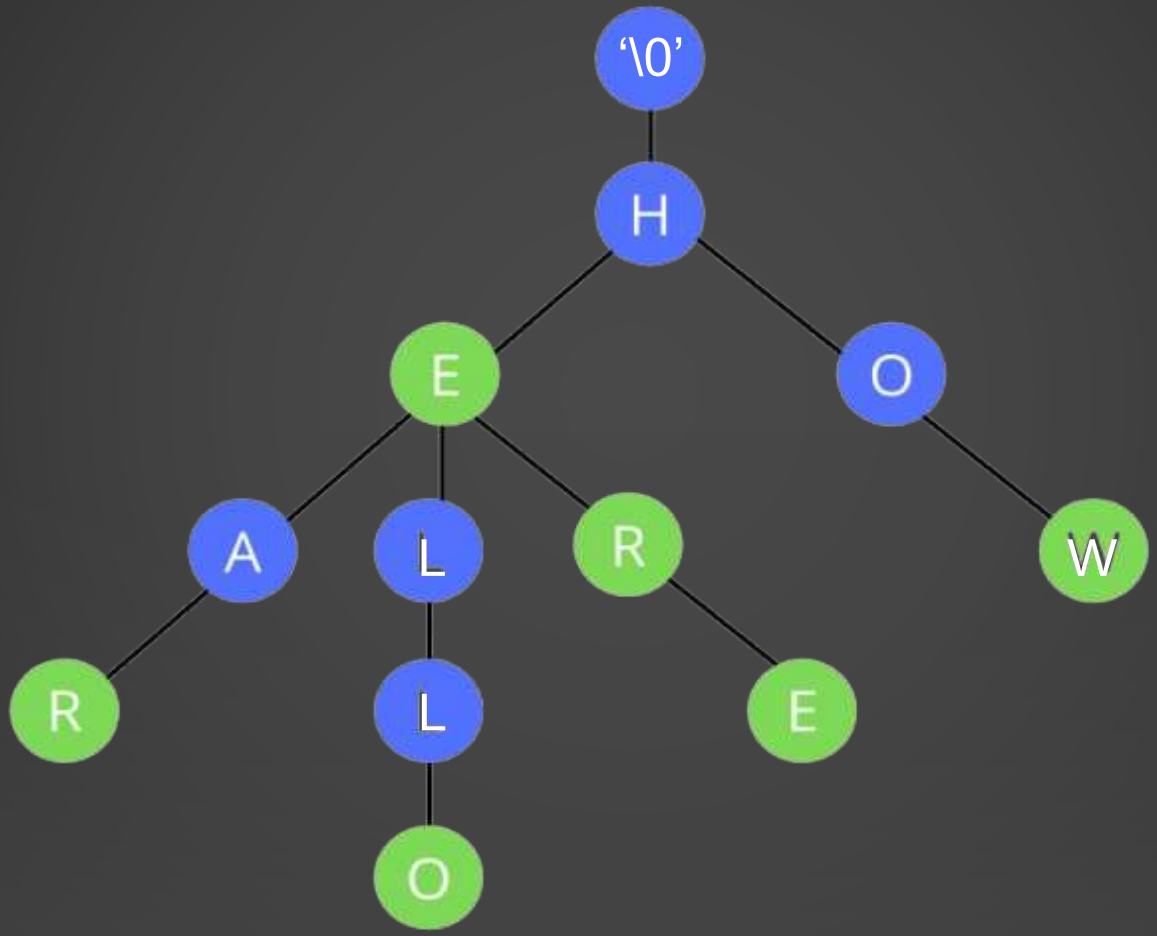
TRIE – THE MAGIC TREE



WHAT IS TRIE ?

- Trie is a type of Search Tree , a Tree data structure whose nodes stores letters of alphabet.
- By using Trie searching complexities can be reduced as Tries are used for locating specific keys from within a set in less time.
- Trie can be used to implement dictionary. There are numerous other applications of Tries as well.

APPLICATIONS

- Auto-Complete in Search Engines.
- Pattern matching Algorithms.
- Text Search.
- Browser History.
- Spell Checkers or Auto-Correct.

ABOUT IDEA

- The main source of Inspiration was Search Engines.
- There was keenness to know how Auto-Complete , Pattern matching and Spell Checkers on search engines actually work.



TRIES  HASHMAPS

- Tries are more memory efficient than Hashmaps as they store only a single Node , for a particular character of many words.
- Auto-Complete and Spell Checkers are some amazing features of Tries which can't be implemented in Hashmaps in less runtime complexities.
- Tries can provide an alphabetical ordering of enteries by key in less runtime complexity than Hashmaps.

BASIC OPERATIONS IMPLEMENTED

- Insert a word with its meaning.
- Search a word.
- Get all words with their meaning.
- Find meaning of a word.
- Delete a word.
- Auto – Correct implementation.
- Auto – Complete implementation.

Thank You.

Let's head towards demo now