**Task 01**

You are tasked with implementing a C++ function that evaluates an arithmetic expression in postfix notation using a stack. The function should take a string as input, which represents the postfix expression, and return the result of the evaluation.

The postfix expression will consist of one or more operands and operators, separated by spaces. The supported operators are addition (+), subtraction (-), multiplication (\*), and division (/). The operands will be integers, but may be negative.

Your implementation should follow these guidelines:

1. Your function should use a stack to evaluate the expression.
2. You should handle errors gracefully. If the input expression is not a valid postfix expression, your function should return an error message.
3. You should support negative operands and results.
4. Your function should be case-insensitive.
5. You should handle division by zero by returning an error message.

Note: You may use the built-in stack library.

**Task 02**

Implement a recursive function to count the number of nodes in a binary tree. The function should take a pointer to the root node of the tree as a parameter and return the total number of nodes in the tree.

You can assume that the nodes in the tree are represented by a struct that looks like this:

struct Node {

int data;

Node\* left;

Node\* right;

};

The data member of each node contains the integer value stored at that node, and the left and right pointers point to the left and right child nodes, respectively. If a node has no left or right child, its left or right pointer should be nullptr.