Construction of Binary tree from Infix (Inorder) and prefix (Preorder) Expressions

Prof. Harish D.G.
Dept. of Computer and IT
College of Engineering,Pune(COEP)
www.harishgadade.com

Construction of Binary tree from Infix (Inorder) and prefix (Preorder) Expressions

Consider the Inorder & Preorder Expressions

a – b * c + d

* - a b + c d

Construction of Binary tree from Infix (Inorder) and prefix (Preorder) Expressions

Consider the Inorder & Preorder Expressions

a – b * c + d

* - a b + c d

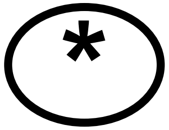
Steps:

1. Read the preorder sequence, the first element becomes the root.
2. Now Scan the infix sequence, till you get an element found in step 1. Place all the elements left of this element to the left of root and other to right.
3. Repeat step 1 & 2, till all the elements from infix sequence gets placed in tree.
4. Stop.

Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

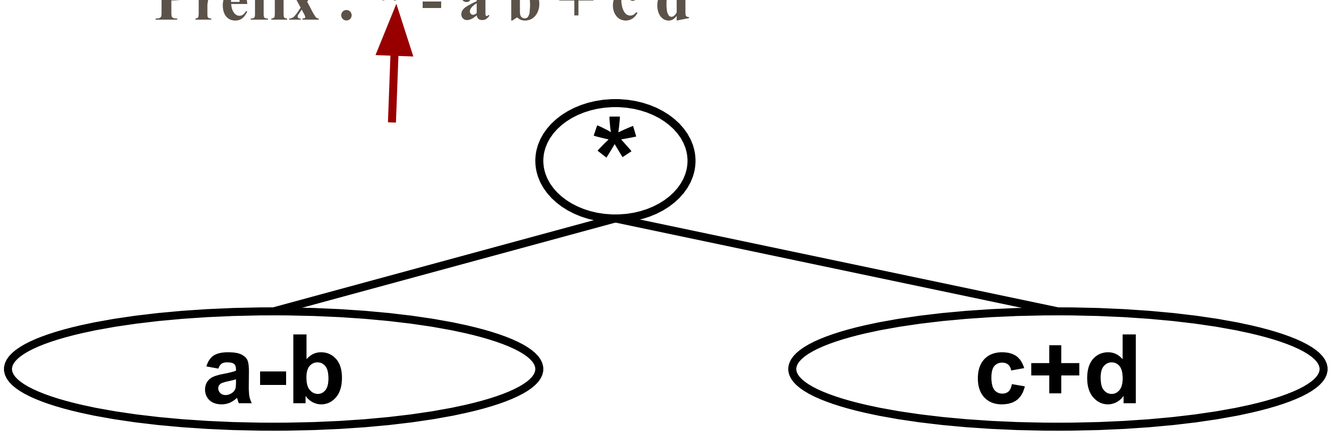
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

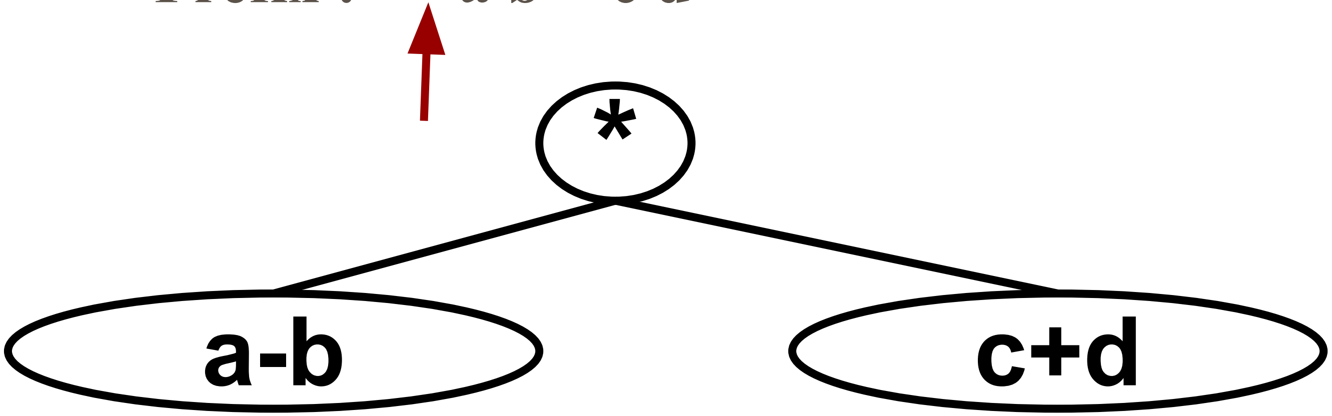
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

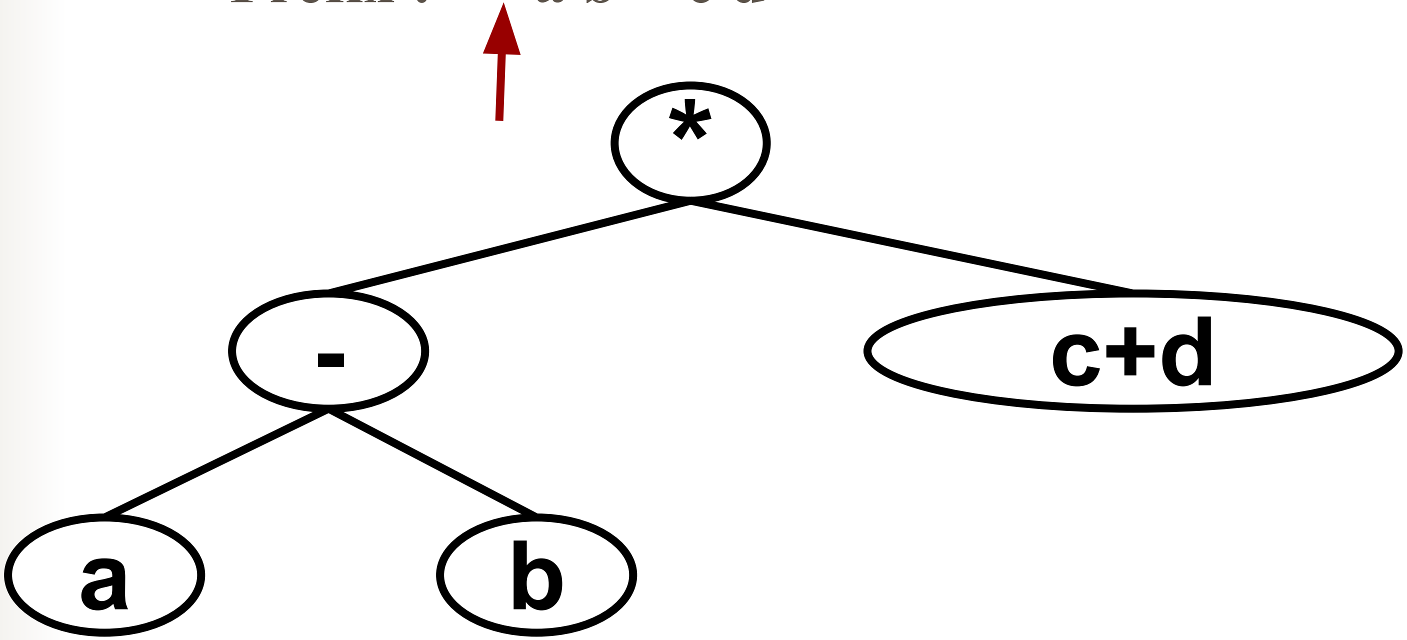
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

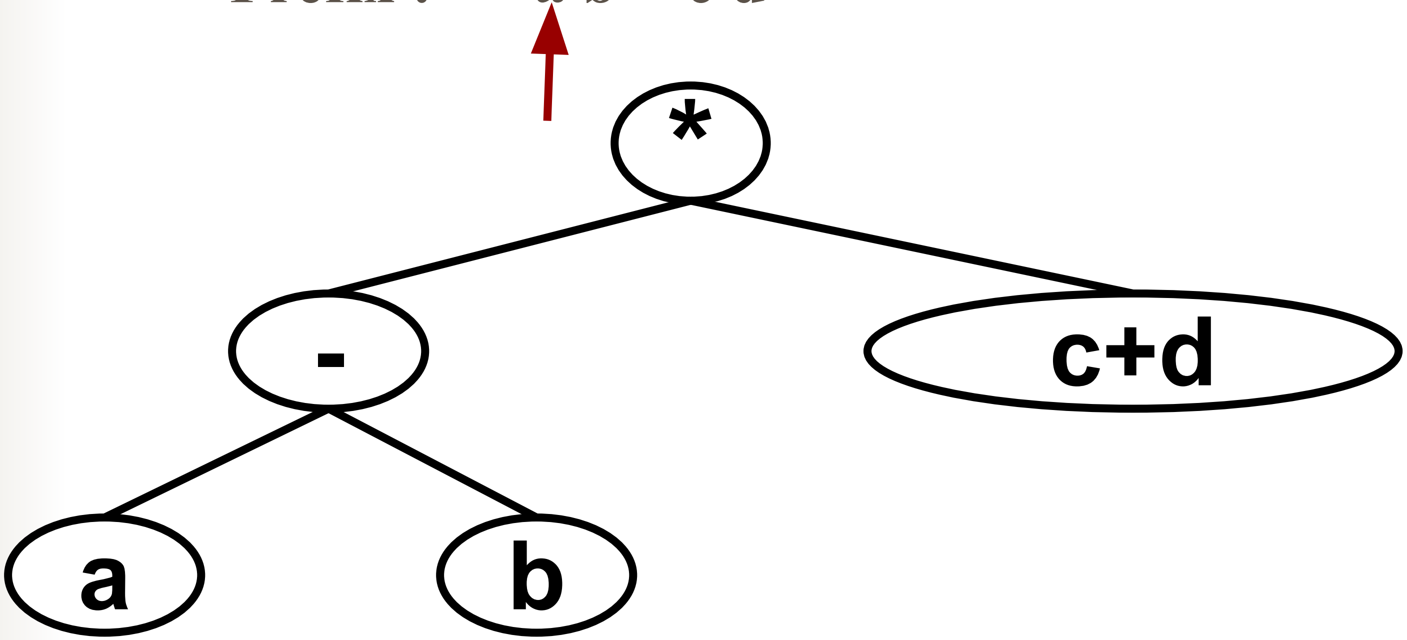
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

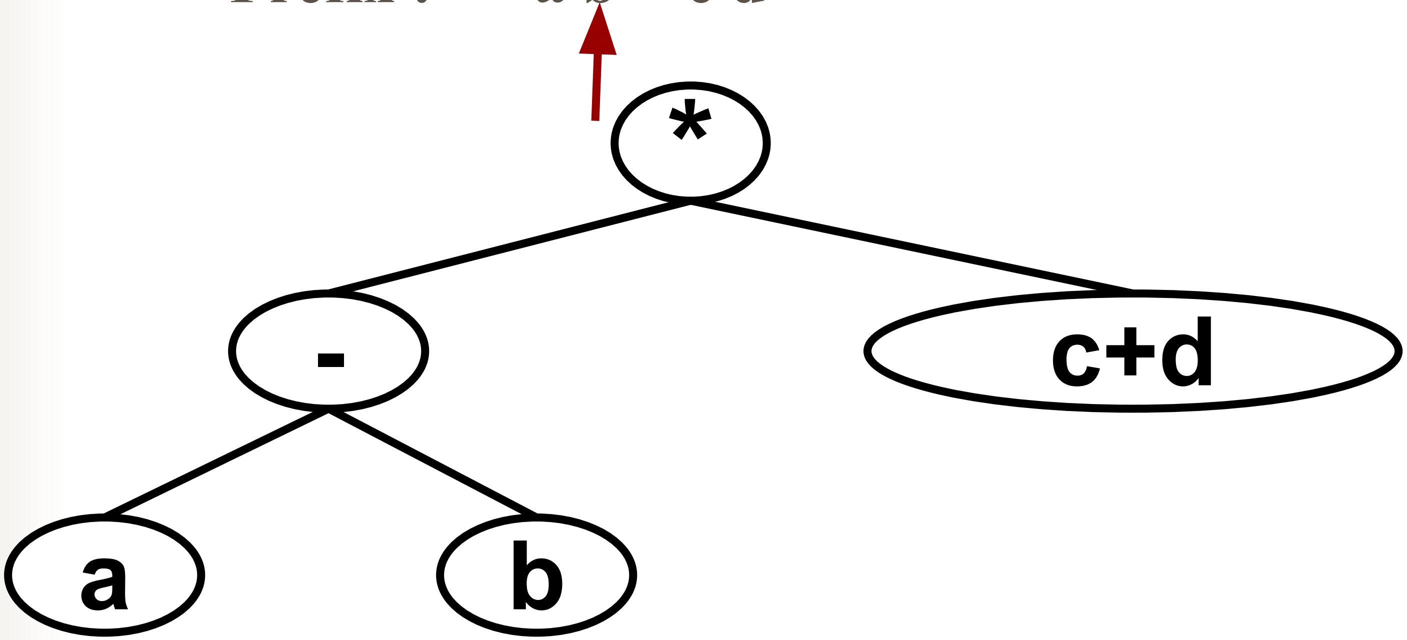
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

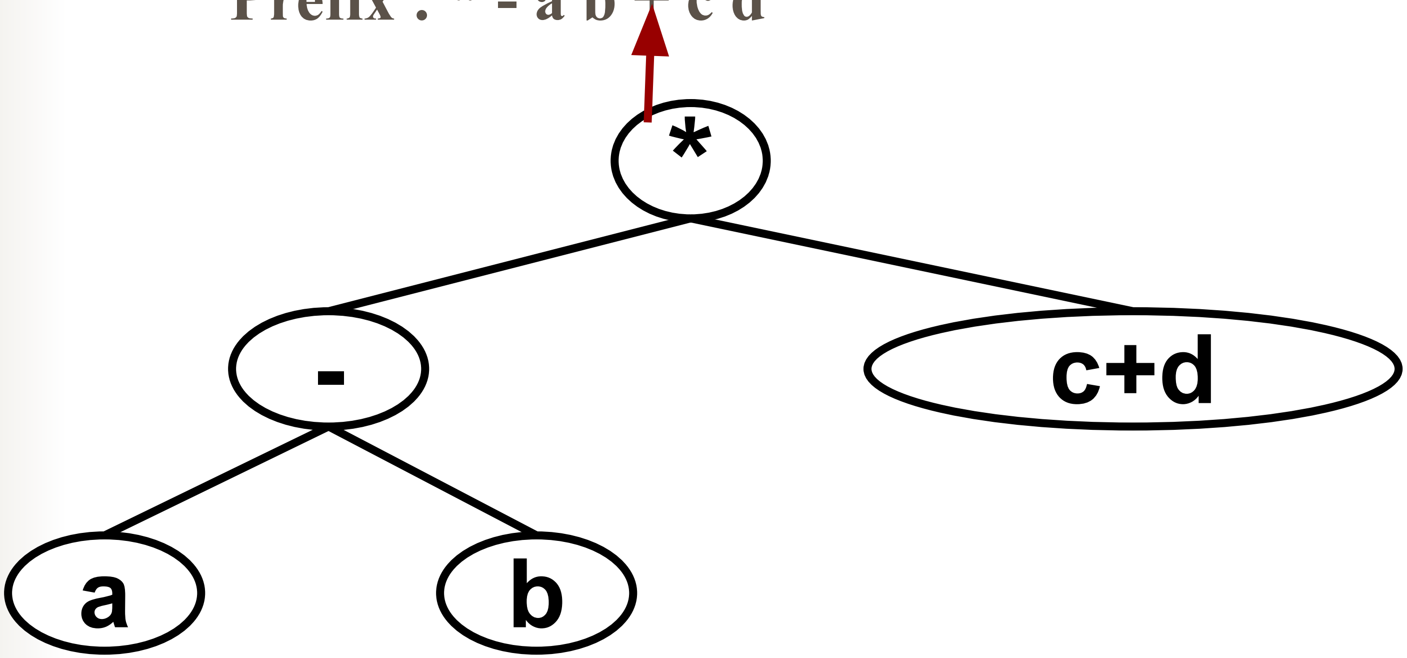
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

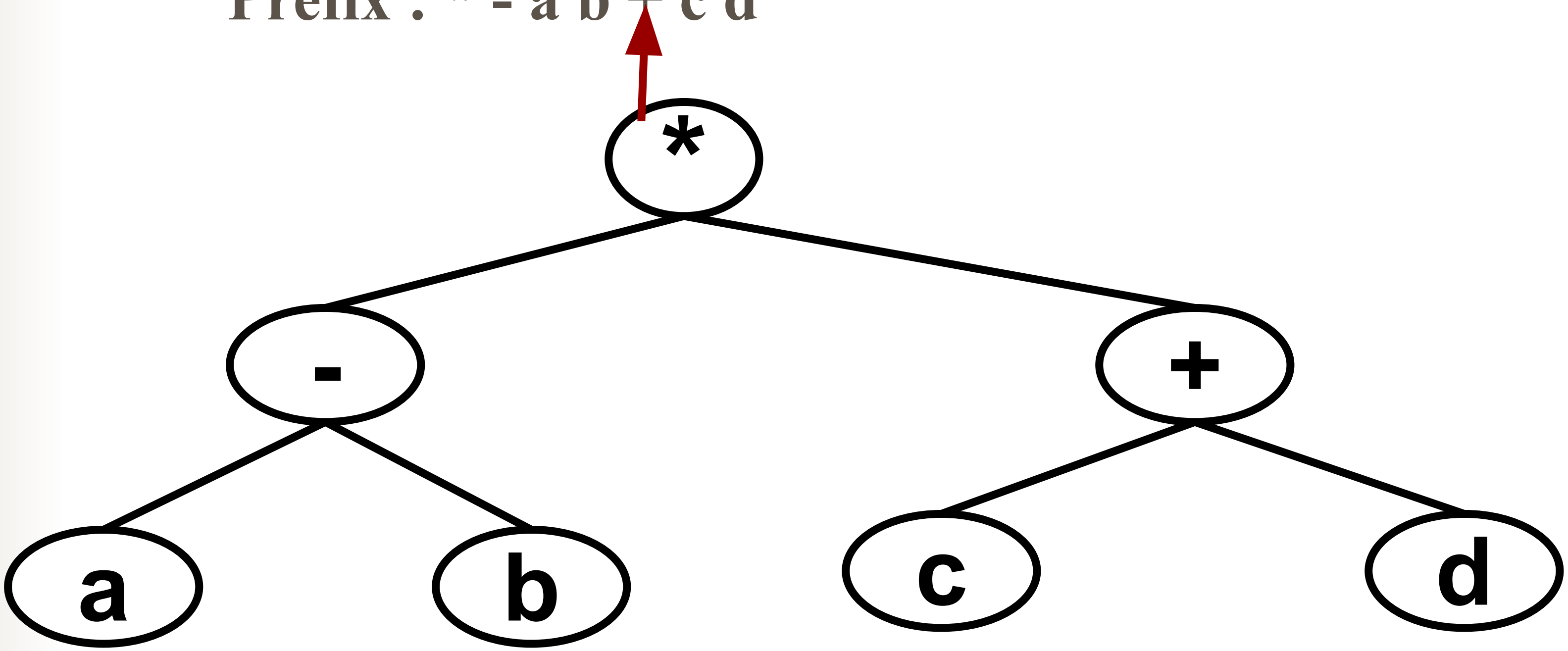
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

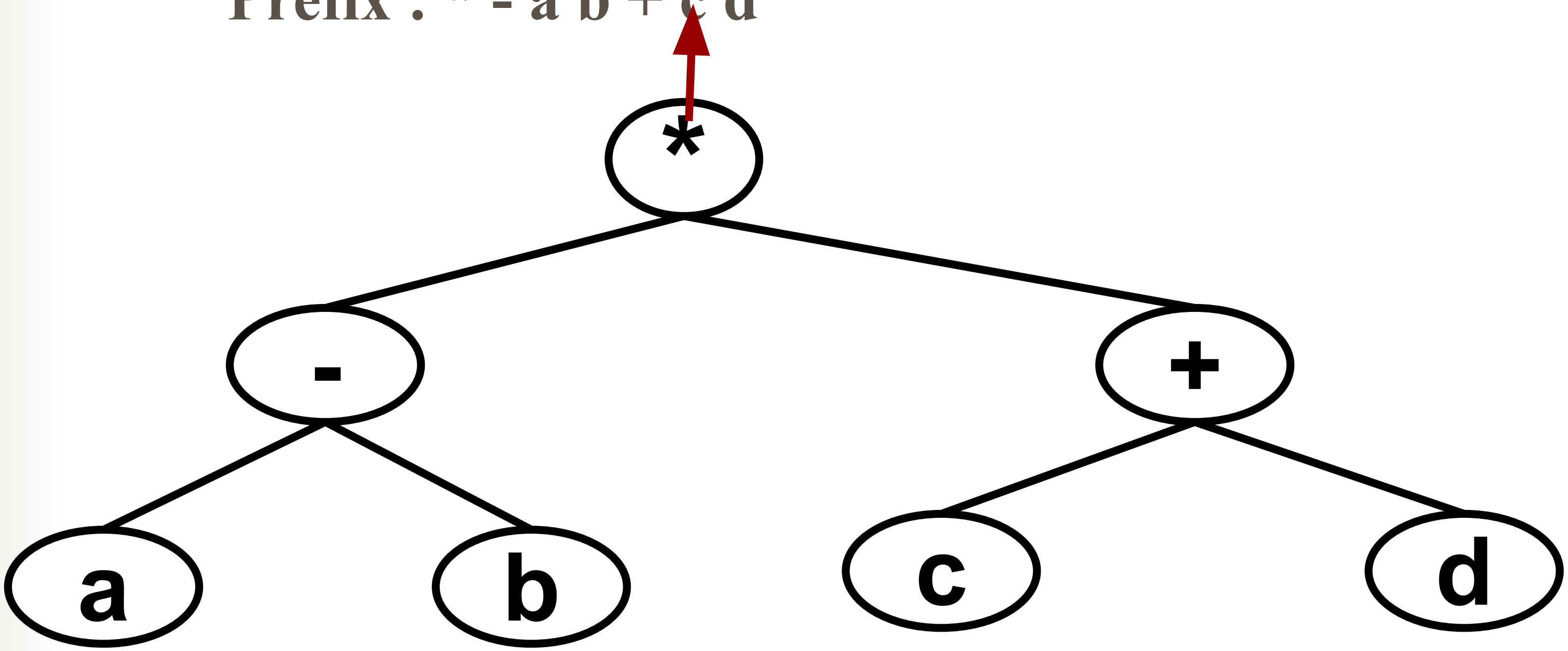
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

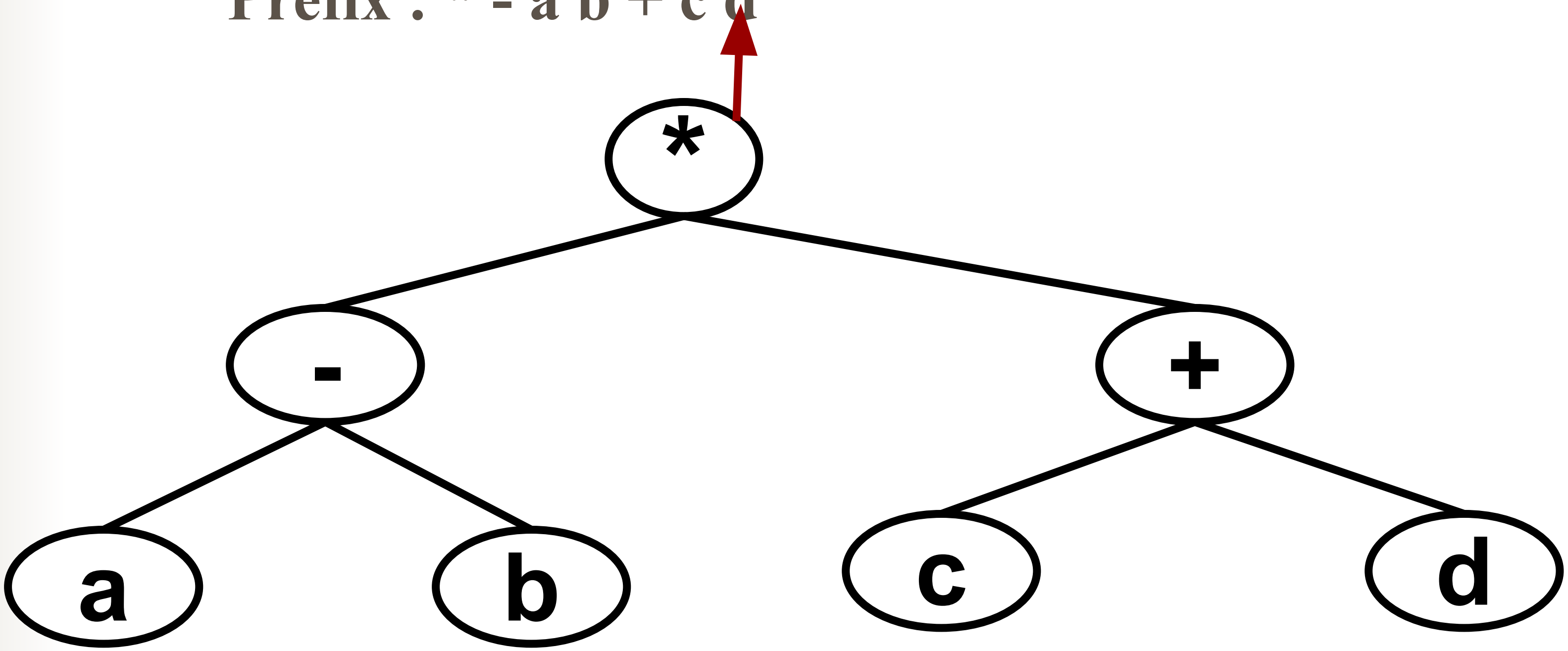
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

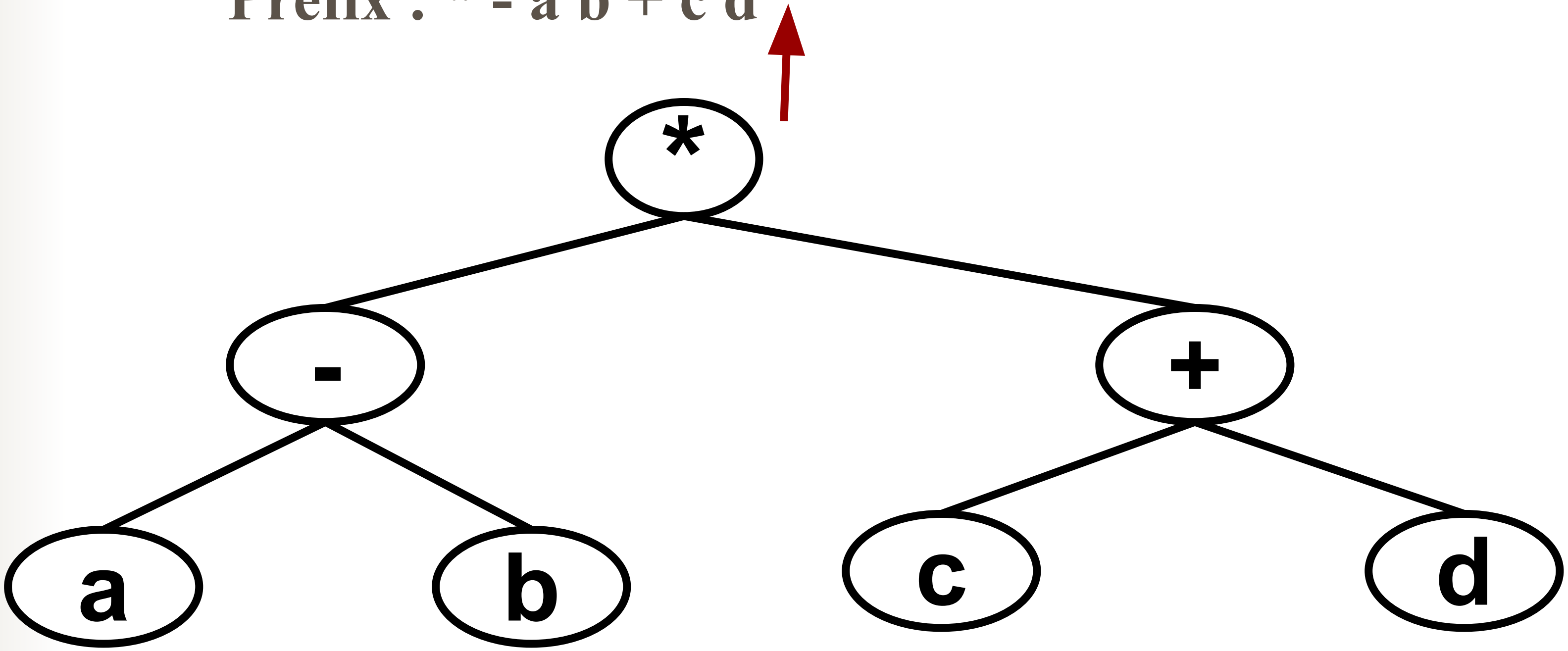
Prefix : * - a b + c d



Consider the Inorder & Preorder Expressions

Infix : a – b * c + d

Prefix : * - a b + c d



Final Binary Tree

- Construct a Binary tree from following infix and prefix expressions.

1. (a +b) * (c – d)

   * + a b - c d

2. a – b / (c * d ^ e)

   - a / b * c ^ d e

3. a + b – c

   - + a b c

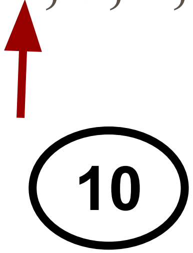
4. 4, 5, 8, 10, 15, 20, 25 (Inorder)

   10, 5, 4, 8, 20, 15, 25 (Preorder)

Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

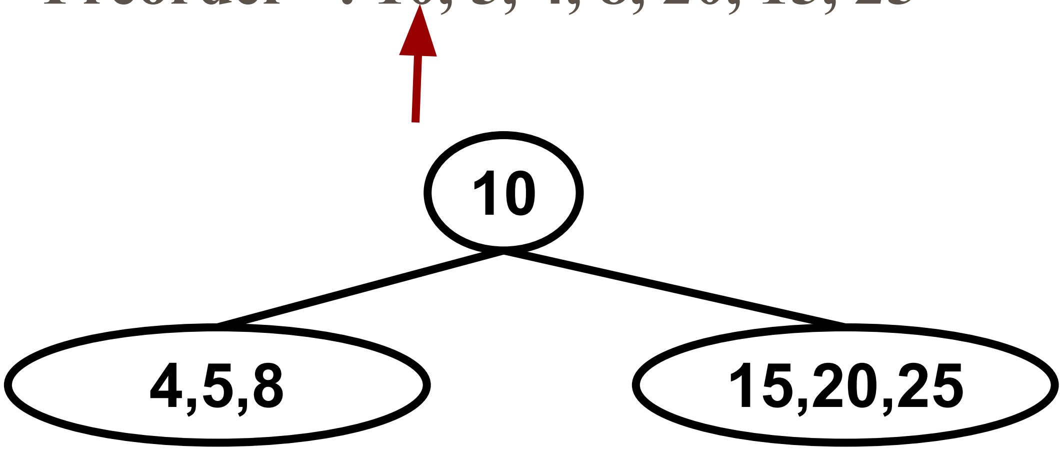
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

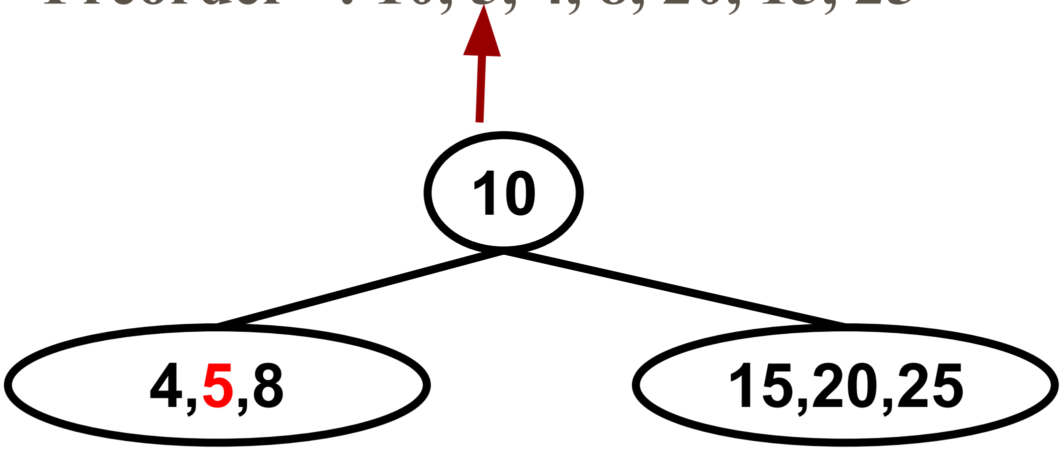
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

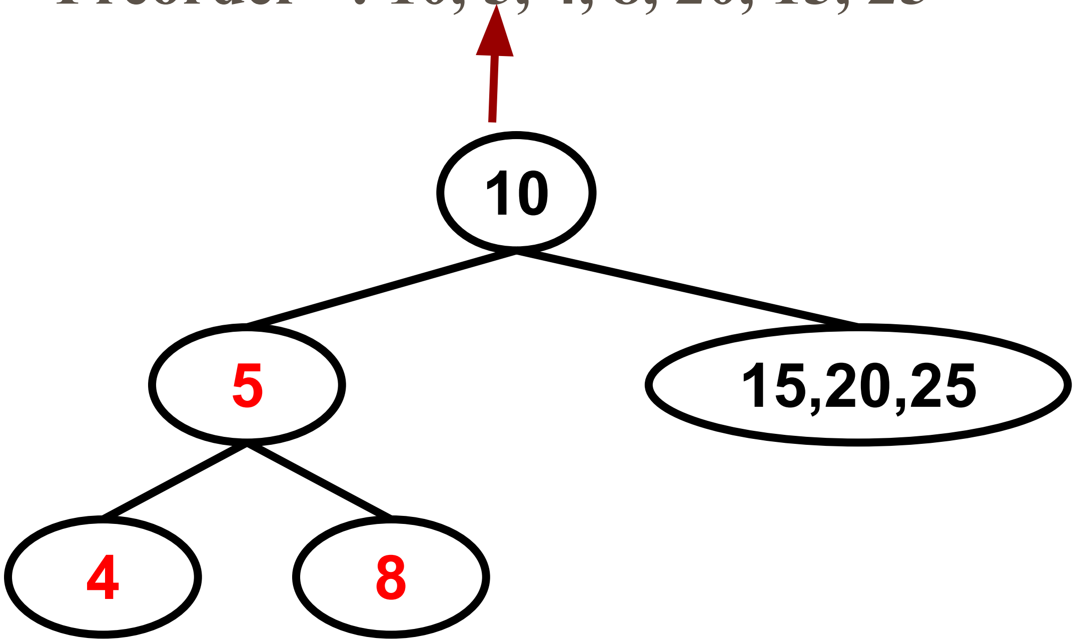
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

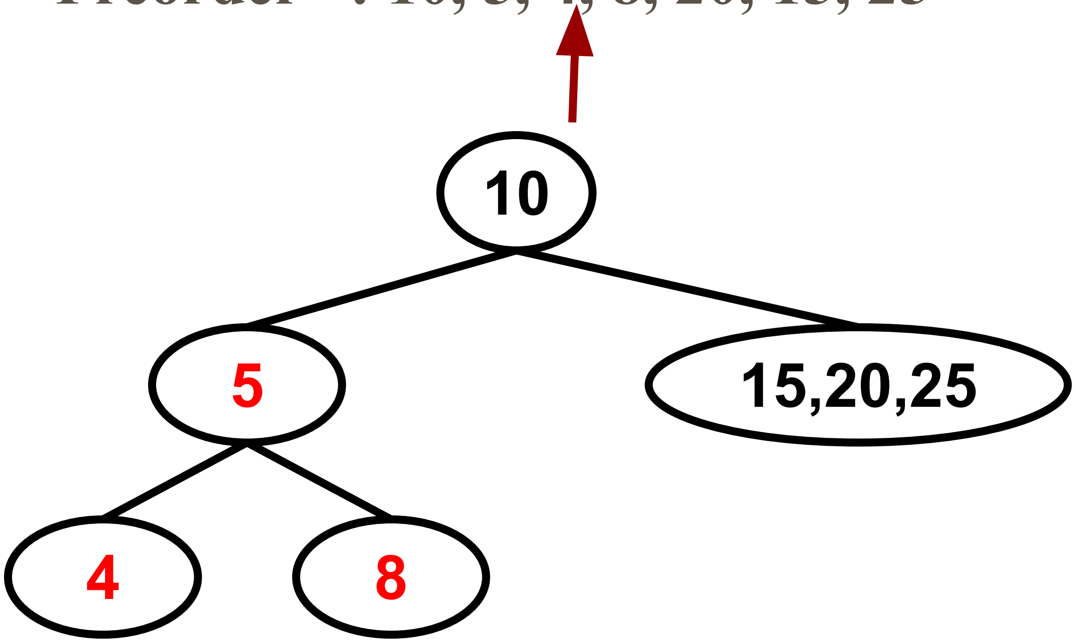
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

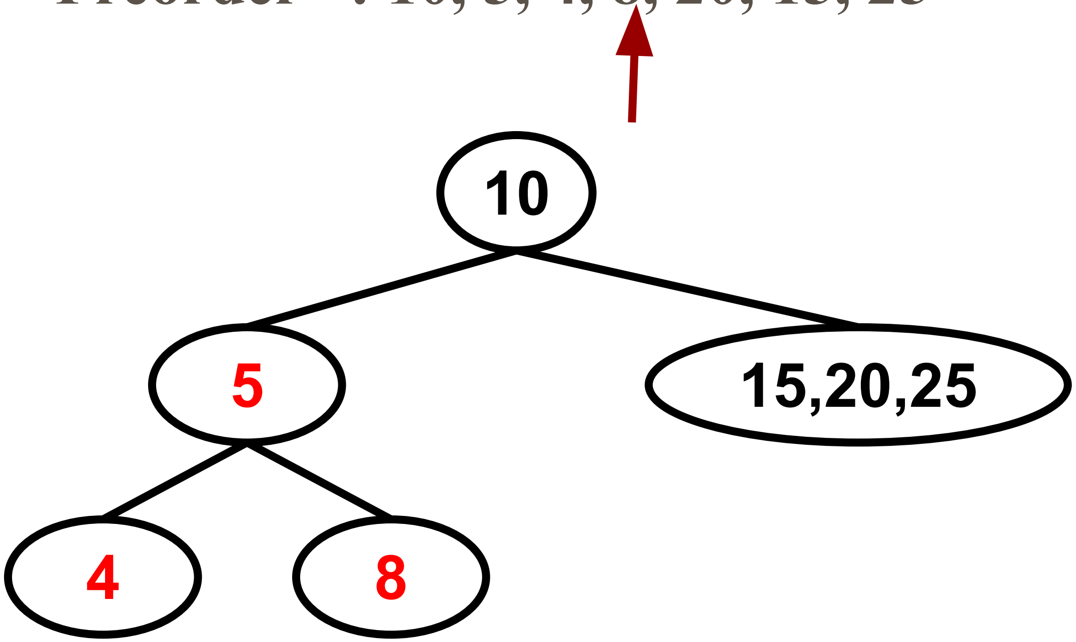
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

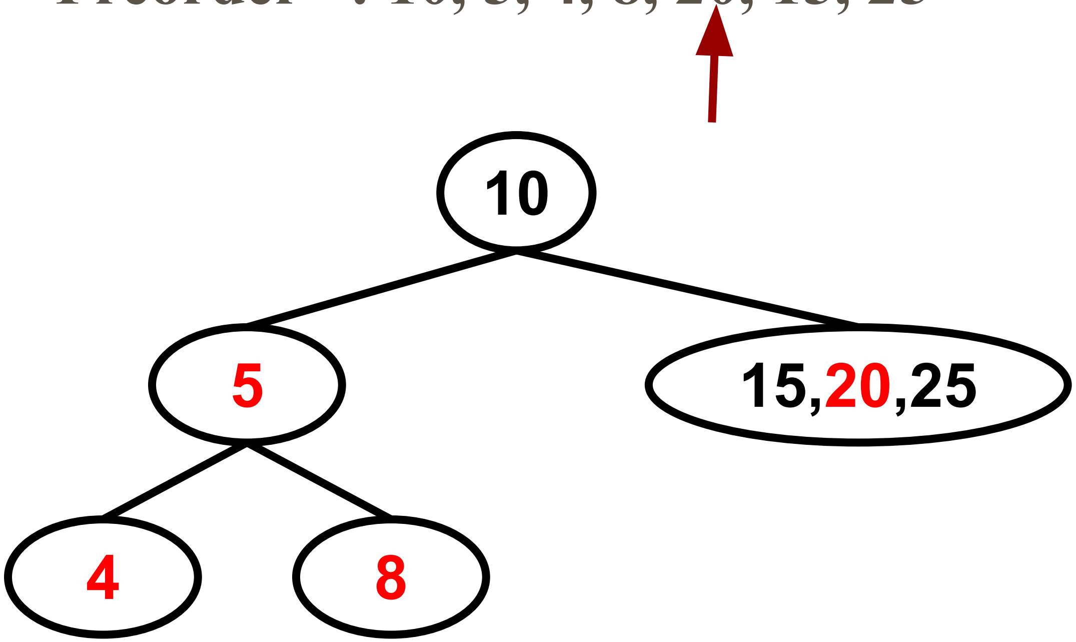
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

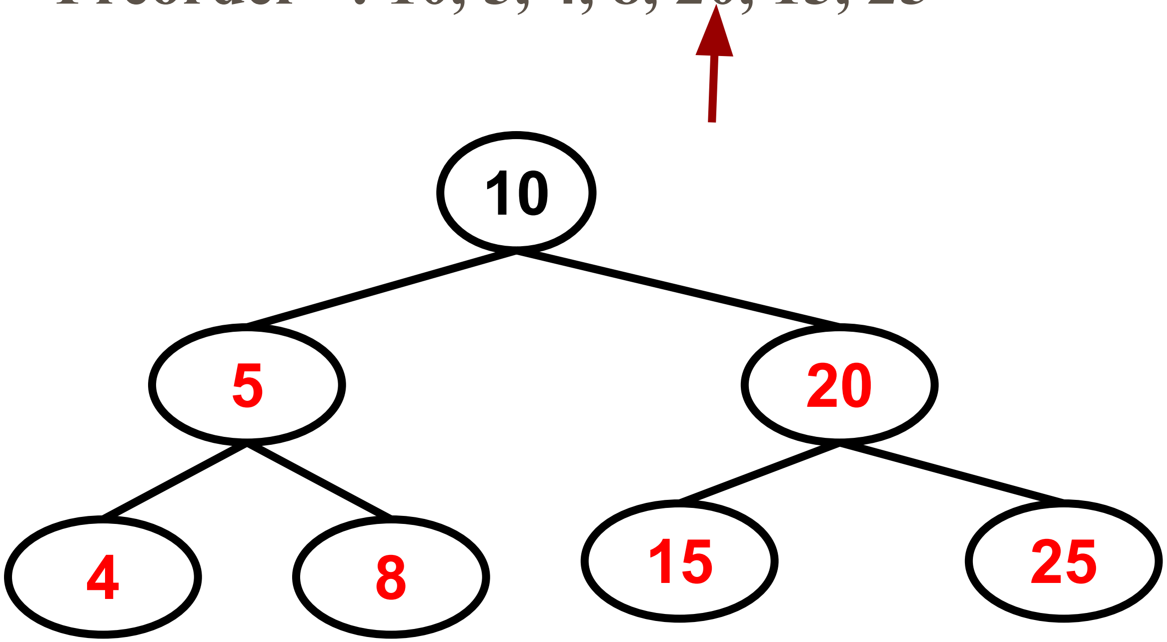
Preorder : 10, 5, 4, 8, 20, 15, 25



Consider the Inorder & Preorder Expressions

Inorder : 4, 5, 8, 10, 15, 20, 25

Preorder : 10, 5, 4, 8, 20, 15, 25


10
5
20
4
8
15
25