BIRZEIT UNIVERSITY

Department of Computer Science

COMP2421 - Data Structures and Algorithms (Spring 2023/2024)

Project#3 Due Date: May 25th, 2024 (by 10:00 PM)

In this project, you will implement an application of AVL and Hash table. The application will efficiently index words found in a text file. The application will use an AVL tree to maintain a balanced order of words for fast retrieval and sorting. Additionally, the words will be stored in a hash table for fast access. The project has the following features:

1. **Input Data:** Text files containing text data (the file may contain more than one line).
2. **Reading Data**: read words from a text file (input.txt), consider ignoring all input except the alphabetical characters. "Hello" and "hello" should be considered the same word). As you know, the word may consist from one letter only.
3. **Building the AVL Tree:** As words are read from the file, insert each word into the AVL tree with frequency equals to 1, if the word already exists, update its frequency (counter).
4. **Insert a word to the AVL tree:** insert a new word or update the frequency of existing word.
5. **Delete word from the AVL Tree:** Search for a word in the AVL tree and delete it. If the word does not exist, then show a clear message about this.
6. **Display words:** print the words and their frequencies using in-order traversal.
7. **Hash Table Creation:** After building the AVL tree, traverse the tree and insert each word and its frequency into a hash table. The hash table will allow for fast retrieval of word frequency by direct access through the hash of the word.
8. **Collision handling:** implement collision handling in the hash table (feel free to use any method you learned in this course)
9. **Search**: Allow users to search for a word and retrieve the frequency of that word directly from the hash table.
10. **Insert a Word to the Hash Table:** insert a new word in the hash table or update the frequency of existing wors.
11. **Delete word from the Hash Table:** Search for a word in the hash table and delete it. If the word does not exist, then show a clear message about this.
12. **Word Count Statistics**: Provide statistics such as total number of unique words, most frequent word, and words repeated more than specific threshold in the tree from the hash table.

Your application must have the following menu:

1. Load data from the file.
2. Create the AVL tree.
3. Insert a word to the AVL tree.
4. Delete a word from the AVL tree.
5. Print the words as sorted in the AVL tree.
6. Create the Hash Table.
7. Insert a word to the Hash table.
8. Delete a word from the hash table.
9. Search for a word in the hash table and print its frequency.
10. Print words statistics.
11. Exit the application.

The deadline for this assignment will be May 25th, 2024 (by 10:00 PM). LATE SUBMISSIONS will not be accepted for any reason. Before the discussion, please ensure your application runs properly on your laptop. Project discussions will be decided later.

Grading policy:

1. Your application should have all functionalities working properly.
2. Your application should contain a menu to allow the user to select which option (s) he would like to run.
3. Properly handling file, I/O errors, invalid user inputs, and edge cases in text editing operations.

Notes and submission instructions:

1. **This is individual work**. It should represent your efforts. It is fine to discuss your work and ask your colleagues, but you are not allowed to copy/paste part of the work of others or give it to anyone else. You are not allowed to post/copy from other websites and/or social media, which will be considered cheating.
2. Any **plagiarized** code will not be marked, resulting in a **zero** grade.
3. You are responsible for the submitted code.
4. **Document format**. Please submit only the code file (**c** file) containing your project's code. Please rename it as follows: "**P3_YourStudentID_FirstNameLastName_SectionNo.c**".
5. **Input file name**. Ensure the input file name is the same as in the specifications.
6. Include your full name, student ID, and section number at the beginning of your file.
7. Please do not compress the file; only the C-file is needed.

Good luck!