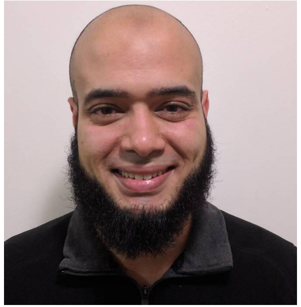
Data Structures

Trie Homework 2

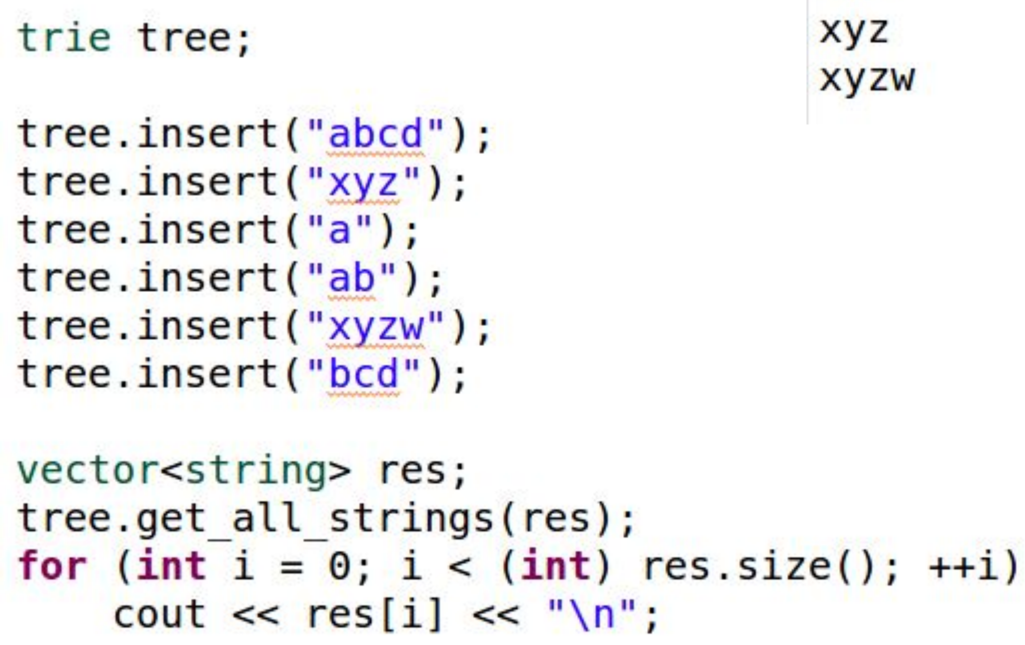
Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

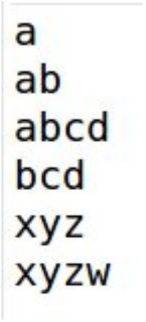


Problem #1: Listing tree content

- void get_all_strings (vector<string> &res)
- The function returns all the current strings in the trie sorted
  - You will need to recursively search the tree for leaf nodes



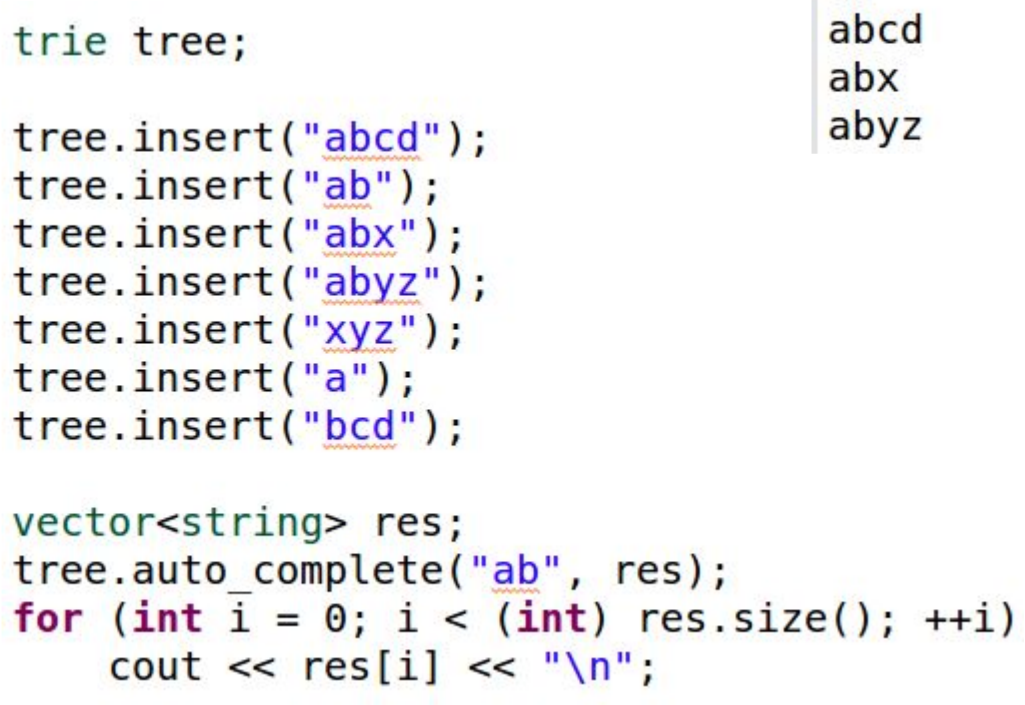
```
trie tree;

tree.insert("abcd");
tree.insert("xyz");
tree.insert("a");
tree.insert("ab");
tree.insert("xyzw");
tree.insert("bcd");

vector<string> res;
tree.get_all_strings(res);
for (int i = 0; i < (int) res.size(); ++i)
    cout << res[i] << "\n";
```



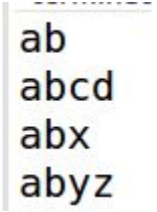
```
a
ab
abcd
bcd
xyz
xyzw
```

Problem #2: Autocomplete

- void auto_complete(const string &str, vector<string> &res)
- Given a string, the function will make an auto-complete for it. In other words, return all full words that are in trie and has this prefix



```
trie tree;

tree.insert("abcd");
tree.insert("ab");
tree.insert("abx");
tree.insert("abyz");
tree.insert("xyz");
tree.insert("a");
tree.insert("bcd");

vector<string> res;
tree.auto_complete("ab", res);
for (int i = 0; i < (int) res.size(); ++i)
    cout << res[i] << "\n";
```



```
ab
abcd
abx
abyz
```

Problem #3: Word with exactly single letter change

- bool word_exist_with_1_change(string str)
- Returns true if you can change exactly one character in str to match any string in the trie words, otherwise returns false.
- Assume trie is: "hello", "leetcode"
- Input ⇒ output
    - hello ⇒ False   [0 changes]
    - xello ⇒ True   [1 change x to h]
    - xyllo ⇒ False   [2 change to match the trie words]
- *Tip: think brute force*

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”