Lab 4
Binary Search Trees

1 Problem Statement

In this assignment, you are required to implement a spell checking system based on a binary search tree. You will be given a file containing all language words. The file would contain one word per line. At the beginning, print the height and size of the generated Binary Search Tree. Your program should prompt the user to enter a sentence to check its spelling.
For each word in the entered sentence, you check whether the word exists in the binary search tree.

(a) If the word exists, you will print that the word is correct.
(b) If the word does not exist, you will print three suggestions for the correct word:
- The word in the leaf node you reached before declaring that the word does not exist.
- The word in the inorder predecessor of that leaf node.
- The word in the inorder successor of that leaf node.

2 Definitions

- In Binary Search Tree, the inorder successor of an input node is defined as the node with the smallest key greater than the key of the input node.
- Similarly, the inorder predecessor of an input node is defined as the node with the largest key smaller than the key of the input node.
- For further explanataion you can see this video.

3 Application

- You will be given passed the path of the file containing the language words as a command line parameter.
- Use the `strcmp` C function to compare strings while building the Binary Search Tree or while lookinp up a word.

4 Example

Height of Tree = 43
Size of Tree = 98915
Input Sentence: “I lovd data structurew”
Output:

- I - CORRECT
- lovd - lovable, love, louvres
- data - CORRECT
- structurew - structures, structuring, structured

5 Deliverables

- Make sure you follow the course academic integrity policy.
- Your source code (in one C file), which should be well commented.

6 How to Submit?

- Your filename should be renamed to be your student id. If your id is 9876, then rename the file to 9876-bst.c
- Upload your deliverables to **https://forms.gle/k3D5KW1eYKUym3U97**