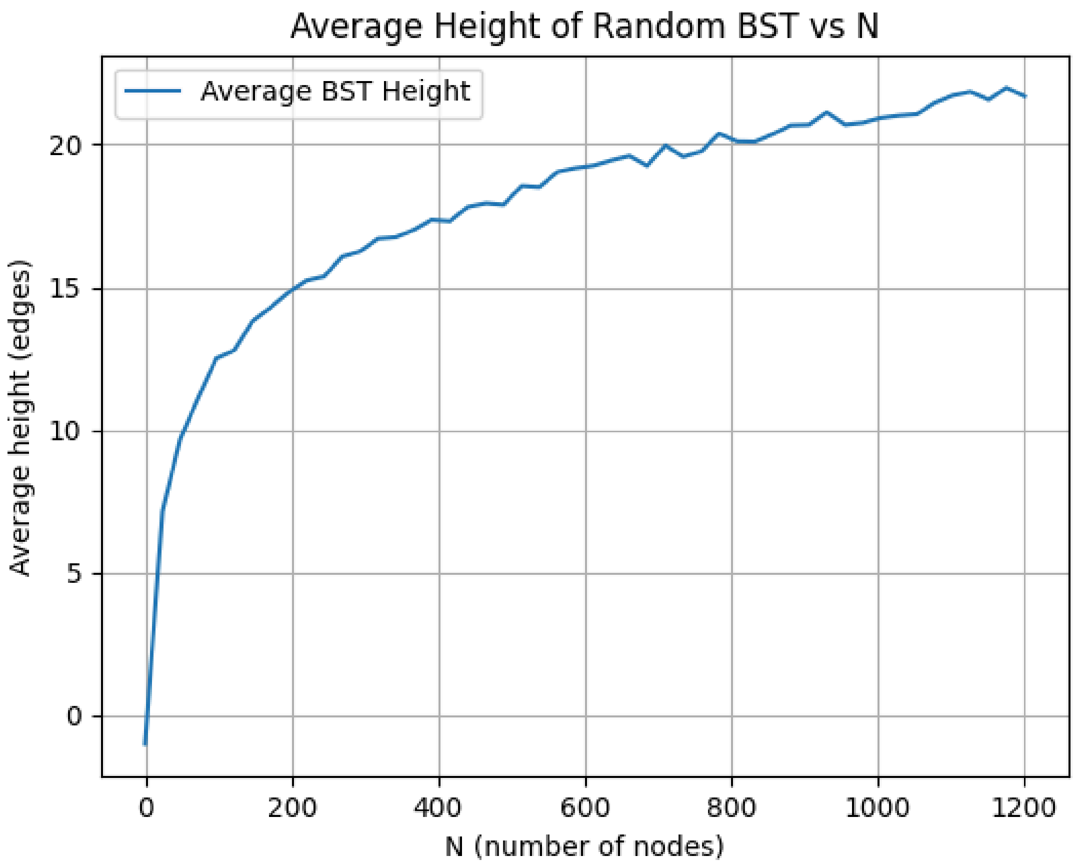
Insert into Random BST vs N



At small values of N, the average insertion time starts off very low and increases slowly, which makes sense because smaller trees require fewer comparisons to find the correct spot for insertion. As N grows, the time gradually increases which shows that larger trees have more levels to search through. The ups and downs throughout the curve do not match the theoretical behavior but it is expected since each randomly generated tree has a slightly different shape. Some random trees end up more balanced (faster insertions), while others are more skewed (slower insertions). The spike around N = 600 is probably caused by random variation or a few unusually unbalanced trees that required more comparisons than average. However, the trend still shows that the insertion time grows very slowly with N, which matches the expected O(log N) average-case complexity of insertion in a random BST.

Average Height of Random BST vs N



At small N, the average height rises quickly, since adding even a few nodes creates new levels in the tree. As N gets larger, the curve begins to flatten which shows that height increases more slowly. This pattern reflects the fact that random BSTs tend to be approximately balanced on average. The small fluctuations in the graph come from randomness in the structure of individual trees. Some happen to be taller or shorter than average but the overall trend is smooth and consistent. The shape of the curve matches the expected theoretical behavior. The average height of a random BST grows roughly proportional to O(log N), not linear to N. Doubling the number of nodes adds only a few more levels to the tree.