Lab 8 - Tries

Insertion

bank: First, we insert character b into the root with endOfWord value set to false. We give this a child node with character a that has a value of false. This then gets a child node n with value false and points to a leaf node k with a value of false that points to true. The word bank has been entered into the trie.

book: We go to the root node and see b is already there. We give the current child node the letter o, so it contains “a”, ”o” and value false. This then points to a new child node o with value false which points to a leaf node k of value false which points to true.

bar: We go to the root node and see it contains b. We then go to the child node and see it contains a. We create a new leaf node from this child node a with value of false and points to true.

bring: We go to the root node and see it contains b. we then go to the child node and add r to it, so it becomes “a”, ”o”, “r” and maintains value false. This node points to a new child node i with value false. This points to a new child node n with value false which points to a leaf node g with value false which points to true.

film: We go to the root node and insert letter f. We point to a child node i with value false. This then points to a child node l with value false which points to a leaf node m with value false that points to true.

filter: We go to the root node and see f is already there, so we go to the child node and see i is already there. We go to this child node and notice l is already there. From this child node we go to the next child node and add t and value remains false. This points to a new child node e with value false. This points to a leaf node r with value false which points to true.

simple: We go to the root and add s. This then points to the child node containing i and the value remains false. This points to a new child node m with value false. This points to a new child node p with value false. This then points to a new child node l with value false which points to a leaf e with value false which points to true.

silt: We go to the root and see s is already there, so we go to the child node and see i is already there. From this child node we point to a new child node l with value false. This then points to a leaf node t with value false which points to true.

silver: We go to the root and see s is already there, so we go to the child node and see i is already there. We also see l is contained in the next child node. This child node then points to a new child node v with value false. This points to a child node e with value false which points to a leaf node r with value false which points to true.

Search

For each character in the string we check if there is a child node that contains that character. If that character does not exist, we return false. If that character exists, we check if there is a child node that contains the next character. This repeats until the end of the string is reached. When the end of the string is reached, if the current node points to true, return true else, return false.