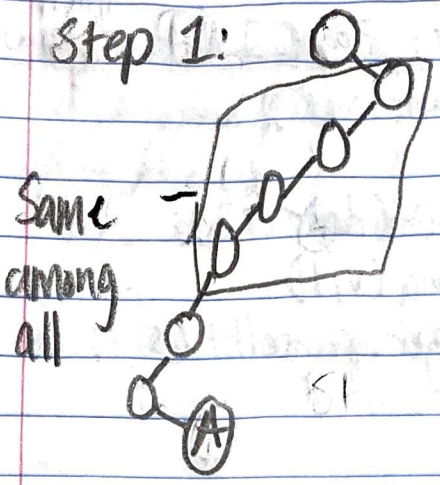
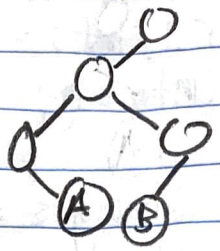
Module 4

Execise 4.9

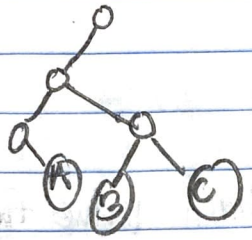
Step 1:



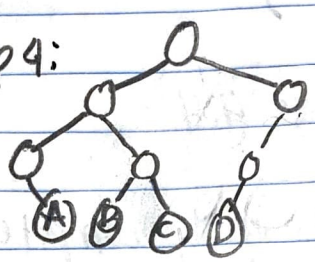
Step 2:



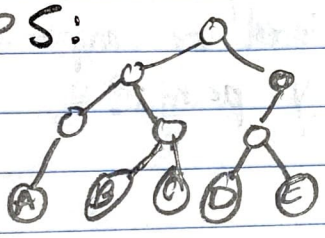
Step 3:



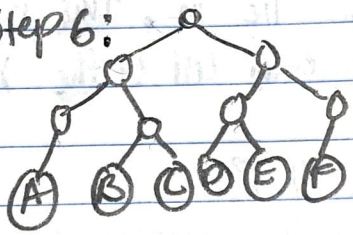
Step 4:



Step 5:



Step 6:



4.1

The second signature is better for uniqueness because two different items can't have the same binary order #. The first signature is better for brevity because it has one char as its signature. The first is better for efficiency for the same reason as above. And finally the second signature is better for order-preservation because it has specific keys that specify the order.

Exercise 4.3

Algorith: hashInts ()

```
1. int [] stats = new int [m]; float [] data = new float [m];
2. for (int i=0; i< stats.length; i++)
3. {
        String s = String.valueOf(data[i]);
        String ss = s.substring(0,1)
        int first_digit = Integer.parseInt(ss);
        stats[first_digit]++;
   }
```

Exercise 4.4

To efficiently compute the hash value for point at (x, y) we need to find which cell the points x value and y value lie in. This is computated by comparing the bounds within each cell to the x and y points.

Exercise 4.8

The full trie wastes O(n) storage because each of the numbers differentiate from one another so they have unique paths to them. These unique paths all directly end up at the leaf node to then compare the keys. So this leaves empty internal nodes as each insertion occurs in the full trie.

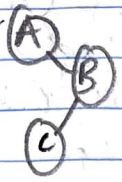
Exercise 4.5

The way I would design this tree would be with each node holding a point with a left pointer to other nodes with a shorter distance to the root point, and a right pointer to other nodes with a larger destance to the root.

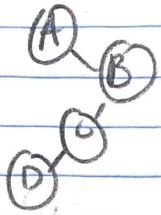
Exercise 4.6

Step 1: (A) Step 2: (A)—(B)

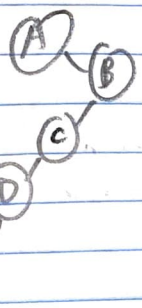
Step 3:



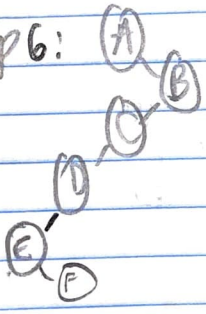
Step 4:



Step 5:



Step 6:



Exercise 4.7

This is true because with n bits there are 2^n possible keys. And $\log_2 n$ is the inverse of 2^n so we need $\log(n)$ bits to represent the keys.

We use reverse ASCII instead because it is more effient this is due to the fact that the bits at the end of each letter is more unique, and the changes are more frequent.