# Prefix and Suffix Search

Approach #2: Paired Trie [Accepted]

**Intuition and Algorithm**

Say we are inserting the word apple. We could insert ('a', 'e'), ('p', 'l'), ('p', 'p'), ('l', 'p'), ('e', 'a') into our trie. Then, if we had equal length queries like prefix = "ap", suffix = "le", we could find the node trie['a', 'e']['p', 'l'] in our trie. This seems promising.

What about queries that aren't equal? We should just insert them like normal. For example, to capture a case like prefix = "app", suffix = "e", we could create nodes trie['a', 'e']['p', None]['p', None].

After inserting these pairs into our trie, our searches are straightforward.