This program is a simple calculator that reads its input from a character buffer. If integers are read, they are pushed on a stack. If one of the operators (+ - \* ⁄) is read, the top two elements are popped off the stack, the operation is performed on them, and the result is pushed on the stack. The = operator writes out the value of the top element of the stack to a buffer.

***In CALC.c***

1. We need to include the PUSHPOP.c and READTOKN.c header files in CALC.c

2. We Changed the IntStack stack = { 0 }; from global to main () function Because it is useless to declare globally.

3. sprintf(char \*str, const char \*string,...); function just to store the string in to the str ,But we need to print it So We use a print statement in 48th line.

***In CALC.h***

1. In the enum datatype we are going typedef there no need to use enum name so just removed that
2. Other then that the CALC.h is perfect

***In PUSHPOP.c***

1. Here First we included the CALC.h header file in PUSHPOP.c
2. In the push() user defined function, instead of hypen there is some wrong usage of character is there So, We cleared it to access the structure members using structure point(ptr,stk).
3. And in the pop() user defined function also the same mistake in happened, So we correct it by misspelled character to hypen(-).
4. For more information we made a lot of comments in that program.

***In READTOKN.c***

1. In 41th line ,for loop as same mistake with semicolon ,with curly braces and even if the character (c) is a space also the for loop is executed So We changed it by using “!” not = .
2. And in 24th line in char \*buf\_in=”2 18 + = 5 / = ” instead of (/) division symbol some wrong character is there ,we changed it by /.

And those are the things We modified the programs . For more information we added small comments in the program.