

2. If the character is a digit, convert the character into int and push the integer into the stack.
3. If the character is an operator,
 - Pop the elements from the stack twice obtaining two operands.
 - Perform the operation
 - Push the result into the stack.

Conversion of infix to postfix

Here, we will use the stack data structure for the conversion of infix expression to prefix expression. Whenever an operator will encounter, we push operator into the stack. If we encounter an operand, then we append the operand to the expression.

Rules for the conversion from infix to postfix expression

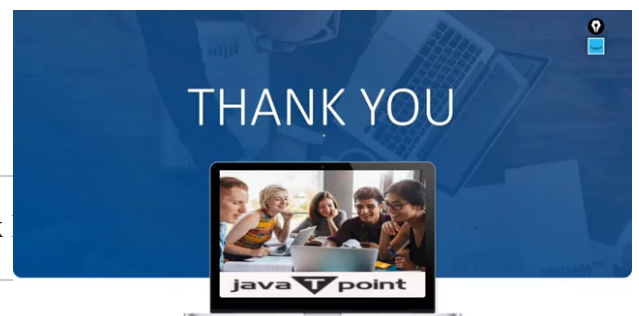
1. Print the operand as they arrive.
2. If the stack is empty or contains a left parenthesis on top, push the incoming operator on to the stack.
3. If the incoming symbol is '(', push it on to the stack.
4. If the incoming symbol is ')', pop the stack and print the operators until the left parenthesis is found.
5. If the incoming symbol has higher precedence than the top of the stack, push it on the stack.
6. If the incoming symbol has lower precedence than the top of the stack, pop and print the top of the stack. Then test the incoming operator against the new top of the stack.
7. If the incoming operator has the same precedence with the top of the stack then use the associativity rules. If the associativity is from left to right then pop and print the top of the stack then push the incoming operator. If the associativity is from right to left then push the incoming operator.
8. At the end of the expression, pop and print all the operators of the stack.

Let's understand through an example.

Infix expression: $K + L - M * N + (O \wedge P) * W / U / V * T + Q$

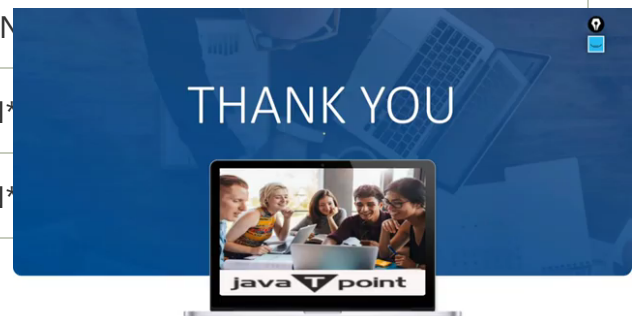
Input Expression	Stack	Postfix
------------------	-------	---------

↑ SCROLL TO TOP



K		K
+	+	
L	+	K L
-	-	K L +
M	-	K L + M
*	- *	K L + M
N	- *	K L + M N
+	+	K L + M N* K L + M N* -
(+	K L + M N* -
O	+	K L + M N* - O
^	+	K L + M N* - O
P	+	K L + M N* - O P
)	+	K L + M N* - O P ^
*	+	K L + M N* - O P ^
W	+	K L + M N* - O P ^ W
/	+	K L + M N* - O P ^ W *
U	+	K L + M N* - O P ^ W * U
/	+	K L + M N* - O P ^ W * U /
V	+	K L + M N
*	+	K L + M N *
T	+	K L + M N *

↑ SCROLL TO TOP



+	+	KL+MN*-OP^W*U/V/T* KL+MN*-OP^W*U/V/T*+
Q	+	KL+MN*-OP^W*U/V/T*Q
		KL+MN*-OP^W*U/V/T*+Q+

The final postfix expression of infix expression $(K + L - M * N + (O \wedge P) * W / U / V * T + Q)$ is $KL+MN*-OP^W*U/V/T*+Q+$.

[< Prev](#)
[Next >](#)

 [For Videos Join Our Youtube Channel: Join Now](#)

Feedback

- Send your Feedback to feedback@javatpoint.com

Help Others, Please Share

↑ SCROLL TO TOP

