Project 3 Calculator

Calculator:

For this project, you will design and implement a simple integer calculator, which converts infix expressions to postfix expressions and evaluates the postfix expressions. Graphical interface is not required. Calculator input will be provided from the keyboard and feedback from the calculator application will be displayed on the screen.

Inputted infix expressions will contain only operands, operators and parenthesis, where

- An operand is a sequence of one or more decimal digits, optionally preceded by a plus or minus sign, and containing no embedded spaces or tabs.
- Operators include the characters +, -, *, /, (and).

Your design should have some fault tolerance. For example in the input expression there could be some illegal numerical operations (division by zero) or unbalanced symbols. Your program must deal with these situations without crashing. If an error occurs, your program should print out an error message, flush the current expression and proceed to the next input command.

Data structures:

The primary data structures element of this project is the stack used to convert and parse the postfix expressions. The implementation of the stack is under the following specific requirements:

- You must use a well encapsulated stack
- The stack should be implemented with a linked list and the memory space took by the stack must be allocated dynamically.
- The stack must be protected against overflow and underflow. When such an error occurs, the users should be notified.