PA - 3
Expression Trees

Description:

For this programming assignment, you will implement expression trees. You will making be a simple expression tree creation and evaluation program. You can review the concept of creating and evaluating expression trees from lecture 6 slides of the DSA course.

While implementing this program you will have to answer certain questions and take certain judgement calls to define the process of creating the expression tree as well as evaluating it. Some of those questions could be:

- Do you want to make use of stacks while creating the expression tree or directly parse the expression to form the tree?
- Do you want to directly parse the input expression (infix) or convert it to a postfix form before creating the expression tree?

Make sure to document the answers to these questions and trace the steps taken during the implementation for a better understanding of the workings of your program.

Skeleton code is provided. Please do not change function names or class names. Additional classes and functions can also be added.

Coding Portion (20 Points):

- Create the implementations of the provided functions. Be sure to test the correctness of your program.
- If you choose to use stacks or functions for the infix to postfix conversion from previous lab exercises and assignments, make sure to include them all in an additional helper file named **helper.py** Submit the **PA3.py** file and additional helper files you create (if any).
- Your code will be graded based on whether or not it runs, produces correct output on test inputs, and the documentation you include about the design choices for your program.
- Required functionality for provided functions:
    - `build_expression_tree(expression)` – takes a string infix expression as the input and creates an expression tree returning the root node of the tree as the output. If you use stacks or infix to postfix conversion, make sure to include all those function calls within this function itself.
    - `evaluate_expression_tree(root)` – takes the root of the expression tree as the input and returns the result value as the output. The output should be an integer value.

Submission:

Once you have completed the assignment, you should upload the python script to the PA-3 codePost portal (will be visible soon). Please ensure the following while submitting:

- Once satisfied with your code, you should download the file as a python script (.py file), by going to **File > Download > Download .py**
- The name of the file should be PA3.py
- Upload the python script file to codePost under the PA-3 assignment.
- You can run the test cases on your script up to a limit of 50 times.
- Once satisfied with the test runs, complete your submission.