**DATA STRUCTURES – FALL 2023**

**LAB 12**



# **Lab Task: AVL Tree for Dictionary Autocomplete**

**Objective:** Implement an AVL tree to efficiently perform autocomplete suggestions based on a dictionary of words. Enhance the program to handle duplicates and allow deletion of nodes.

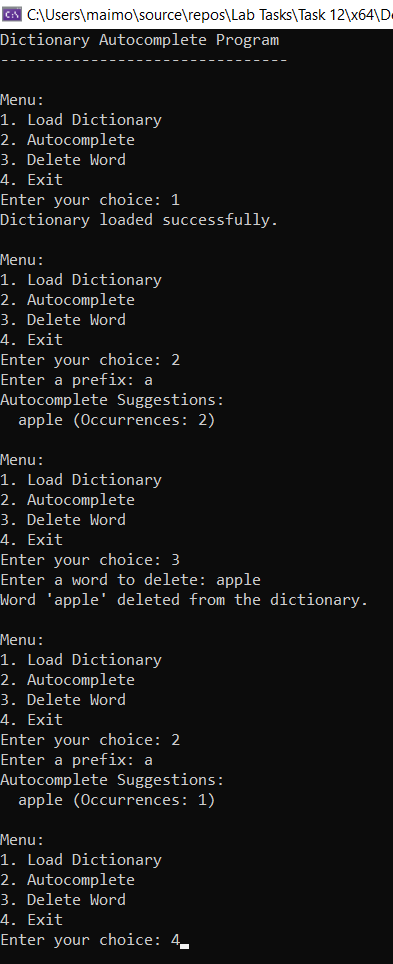
**Task Description:**

1. **AVL Tree Implementation:**
   * Implement a basic AVL tree with insertion, deletion, and search operations.
   * Modify the insertion operation to handle duplicate words. If a word already exists in the dictionary, update its occurrence count.
2. **Dictionary Loading:**
   * Write a function to load a dictionary of words from a text file given “dictionary.txt” into the AVL tree. Each word will be a node in the AVL tree. Handle duplicates appropriately.
3. **Autocomplete Function:**
   * Implement a function that takes a prefix as input and returns a list of words from the dictionary that starts with that prefix. This function should efficiently use the AVL tree to find matching words.
4. **User Interface:**
   * Create a simple console-based user interface that allows users to:
     + Load a dictionary file into the AVL tree.
     + Enter a prefix to get autocomplete suggestions.
     + Display the autocomplete suggestions.
     + Perform deletion of a word from the dictionary.
     + Exit the program
5. **Handling Duplicates:**
   * Modify the AVL tree to handle duplicate words by updating their occurrence count. Display the occurrence count along with the word when providing autocomplete suggestions.
6. **Deletion Operation:**
   * Implement a function to delete a specified word from the AVL tree. Ensure that the tree remains balanced after the deletion.

**Tips:**

* Handle duplicates by adding an occurrence count to each AVL node.
* Implement the deletion operation carefully to maintain the AVL tree's balance.
* Test the program with dictionary files containing duplicate words and verify that the autocomplete suggestions and deletion operations work correctly.

**Sample Output:**



A website has been created to help you visualize AVL Tree algorithms. You can find it here <https://www.cs.usfca.edu/~galles/visualization/AVLtree.html>