

Algorithm for Prefix to Infix Converter

Here, is the algorithm for prefix to infix converter:

- Start scanning the prefix expression from right to left.
- If the current character is an operand (a number or a variable), push it onto a stack.
- If the current character is an operator (+, -, *, /, ^), pop two operands from the stack and concatenates them with the operator in between to form an infix sub-expression. Then push the infix sub-expression onto the stack.
- Repeat steps 2 and 3 until the entire prefix expression has been scanned.
- The final infix expression will be the only item remaining on the stack.

NOTE: '+' denotes the concatenation of strings.

Prefix to Infix Conversion Example with Dry Run

(1) Prefix expression

* - a / b c - / a d e

↑
'e' is an operand push it into the stack.

Stack



(2) Prefix expression

* - a / b c - / a d e

↑
'd' is an operand push it into the stack.

Stack



(3) Prefix expression

* - a / b c - / a d e

↑
'a' is an operand. So, push it into the stack.

Stack



(4) Prefix expression

* - a / b c - / a d e

↑
'/' is an operator. Pop two expressions from the stack.

Operand 1 = a

Operand 2 = d

Operand 3 = /

Stack



|| Pop
V



Calculate the expression

