Expression Trees

Postfix to Expression Tree Rules

Terms: Append x to y means add child x to node y.

Rule1: Operators can have children but operands can't.

Rule2: Nodes can only have 2 children.

Rule3: When appending nodes, always append to right first. If right is occupied, then append to left.

Postfix to Expr Tree Algorithm

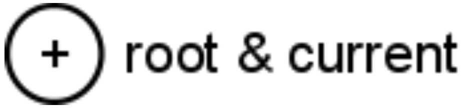
1. Get the last symbol (rightmost) of postfix notation, create a node for it and designate the new node as the root.

2. Set the root node as current node.

3. For each element from right to left (excluding the last):
 3.1 Create a node for it.
 3.2 If current node cannot have more children, search for the first parent/grandparent that can have more children and set it as the current node.
 3.3 Append the new node to the current node.
 3.4 Set new node as current node.

Postfix to Exp Tree (Step by Step)

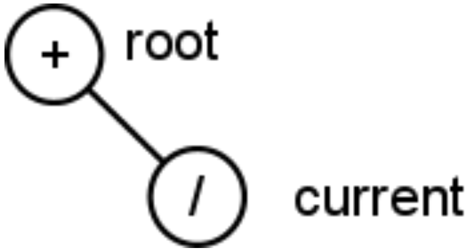
Postfix: 2, 6, *, 3, 8, /, +
Step 1 & 2



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

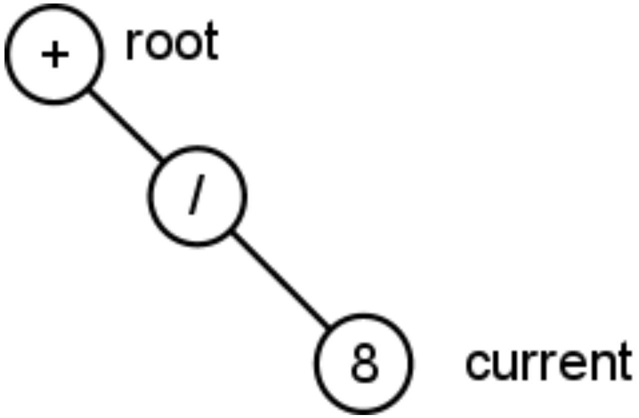
Step 3.1, 3.3 & 3.4



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

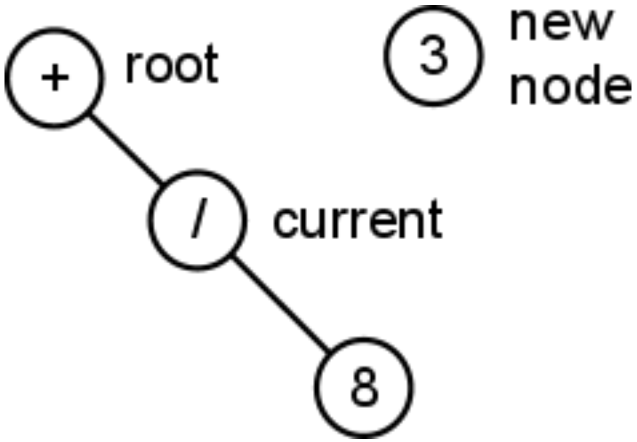
Step 3.1, 3.3 & 3.4



Postfix to Exp Tree (Step by Step)

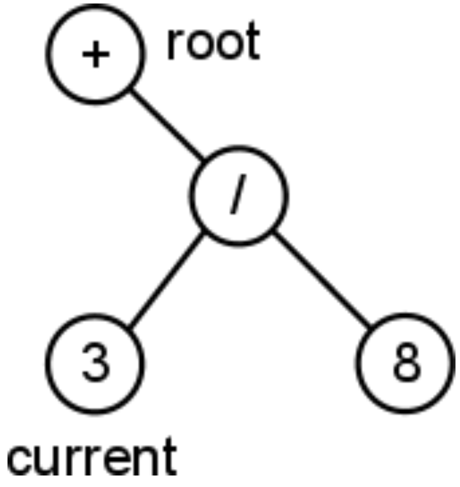
Postfix: 2, 6, *, 3, 8, /, +

Step 3.1 & 3.2



Postfix to Exp Tree (Step by Step)

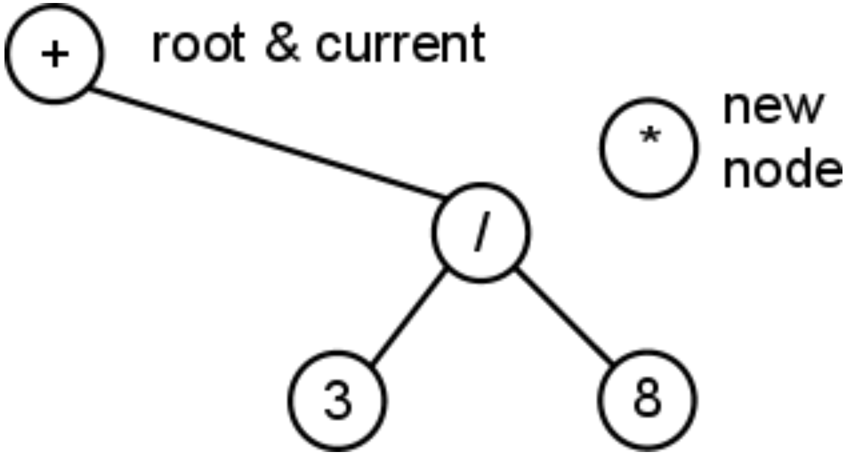
Postfix: 2, 6, *, 3, 8, /, +
Step 3.3 & 3.4



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

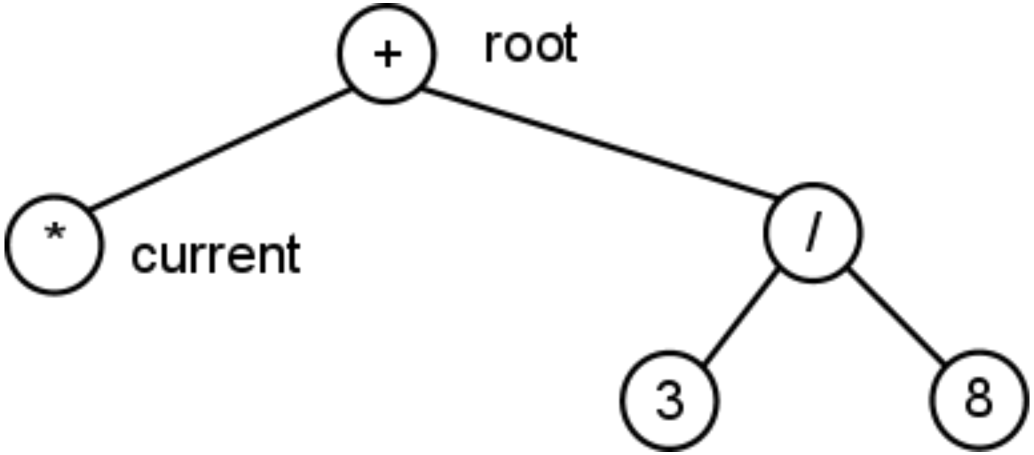
Step 3.1 & 3.2



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

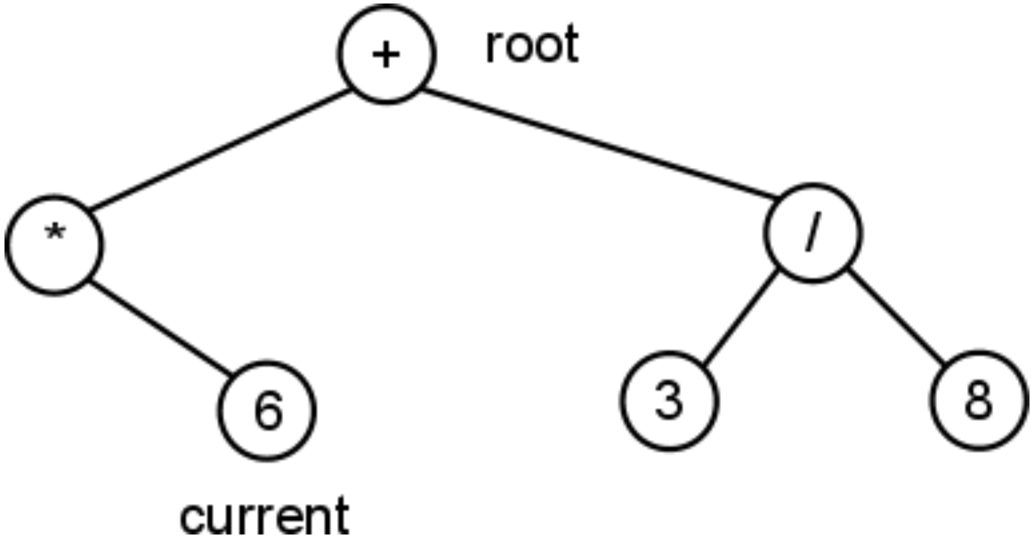
Step 3.3 & 3.4



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

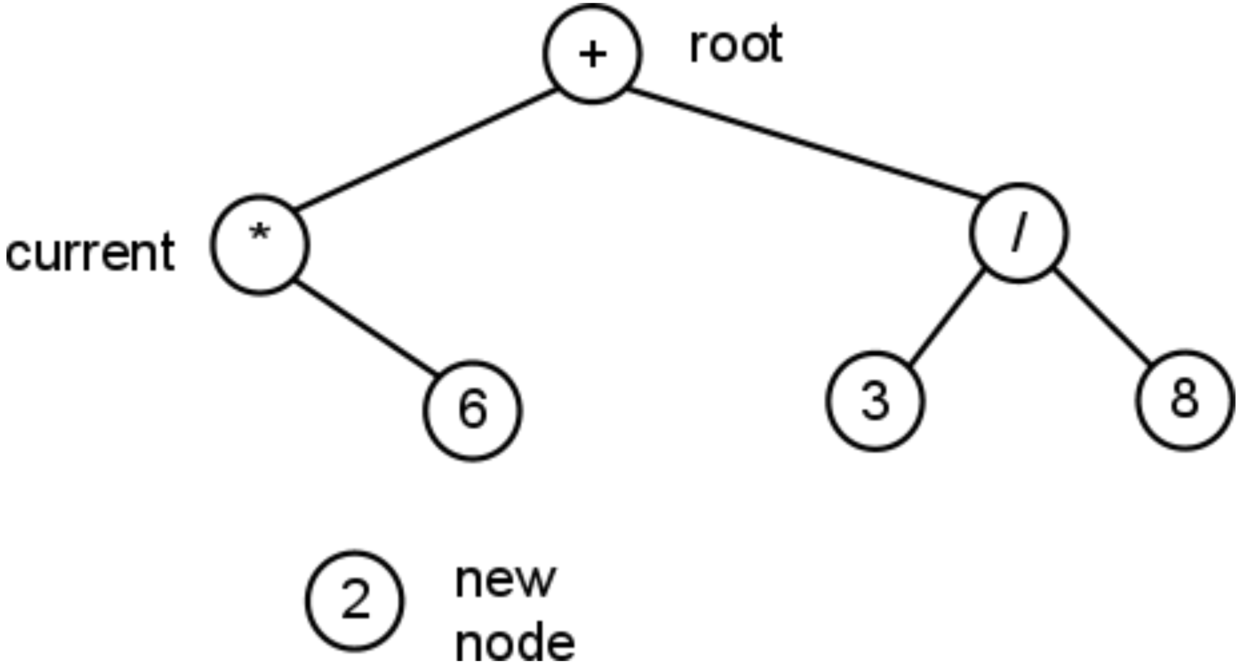
Step 3.1, 3.3 & 3.4



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

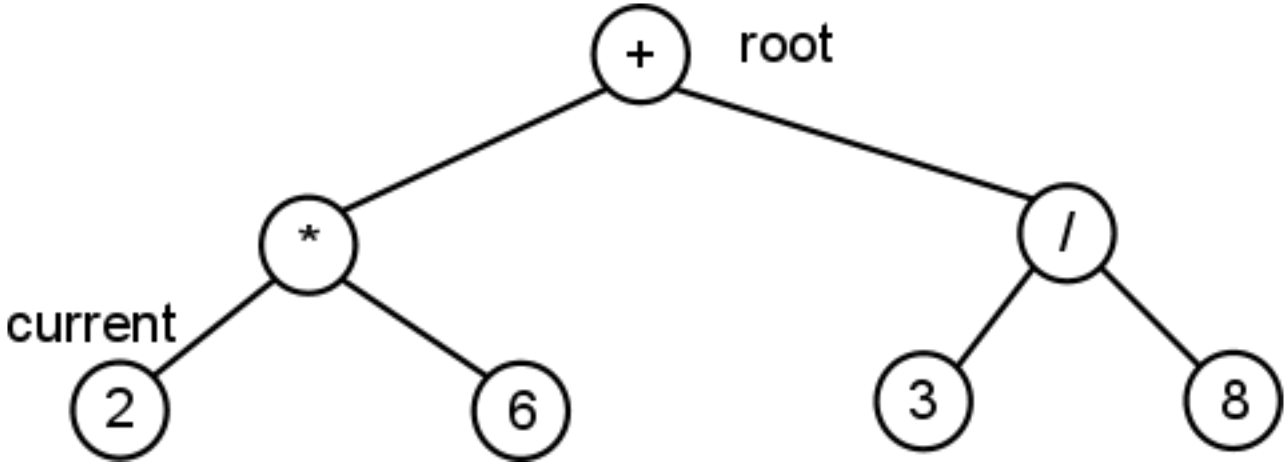
Step 3.1 & 3.2



Postfix to Exp Tree (Step by Step)

Postfix: 2, 6, *, 3, 8, /, +

Step 3.3 & 3.4

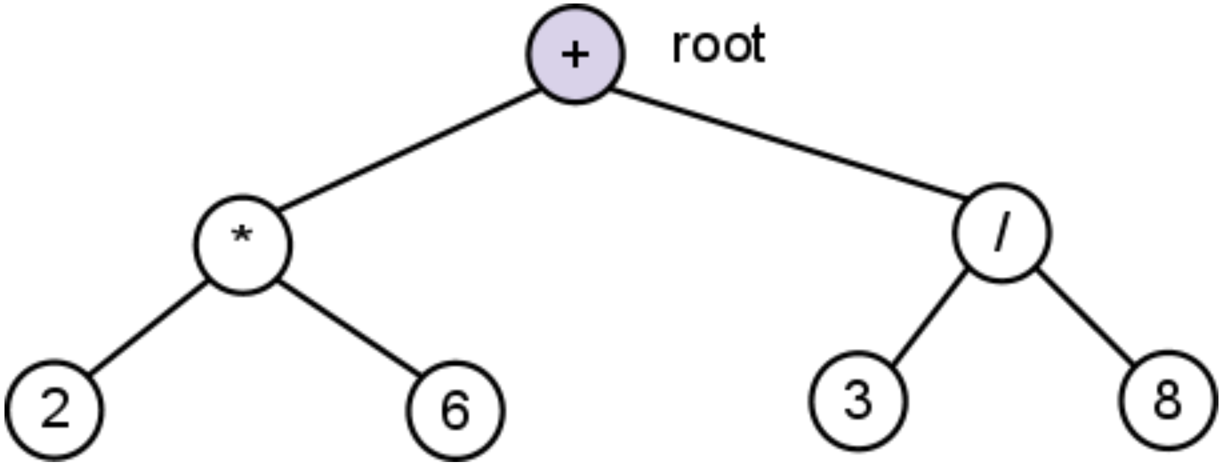


Prefix from Expression Tree

Rule1: Start at the root node.
Rule2: When node is visited for the first time, output value of node.
Rule3: Go from left to right when visiting children.
Rule4: If left child has children, visit their children first before going to right child.
Rule5: Prefix notation is complete when every node is visited.

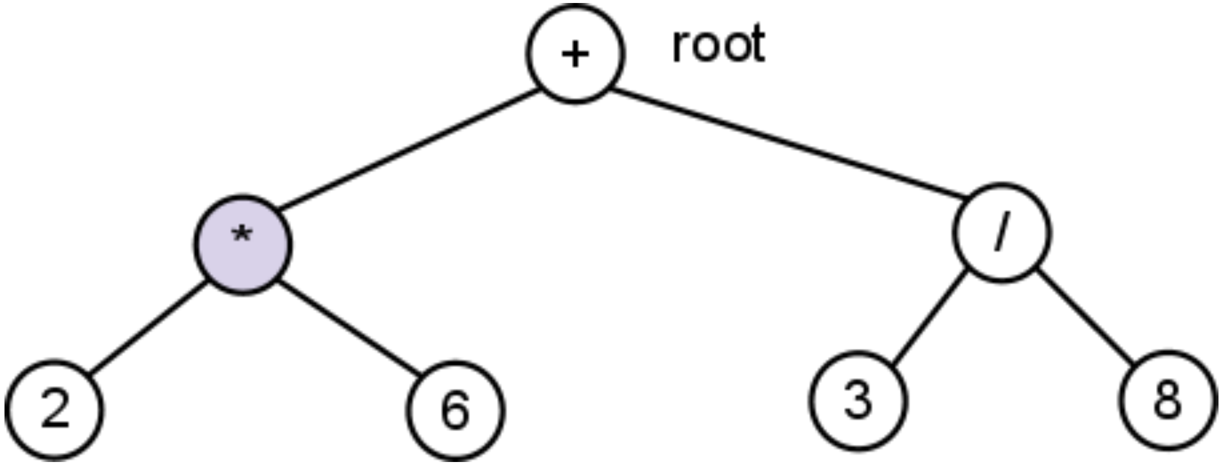
Prefix from Exp Tree (Step by Step)

Output: +



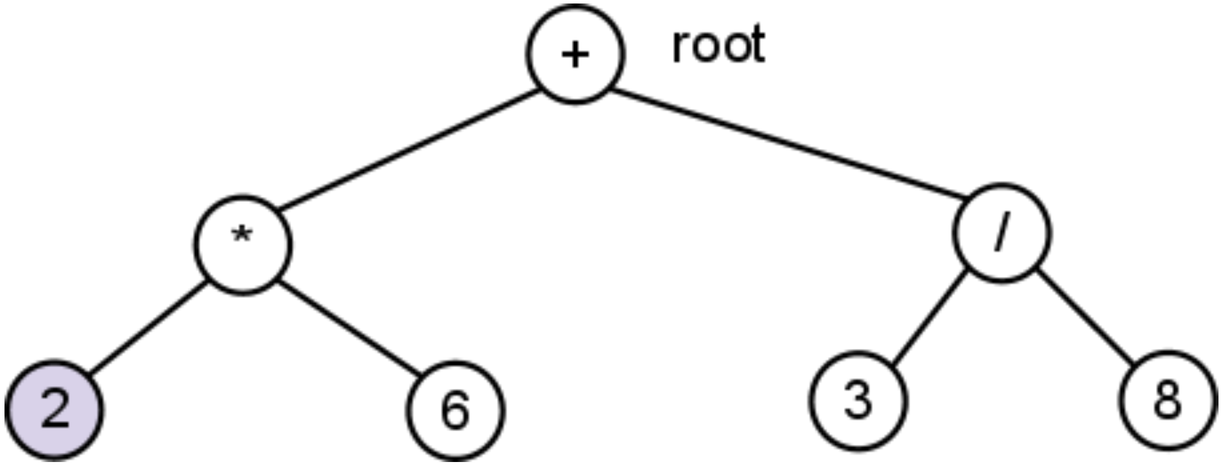
Prefix from Exp Tree (Step by Step)

Output: +, *



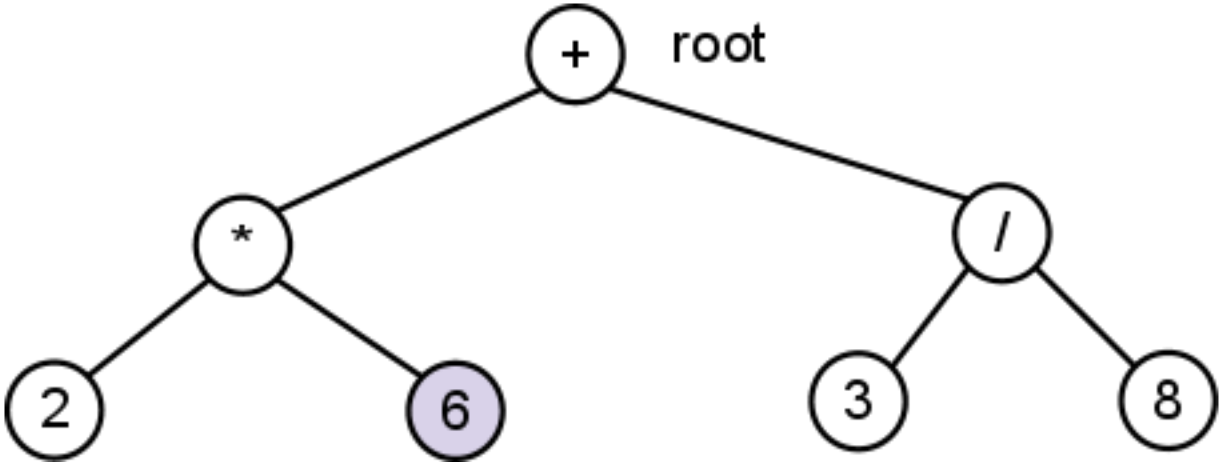
Prefix from Exp Tree (Step by Step)

Output: +, *, 2



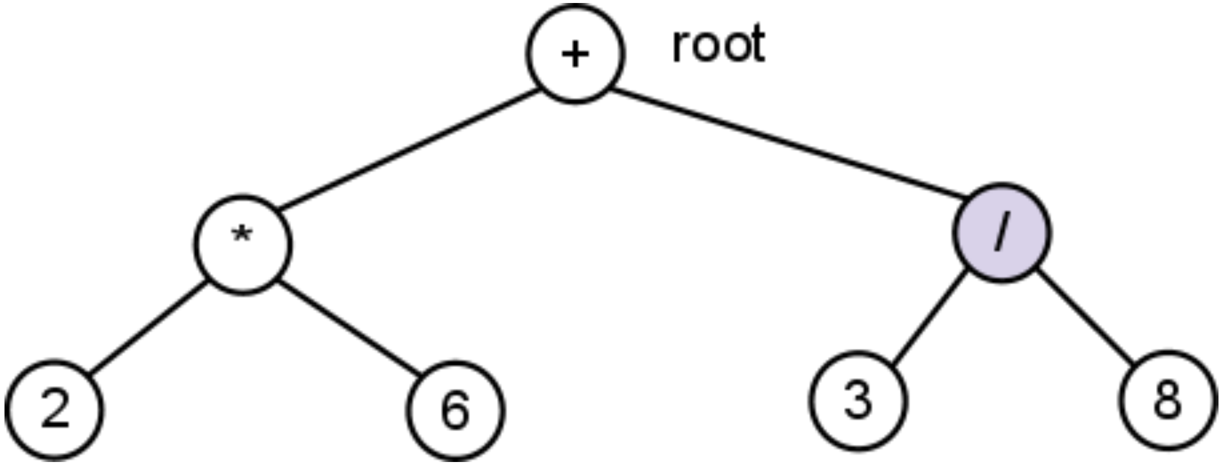
Prefix from Exp Tree (Step by Step)

Output: +, *, 2, 6



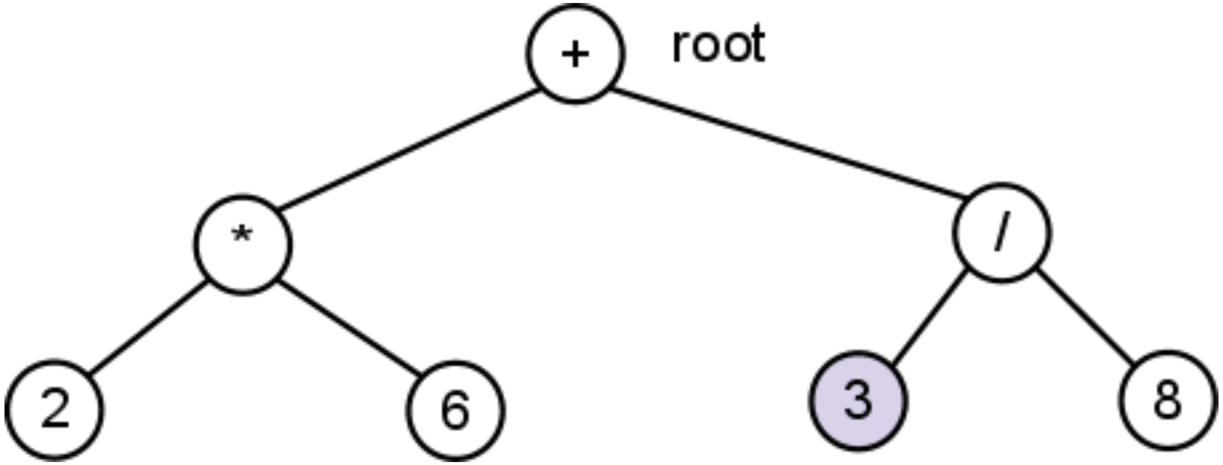
Prefix from Exp Tree (Step by Step)

Output: +, *, 2, 6, /



Prefix from Exp Tree (Step by Step)

Output: +, *, 2, 6, /, 3



Prefix from Exp Tree (Step by Step)

Output: +, *, 2, 6, /, 3, 8