20 October 2023 10:14

1) start scanning from left to right
2) If(current element is an operand)
   Append it to the postfix expression
3) Else if (current element is opening bracket '(')
   Push it onto the stack
4) Else if current element is closing bracket ')'
   - Pop elements from the stack and append them to postfix exp till its corresponding opening brackets does not occur
   - Pop opening brackets from the stack and discard both the brackets

   Else
   // If(current element is an operator)
   While(stack is not empty && priority of topmost element >= priority of current element)
   {
   Pop element from the stack and append it to postfix expression
   }

   Push current element onto the stack.
5) Repeat the above steps till end of infix expression
6) Pop all the remaining elements from the stack one by one and append them to postfix expression

Infix expression :
5+9-4*(8-6/2)+1$(7-3)

stack

Current element :

Postfix expression: 59+4862/-*-173-$+

59+4862/-*-173-$+

5+9-4*(8-6/2)+1$(7-3)

(6) (7) (5) (2) (1) (8) (4) (3)

5+9-4*(8-6/2)+1$(7-3)

5+9-4*(8-62/)+1$(7-3)

5+9-4*862/-+1$(7-3)

5+9-4*862/-+1$73-

5+9-4*862/-+173-$

5+9-4862/-*+173-$

59+-4862/-*+173-$

59+4862/-*-+173-$

( )
$ → ^
*/ %
+-

s9f4862/-*-173-$+ → output