Indexer
Project 3
Daniel Benedicto and Sarah Bizzarro

Design

The main data structure used in this assignment is a trie, which stores the words from all of our text file(s) as character nodes in a tree. The end of each word then points to its own linked list that stores every file it's in and how many times it shows up. This is implemented by defining a TrieNode struct and an OccurrenceNode struct, where TrieNode contains a reference to a node's next sibling and its first child (so a node's children are essentially stored as a linked list) and OccurrenceNode stores the name of the file the word is in and its occurrences in that file. The main function maintains a reference to the root TrieNode of the trie. Since the program is reading in each word from each of the files using fscanf, the trie is being built character-by-character while keeping each TrieNode in alphabetical order and each OccurrenceNode in numeric order, gives this program a runtime of O(nm) and space complexity O(m).