Tree Traversal (cont.)
Infix, Prefix and Postfix Notations

Course Code: 00090 Course Title: Discrete Mathematics

Dept. of Computer Science
Faculty of Science and Technology

Lecturer No:	**24**	**Week No:**	**13**	**Semester:**	**Summer 21-22**
Lecturer:	*Md. Mahmudur Rahman (mahmudur@aiub.edu)*				

Lecture Outline

8.3 Tree Traversal (cont.)

- Infix, Prefix and Postfix Notations
- Representing arithmetic expressions using ordered rooted trees
- Evaluating Prefix and Postfix Expressions

Objectives and Outcomes

- Objectives: To understand how to represent arithmetic expressions using ordered rooted trees, to evaluate the value of prefix and postfix expressions.

- Outcomes: The students are expected to be able to construct the ordered rooted tree for a given arithmetic expression, be able to evaluate the value of prefix and postfix expressions.

Infix, Prefix and Postfix Notations

- Complicated expressions, such as compound propositions, combinations of sets, and arithmetic expressions can be represented using ordered rooted tree.
- For instance, consider the representation of an arithmetic expression involving the operators + (addition), – (subtraction), * (multiplication), / (division), and (exponentiation).

Infix, Prefix and Postfix Notations

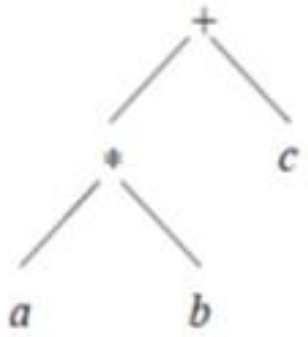
- Complicated arithmetic expressions can be represented by an ordered rooted tree
  - **Internal vertices** represent **operators**
  - **Leaves** represent **operands**
  - **Parentheses** indicate the **order of the operations**
- Build the tree bottom-up
  - Construct smaller subtrees
  - Incorporate the smaller subtrees as part of larger subtrees

Infix, Prefix and Postfix Notations

- **Prefix notation**: The form of an expression obtained from a preorder traversal of the tree representing this expression.
- **Infix notation**: The form of an expression obtained from an inorder traversal of the binary tree representing this expression.
- **Postfix notation**: The form of an expression obtained from a postorder traversal of the tree representing this expression.

Infix, Prefix and Postfix Notations

- Examples: infix, prefix, and postfix notations of $a \times b + c$
  - Infix: $a * b + c$
  - Prefix: $+ * abc$ (also called Polish notation)
  - Postfix: $ab * c+$ (also called reverse Polish notation)
- Represented by ordered rooted trees.



Infix, Prefix and Postfix Notations

- **Example 5:** What is the **ordered rooted tree** that represents the expression $((x + y) \uparrow 2) + ((x - 4) / 3)$?
- **Solution**:

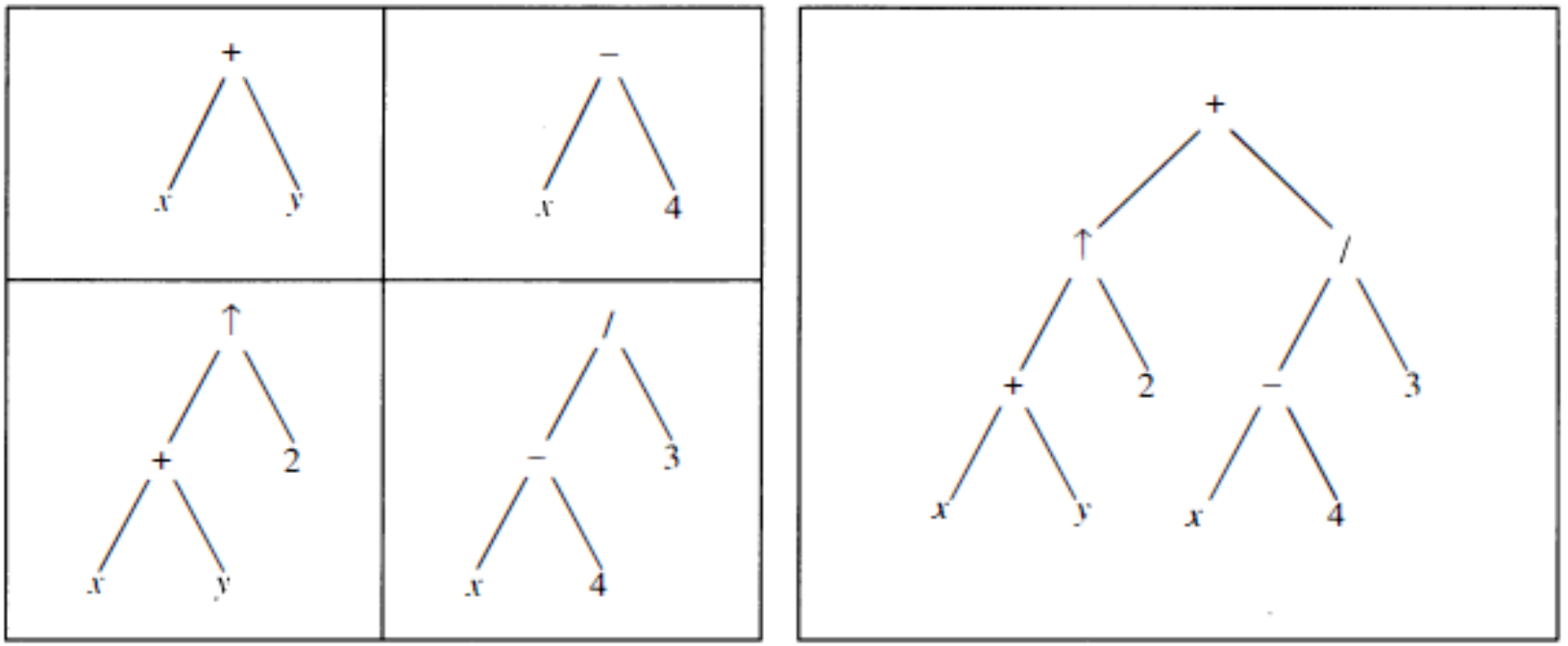


Figure 10

Infix, Prefix and Postfix Notations

- **Example 6:** What is the **prefix form** for $((x + y) \uparrow 2) + ((x - 4) / 3)$?

- **Solution:** We obtain the prefix form for this expression by traversing the **binary tree** that represents it. This produces $+ \uparrow + x\ y\ 2\ /\ -\ x\ 4\ 3$

Infix, Prefix and Postfix Notations

- **Example 8:** What is the **postfix form** for
  ((x + y)↑ 2) + ((x – 4) / 3) ?

- **Solution:** The postfix form of the expression is obtained by carrying out a **postorder traversal** of the binary tree (see slide # 43, Figure 10) for this expression.

  This produces the postfix expression: **x y + 2 ↑ x 4 – 3 / +**

Evaluating a Prefix Expression

- In an prefix expression, a binary operator precedes its two operands
- The expression is evaluated **from right to left**
- Look for the **first operator** from the right
- Perform the corresponding operation with that operator with the two operands immediately to its right

Evaluating a Prefix Expression

- Example 7: What is the value of the **prefix expression**
  + – * 2 3 5 / ↑ 2 3 4 ?

- **Solution:** The steps used to evaluate this expression by **working right to left**, and **performing operations using the operands on the right**, are shown in Figure 12.

  The value of this expression is **3**.

Evaluating a Postfix Expression

- In an postfix expression, a binary operator follows its two operands
- The expression is evaluated **from left to right**
- Look for the **first operator** from the left
- Perform the corresponding operation with that operator with the two operands immediately to its left

Evaluating a Postfix Expression

- **Example 9:** What is the **value** of the **postfix** expression **7 2 3 * – 4 ↑ 9 3 / +** ?

- **Solution:** The steps used to evaluate this expression by **starting at the left** and **carrying out operations when two operands are followed by an operator** are shown in Figure 13.

  The value of this expression is **4**.

Evaluating a Prefix Expression

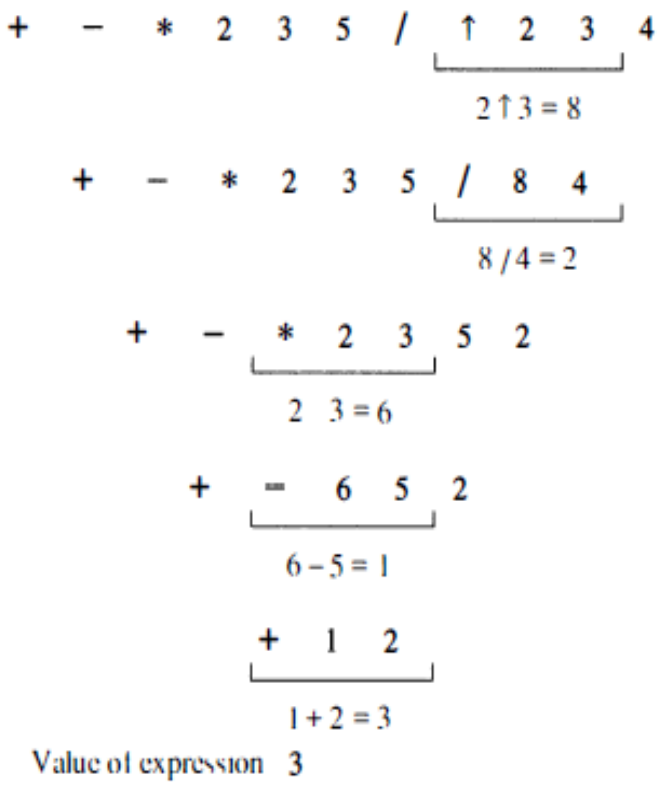


FIGURE 12 Evaluating a Prefix Expression.

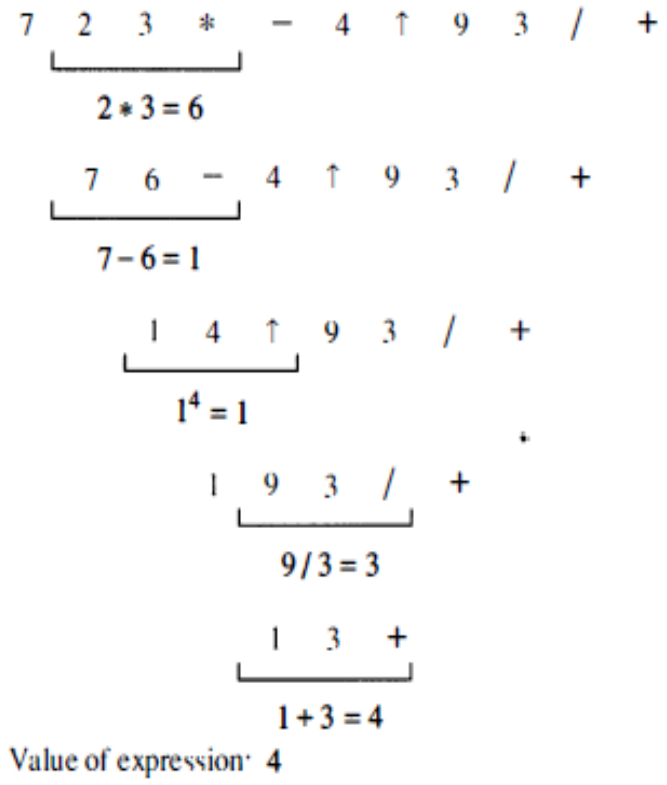


FIGURE 13 Evaluating a Postfix Expression.

Class Work

- What is the value of the **prefix expression**

  + / + 2 2 2 / – 3 2 +1 0 ?

Solution:

+ / + 2 2 2 / – 3 2 + 1 0

+ / + 2 2 2 / – 3 2 1

+ / + 2 2 2 / 1 1

+ / + 2 2 2 1

+ / 4 2 1

+ 2 1

3

Class Work

- What is the value of the **postfix expression**

  2 2 + 2 / 3 2 – 1 0 + / + ?

Solution:

2 2 + 2 / 3 2 – 1 0 + / +

4 2 / 3 2 – 1 0 + / +

2 3 2 – 1 0 + / +

2 1 1 0 + / +

2 1 1 / +

2 1 +

3

Practice @ Home

- **Relevant odd-numbered exercises**

- **Exercises: 7, 9 , 11, 13, 15, 17, 23, 25**

Books

- Rosen, K. H., & Krithivasan, K. (2012). Discrete mathematics and its applications: with combinatorics and graph theory. Tata McGraw-Hill Education. (7th Edition)

References

1. Discrete Mathematics, *Richard Johnsonbaugh*, Pearson education, Inc.
2. Discrete Mathematical Structures, *Bernard Kolman*, *Robert C. Busby*, *Sharon Ross,* Prentice-Hall, Inc.
3. *SCHAUM'S outlines Discrete Mathematics(2nd edition)*, by *Seymour Lipschutz*, *Marc Lipson*

- University of Hawaii
http://courses.ics.hawaii.edu/ReviewICS241/morea/trees/TreeTraversal-QA.pdf
- Florida State University
http://www.cs.fsu.edu/~lacher/lectures/Output/trees_intro/script.html