Experiment 3. Stack Application

I. Experiment purpose

For this lab, you will practice the appliaction of stack by converting the infix expression into postfix form. You should use stack to finish this task and show all of the steps.

II. Use instruments and materials

Computer

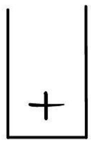
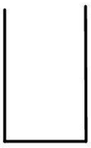
III. Experiment content

Convert the infix expression a –(b + c)/d + e into postfix form. You must show actions/operations and the status of the stack after each step of the algorithm in the table. The following algorithm is one we discussed in class.

```
Convert an infix expression to postfix:
As long as there are more tokens, get the next token.
    if the token is an operand, append it to the postfix string.
    if the token is "(", push it onto the stack.
    if the token is an operator, (order operators by precedence)
        if the stack is empty, push the operator onto the stack.
        if the stack is not empty, pop operators of greater or equal precedence from the stack and append them to postfix string, stop when you encounter "(" or an operator of lower precedence or when the stack is empty. And then, push the new operator onto the stack.
    when you encounter a ")", pop operators off the stack and append them to the end of the postfix string until you encounter matching "(". (Ignore/remove "(". )
    when you reach the end of the infix string, append the remaining content of the stack to the postfix string.
```

IV. Test steps and process records

Token	Actions/Operations	Stack(bottom to top)	Postfix string
a	Append “a” to the postfix string		a
-	Stack is empty, push the “-” onto the stack	–	a
(	Push “(“ onto the stack	(–	
b	Append “b” to the postfix string	(–	ab
+	Push ”+” onto the stack	+ (–	ab
c	Append “c” to the postfix string	+ (–	abc
)	Pop operators off the stack and append them to the end of the postfix string until encounter matching "("	–	abc+
/	Push “/“ onto the stack	/ –	
d	Append “d” to the postfix string	/ –	abc+d
+	Pop operators of greater or equal precedence from the stack and append them to postfix string, then push “+” onto stack	+	abc+d/-

e	Append “e” to the postfix string		abc+d/-e
	Reach the end of the infix string, append the remaining content of the stack to the postfix string.		abc+d/-e+

V. Experimental gains and experiences

1.Learned the operation and application of stack.
2.Review the conversion of infix expression and postfix expression.
3.Learned how to convert expressions through stacks.