Given project is a simple calculator that performs operations on integers that are pushed and popped using stack.

The mail file CALC.c makes use of the custom header file CALC.h which defines the possible tokens of the input string, the structure of integer list , stack and the function declaration statements for the push and pop operations of stack.

The individual files are tested and debugged using breakpoints . Initially to start with the header file, CALC.h , the syntax and the flow of declarations are validated.

* In the READTOKN.c file, the nextchar() function is tested with a random input eg: “2 18 + = 5 / =” . The function returns the next char where the index of stack is initialized to zero. The base condition is checked and when the index exceeds the length of the input values , zero is returned.
* The read\_token function returns the type of token (as mentioned in the header file CALC.h) . Additional conditional is added to check if the token is invalid as T\_INVALID in case of special character and alphabet inputs. The integer condition is checked as well (values between 1 and 9 are considered valid and assigned the token T\_INTEGER)
* The PUSHPOP.c file includes the primary operations performed on the stack (ie) push and pop. The push function is used push a integer node link to the stack and update the top . The pop function returns the top element and decrements the value of top.
* The primary file CALC.C is included with the getinput function which receives the input data in a character array. This acts as a buffer array.
* The condition for T\_INVALID is checked and the evaluation is terminated in such a case.
* Finally, the evaluated value of the postfix expression is displayed.