The difference between One Way Binary Tree and Black Red Tree

In the previous and current labs, we have constructed two types of binary trees: One Way Binary Tree and Black Red Binary Tree. And in this lab, the performance is measured simply by searching each result tree 1000 times at random and time the performance.

In my specific case, the library used are the original library (sorted) and a randomized library. In each run, the corresponding library was imported and then searched for 1000 time. The time is later on reflected in the Terminal window in seconds for comparison.

The result of this run is that the Black Red Tree takes 0.0138 seconds while One Way Tree takes 0.1455 seconds to search after importing a sorted library and 0.0128 seconds while One Way Tree takes 0.00692 seconds to search after importing a randomized library.

When searching for an item in a sorted library, the Black Red Tree creates a balanced tree where from the root to each leave is balanced, while the One Way Binary Tree creates a one way tree where the root is the first item and the furthest leave is the last item. For searching purposes, it is more efficient for the Black Red Tree since each search does not start from the first data point and does not look through each data point until the desired one is found.

However, on the other hand, once the library is sufficiently randomized, each import is supposed to create a balanced tree for both methods. Depending on the complexity of the import, the performance of both trees should be quite similar to each other. In this specific test case, the One Way Tree perform better than Black Red Tree, which has almost no performance difference in between two different import methods. This result could be a case specific result of how the One Way Tree import its data when at random while the expectation is that both trees should perform relatively the same in stead of One Way Tree performance twice quicker than the Black Red Tree.

By comparing these two different import methods, it is obvious that when importing from a sorted library, the Black Red Tree performs better due to its ability of creating a balanced tree that reduce the access time for each search dramatically. As for importing the randomized tree, in this specific case, the performance of the One Way Tree performs better due to case specific data importing and case specific data searching.