Lab report

To begin, I had to acknowledge several cases when converting infix to postfix, which would result in a nested if-else. I had to keep track of operator precedence, operator order, and open/closed parentheses.

I created a stack called opstack (Operator Stack) that will store the operators as they come in. I traverse through the passed-in infix string once using a for loop, using int x as an index.

The top of the nested if-else determines whether the incoming char is an operator or not, because this is the easiest case. Non operators will simply be concatenated into the postfix string. Next I checked for open and closed parentheses, where when a closed parenthesis is entered it will pop the stack up to the next open parenthesis, but at the same time not include the parentheses themselves into the postfix string. I then checked for operators, but in order of precedence: multiplication and division to addition and subtraction. Within each of these I made if statements to check special cases where the incoming operator is of equal or lower precedence than that of the top of the stack.

When the for loop finished reading the infix string I emptied the stack and concatenated the rest of the remaining operators to the end of the postfix string and returned it back to main.