**Problem 5 – Trees**

**Compare number of nodes to identify a match with the hash vs. the tree**

On average, hash uses way less nodes to find a match. Since with hashing it only looks at the specified index and the elements tied to that index. With a binary tree not every element is looked at but the tree does have to be traversed down its nodes to determine if an element is there. However, hash has a worst case complexity of O(n) since potentially all elements can be under one index, a binary tree has worse case complexity at O(log n) because not every element has to be traversed because you know at each level if the element is greater than or less than that node.

**How does the compare to Hashing?**

If each element has a unique index hashing is always the better case because with that worst case will always be O(1), but this not perfect hashing so the usefulness of each program depends on the number of unique indexes. The more elements with the same index the less useful hashing becomes.