Practical 8: Tries

A trie is a tree-like data structure whose nodes store the letters of an alphabet. By structuring the nodes in a particular way, words and strings can be retrieved from the structure by traversing down a branch path of the tree.Tries can be extremely useful in searching for a word in a dictionary of words. It is important to note that a trie will only work if there are no words that are pre-fixes of an other word in the input set.

What am I doing today?

Today's practical focuses on:

1. Implementing a Trie by hand
2. Implementing a Trie in code

Trie Development

Part 1

Let's start by creating a Trie by hand.

Build a trie by hand:

Let's take an example set of strings: S = { bank, book, bar, bring, film, filter, simple, silt, silver}

So work out a trie by hand for this set of words.
You can consult the lecture videos to help you.

Steps:

1. Start at the root note which has no character associated with it
2. Begin with the first word in the set and add child nodes to the root until you reach the last character in the set - adding the leaf node
3. Work your way through all the words in the set adding characters as needed, taking advantage of shared prefixes where possible.

My handwritten Trie.