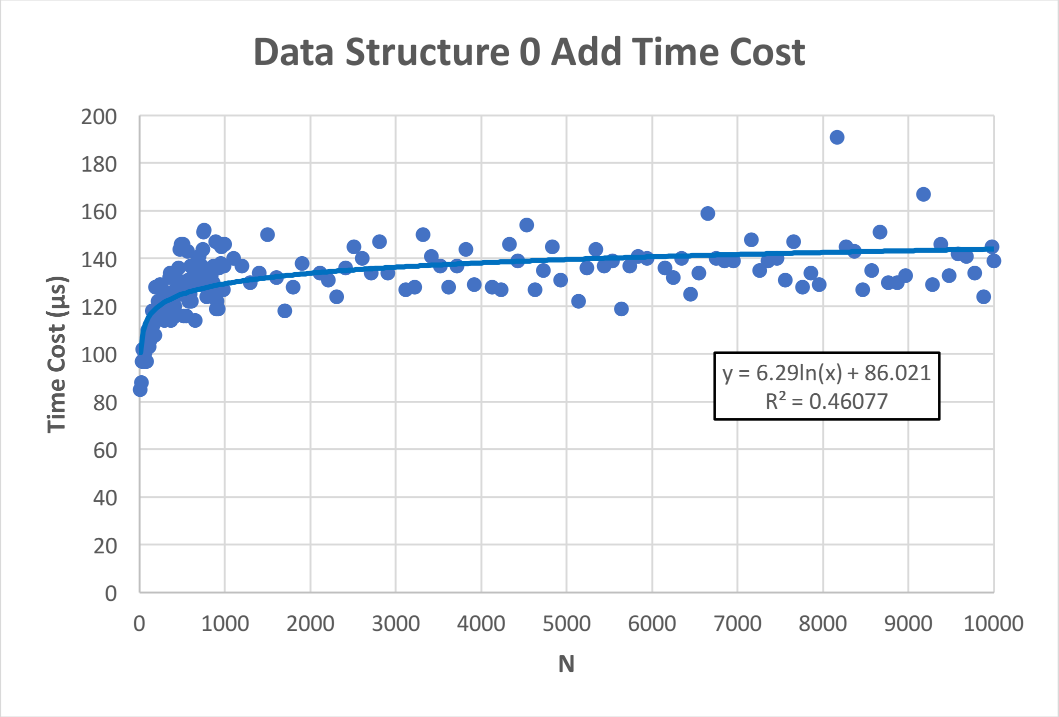
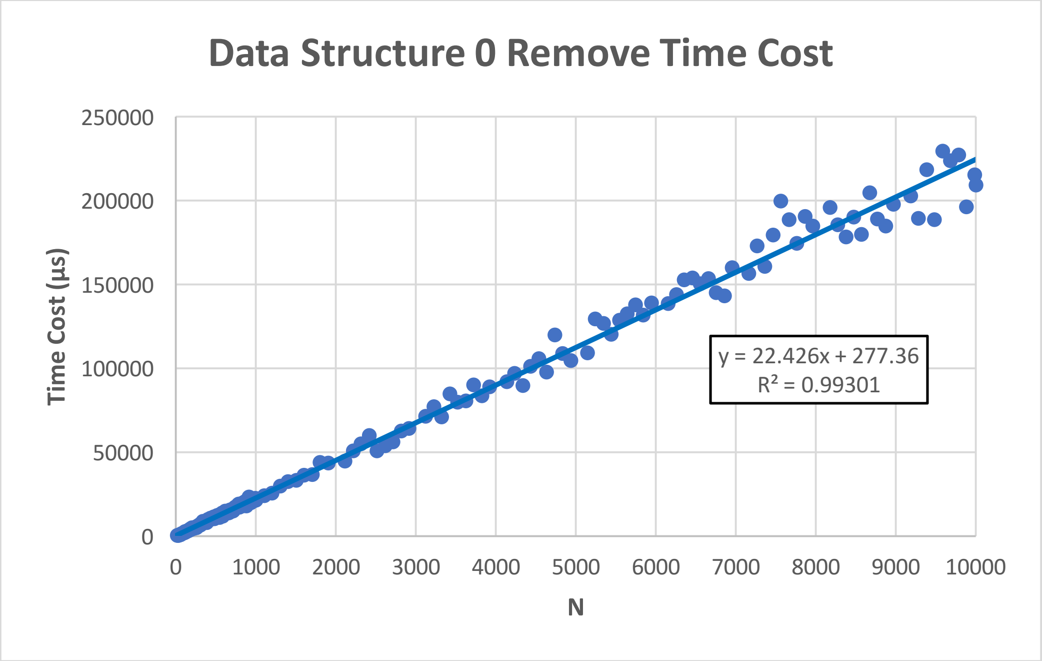
Nils Backe and Varnika Sinha
CS 210X
Accelerated Object Oriented Design Concepts
Project 4 Analysis
vsinha2 data structures

Mystery Data Structure 0

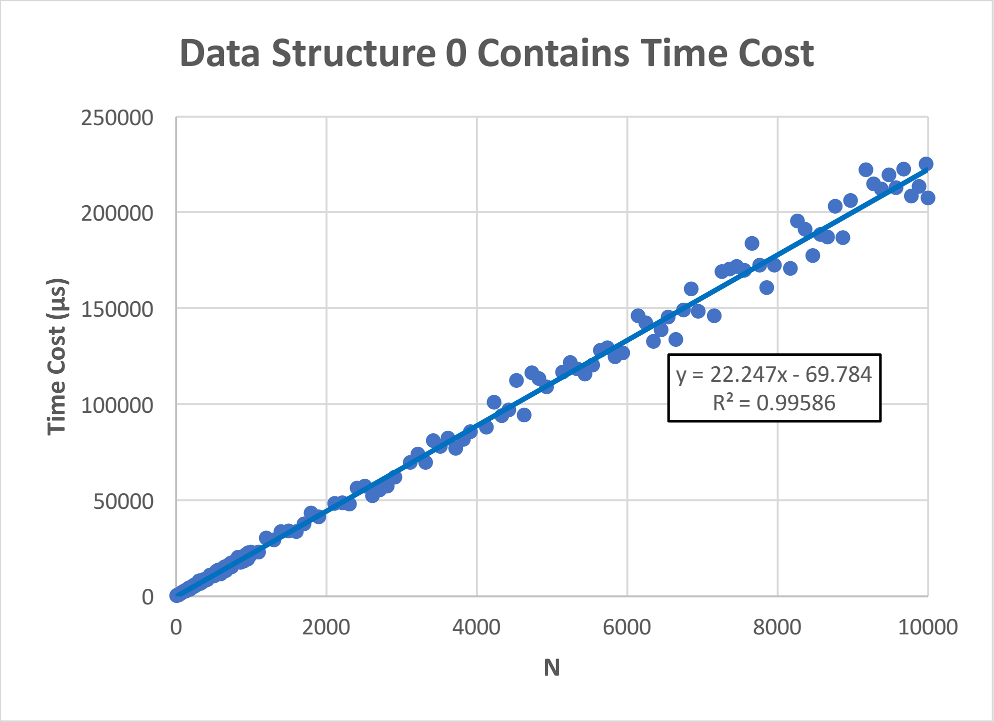
Add Method Time Cost:



Remove Method Time Cost:



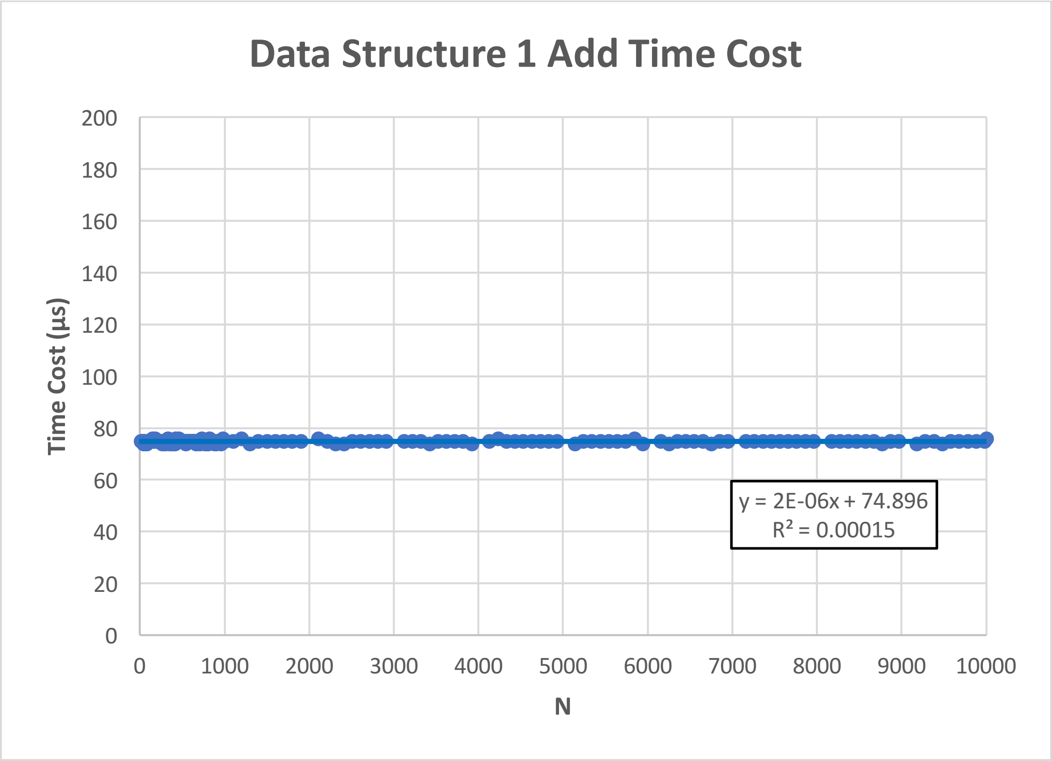
Contains Method Time Cost:



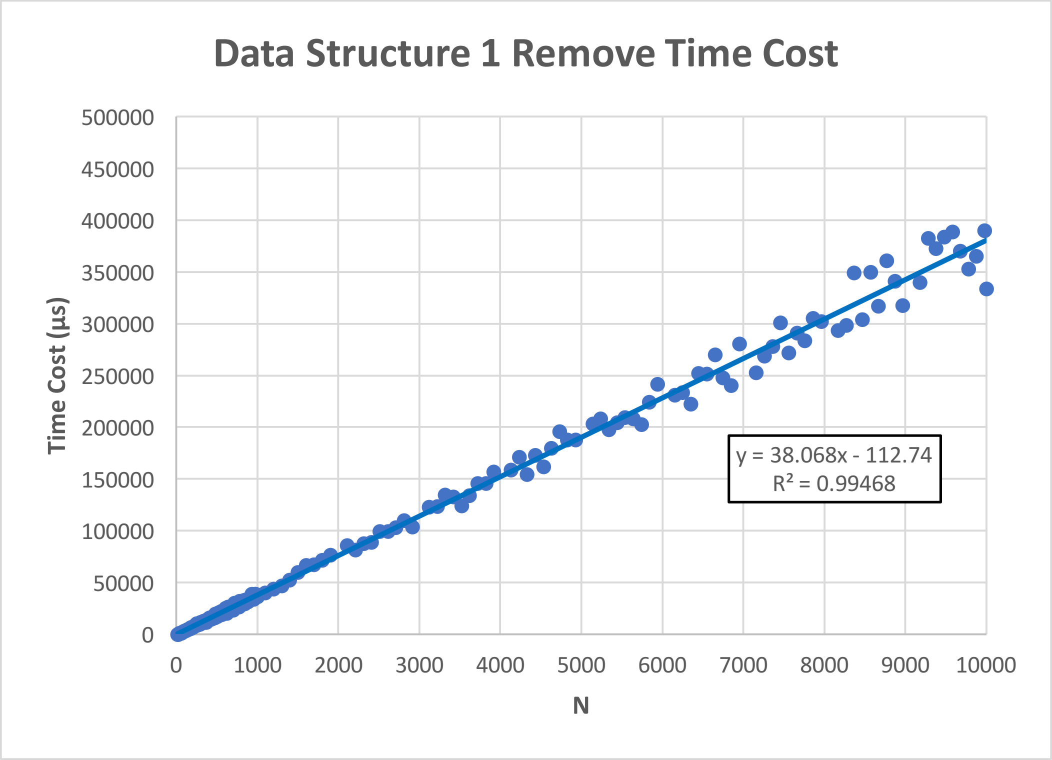
We have deemed data structure 0 to be a heap. The first graph shows a (rough) logarithmic shape, as the rate of change slows as N increases, which matches the O(log n) time cost of the add method for a heap. Furthermore, the second and third graphs, for remove and contains respectively, clearly show a linear shape, which again conforms to the time costs of a heap.

Mystery Data Structure 1

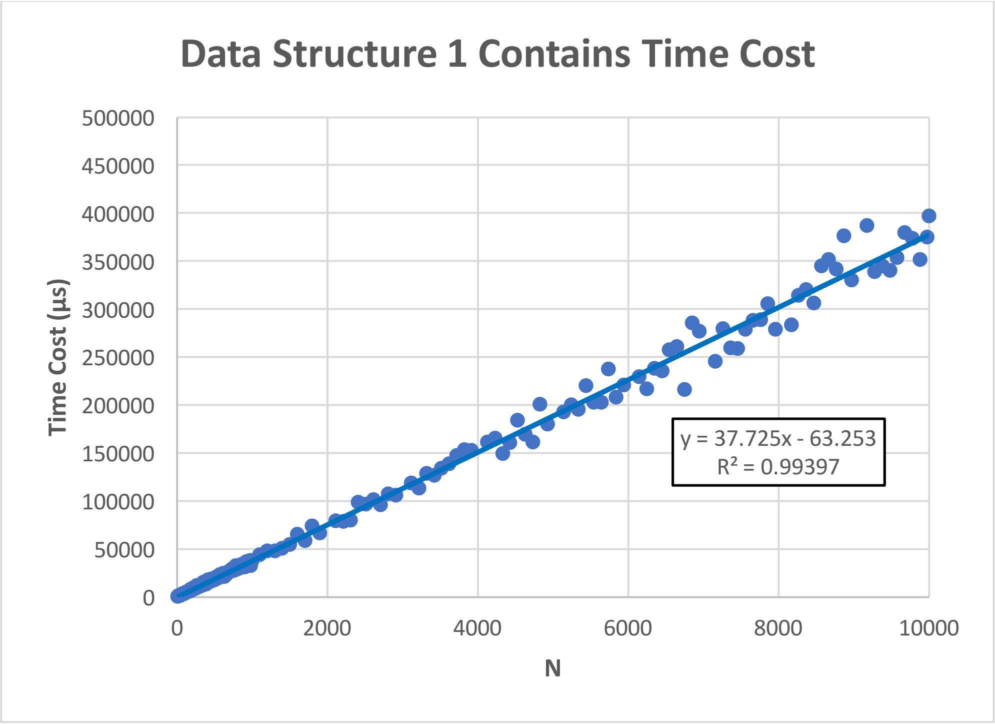
Add Method Time Cost:



Remove Method Time Cost:



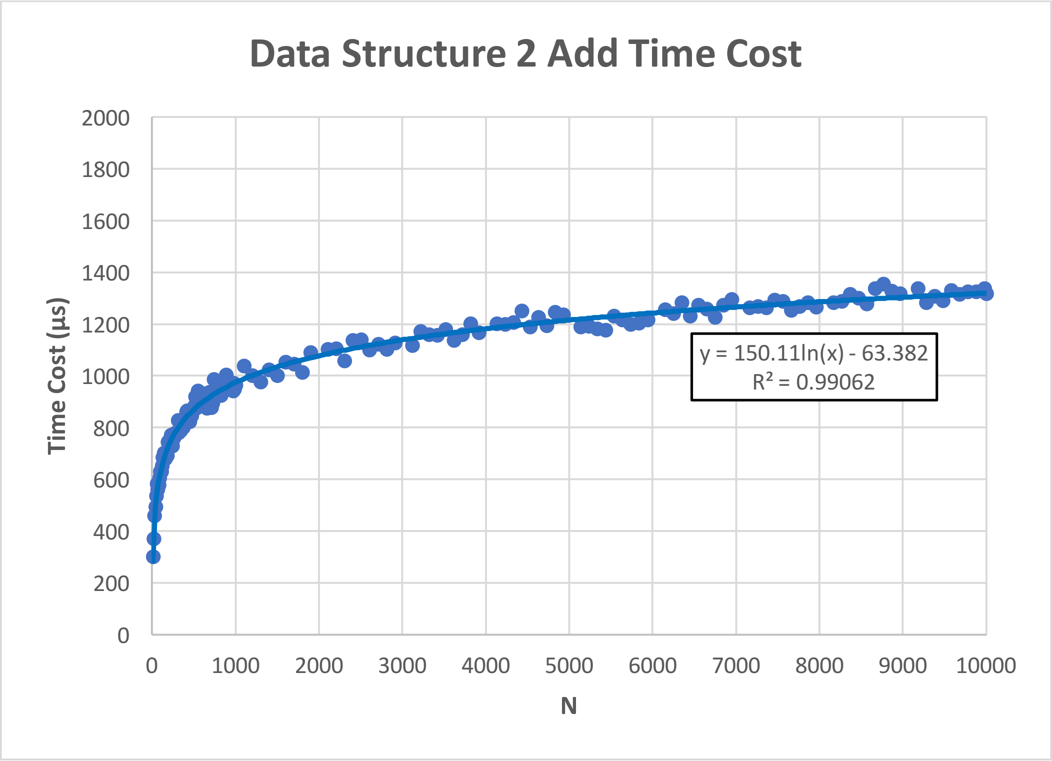
Contains Method Time Cost:



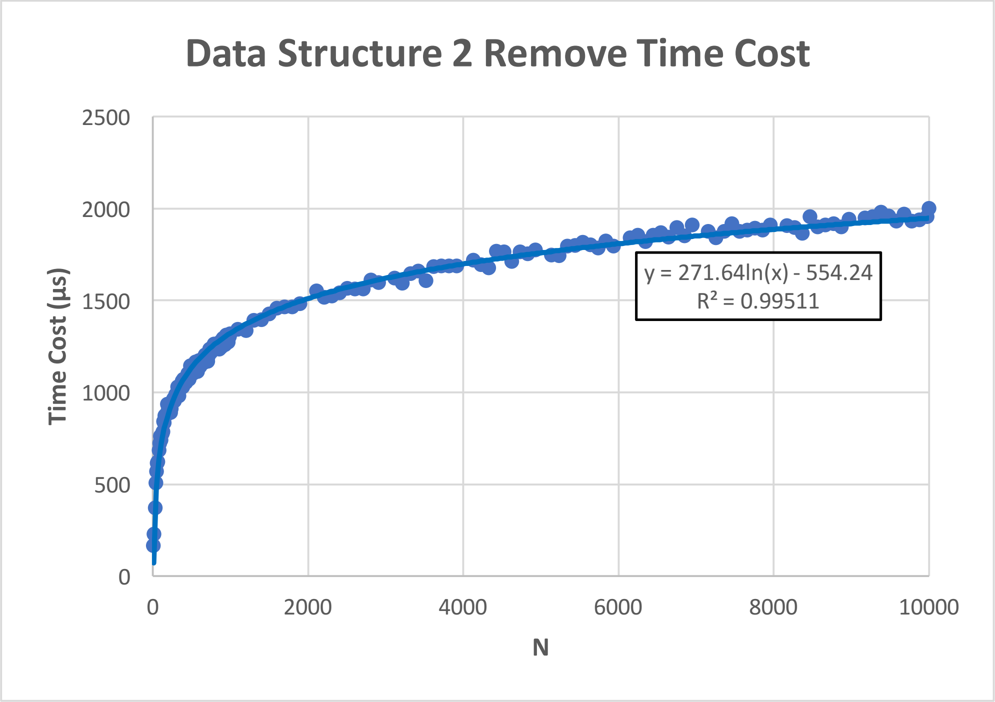
We have deemed data structure 1 to be a doubly linked list. The first graph clearly depicts a O(1) time cost, which matches the time cost of the add method for a doubly linked list. Furthermore, the second and third graphs, for remove and contains respectively, clearly show a linear shape, which again conforms to the time costs of a doubly linked list.

Mystery Data Structure 2

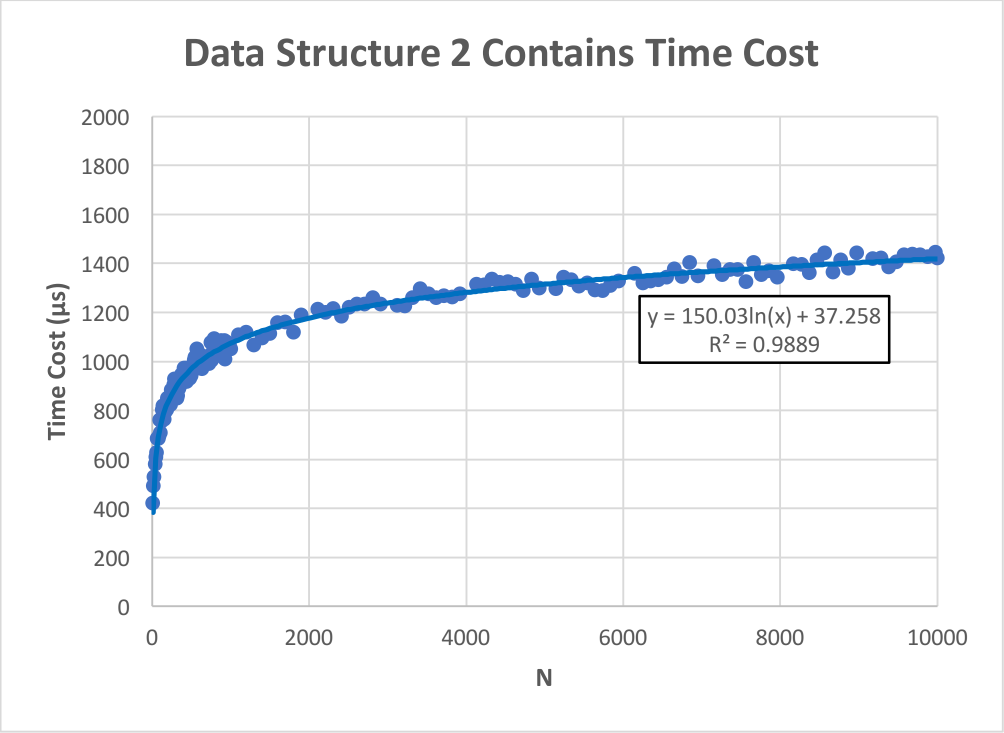
Add Method Time Cost:



Remove Method Time Cost:



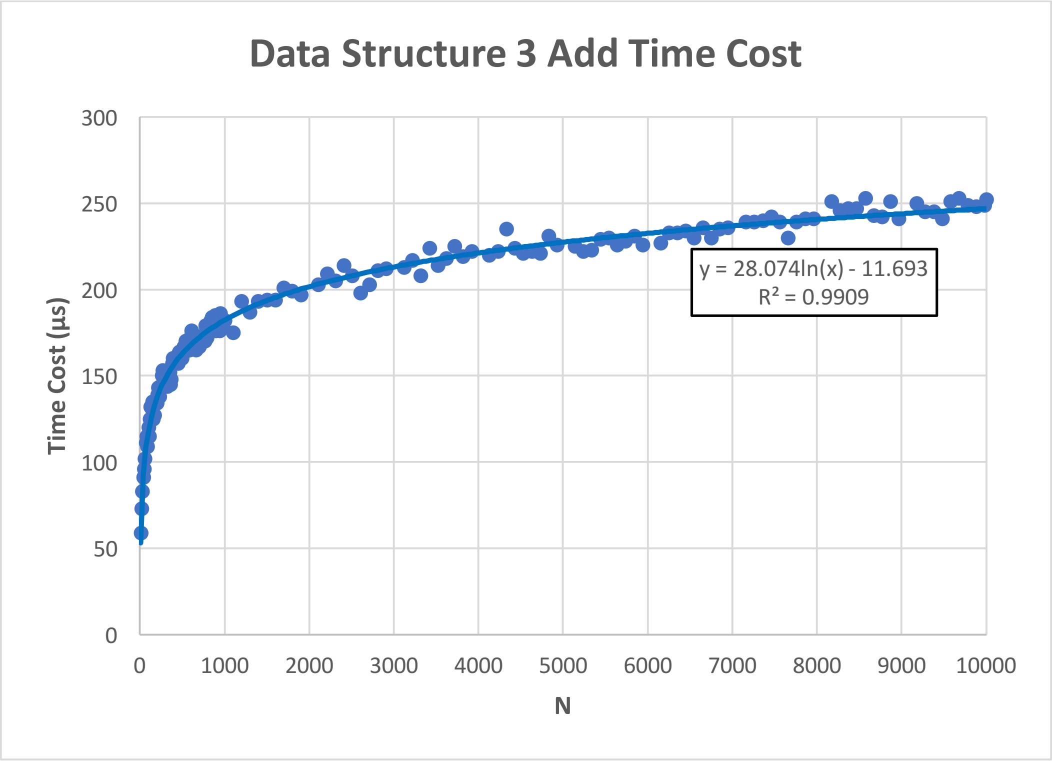
Contains Method Time Cost:



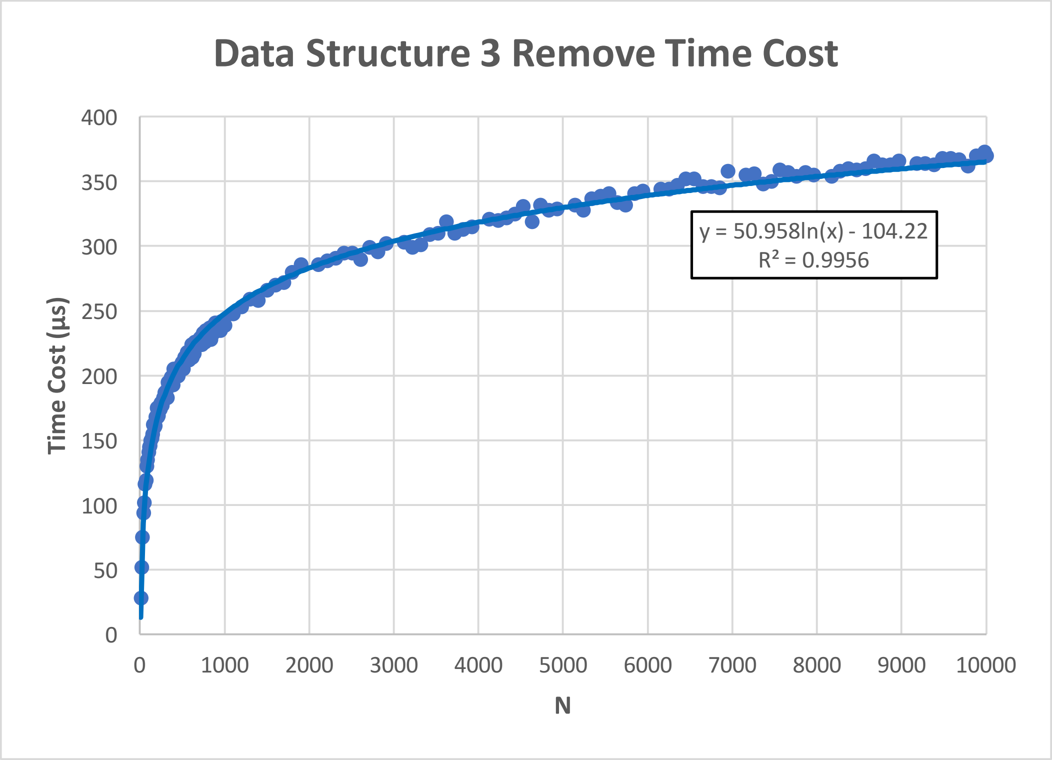
We have deemed data structure 2 to be a binary search tree. All three graphs show a logarithmic relationship between N and the asymptotic time cost, which matches the known time cost of a binary search tree for all 3 main methods: O(log n). In all three graphs, this self-balancing binary search tree shows a very accurate logarithmic curve with little variation.

Mystery Data Structure 3

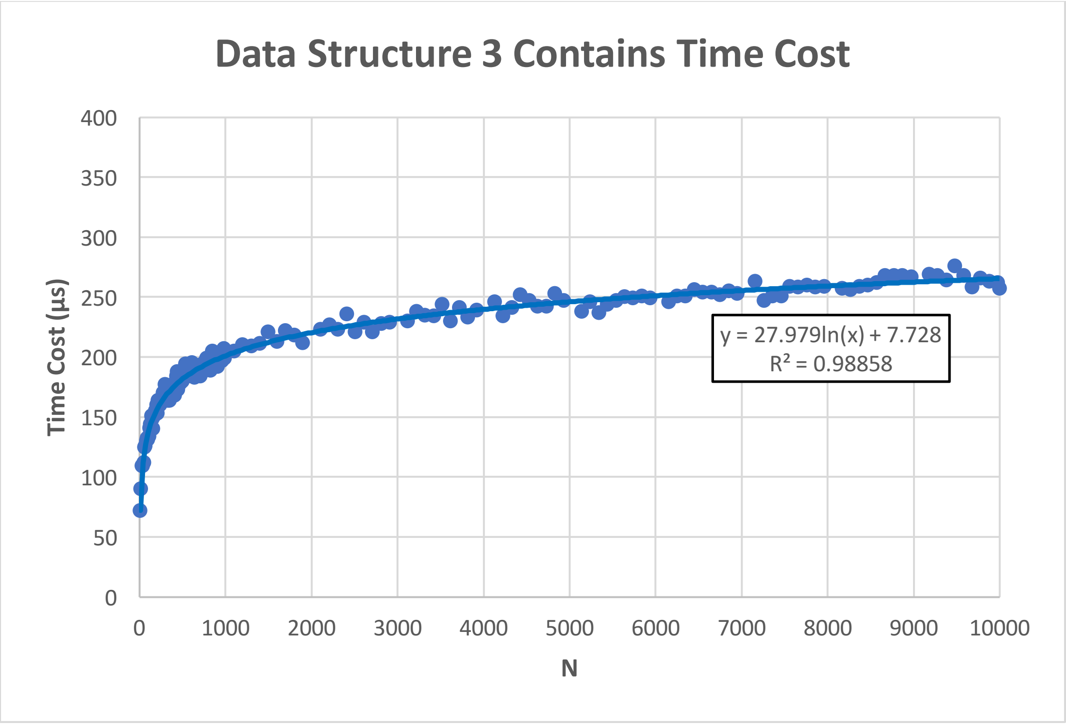
Add Method Time Cost:



Remove Method Time Cost:



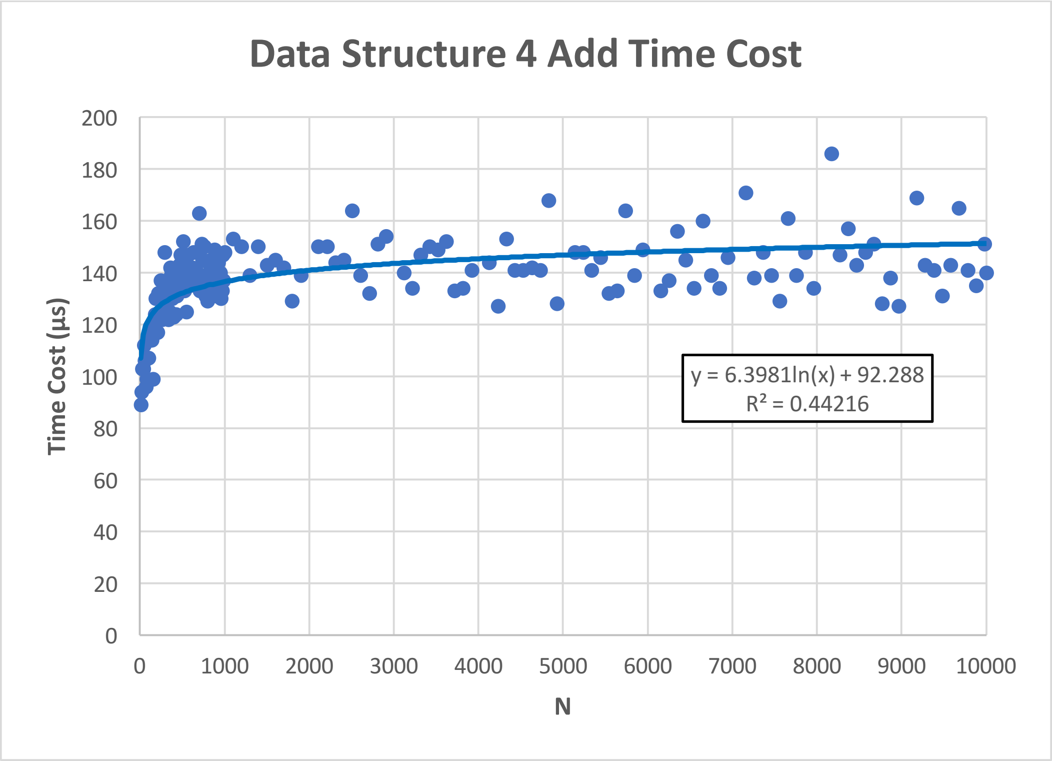
Contains Method Time Cost:



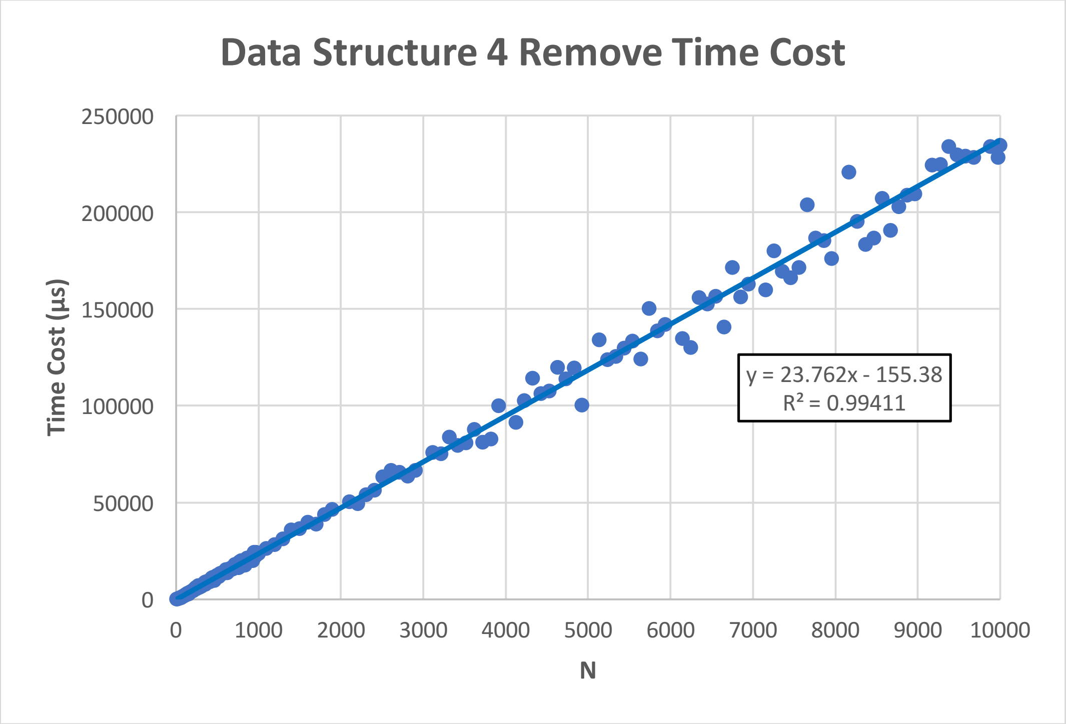
We have deemed data structure 3 to be a binary search tree. All three graphs show a logarithmic relationship between N and the asymptotic time cost, which matches the known time cost of a binary search tree for all 3 main methods: O(log n). In all three graphs, this self-balancing binary search tree shows a very accurate logarithmic curve with little variation as all three R^2 are greater than 0.98.

Mystery Data Structure 4

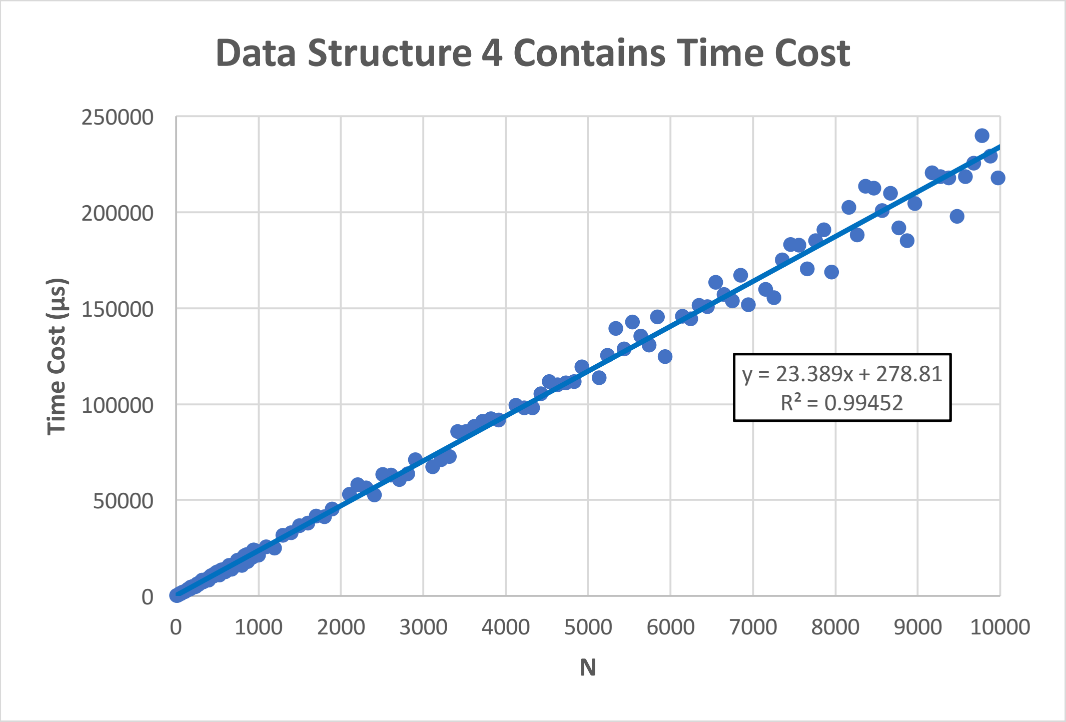
Add Method Time Cost:



Remove Method Time Cost:



Contains Method Time Cost:



We have deemed data structure 4 to be a heap, the same as data structure 0. The first graph shows a (rough) – again very similar to data structure 0 – logarithmic shape, as the rate of change slows as N increases, which matches the O(log n) time cost of the add method for a heap. Furthermore, the second and third graphs, for remove and contains respectively, clearly show a linear shape, which again conforms to the time costs of a heap. Because the R^2 value of the first graph is quite low (high variance) for both data structure 0 and this data structure, we have determined data structure 4 to be another occurrence of a heap.