

Study of Finding Infix, Prefix, Postfix Using Graphical Notation

ER.RAHUL R. PAPALKAR

Assistant Professor
DES'S COET
DHAMANGAON RLY
Amravati University
Maharashtra (India)
rahulpapalkar@gmail.com

ER RAJKUMAR R YADAV

Assistant Professor
DES'S COET
DHAMANGAON RLY
Amravati University
Maharashtra (India)
rajyadav.engg@gmail.com

ABSTRACT -

There are several useful ways in which we can systematically order all nodes of a tree. The three most important orderings are called preorder, in order and post order. These all ordering having a specific algorithm, & this algorithm is complicated for keep in mind for student side ,so this paper gives solution to find systematic order of node using graphical notation, this is simplest way to memorize to find systematic order of binary tree.

Keyword: Tree, Nodes, Inorder, Preorder, PostOrder, DataStructure.

I. INTRODUCTION

In this paper, I present a unified approach to finding systematically order all nodes of a tree using graphical notation of binary tree, the three most important orderings are called preorder, in order and post order.

The tree is a fundamental structure in computer science. Almost all operating systems store files in trees or tree like structures. Trees are also used in compiler

design, text processing, and searching algorithms. A binary tree is an ordered tree in which each node has maximum of two children, referred to as a left and a right tree. A binary tree is different recursively as either empty or consists of a root, a left tree, and a right tree. The left and right trees may themselves be empty, thus a node with one child could have a left or a right child. Commonly, there are three Traversing methods: in order, pre order and post order traversals [1]

A tree is said to be ordered if the children of a node are ordered from left to right. A tree is directed if each child has a given position, e.g. when each child in a binary tree is either a left-child or a right child. The binary search tree is an example of a directed tree. Clearly, a directed tree is ordered [5].

A tree imposes a hierarchical structure on a collection of items. Familiar examples of trees are genealogies and organization charts. Trees are used to help analyze electrical circuits and to represent the

structure of mathematical formulas. Trees also arise naturally in many different areas of computer science. For example, trees are used to organize information in database systems and to represent the syntactic structure of source programs in compilers [2]

II. RELATED WORK & IMPLEMENTATION

Tree traversal techniques are:

- Pre-order
- In-order
- Post-order

I. In Order:

In this traversing order if we seen algorithm this is complicated to keep in mind for finding order, the algorithm is as follow [3]

- traverse the left sub tree,
- visit the root,
- traverse the right sub tree.

These step is easy but very difficult to keep in mind when we face complex binary tree & this is main challenge, don't worry I will give you a solution, we know that as compare to text data , we insolently remember graphical notation, that's why I have search easy procedure to find inorder traversal . as follows

➤ Using Algorithm finding In Order Traversal (Traditional Way)

- traverse the left sub tree,
- visit the root,
- traverse the right sub tree.

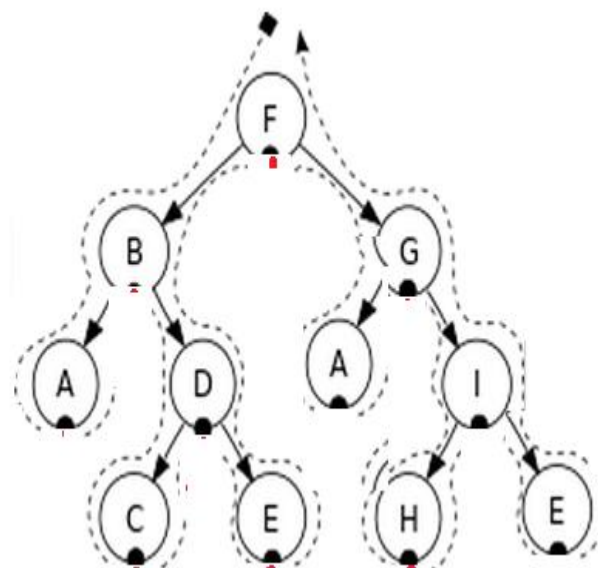
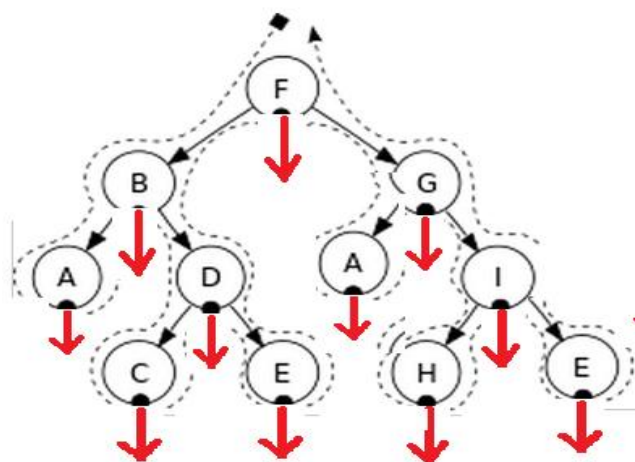


Fig 1: InOrder Traversal Using Algorithm

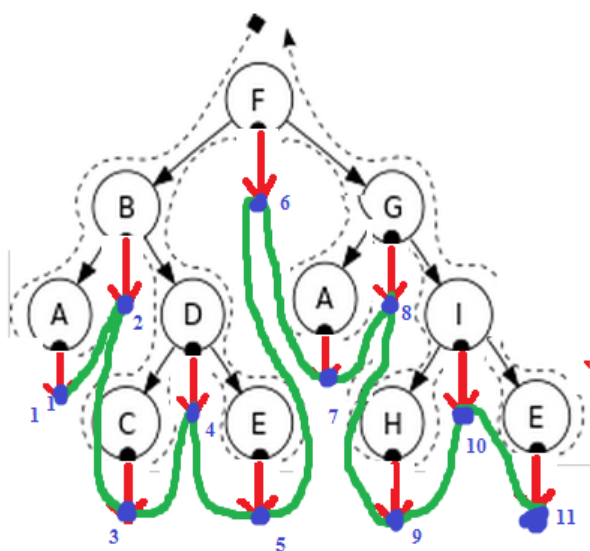
Output is ABCDEFAGHIE

➤ Proposed Method to find In Order Traversal :

Step 1: Draw Tree & Give Notation (↓) To All Nodes of binary tree



Step 2: In this Step Combine all arrow notation start from 1st left side node



In this Figure Start connecting arrow according to ascending order of number like (1, 2, 3, 4...n)

So final **output is :ABCDEFAGHIE**

Same Output obtained with simplest way. & this is beauty of this paper.

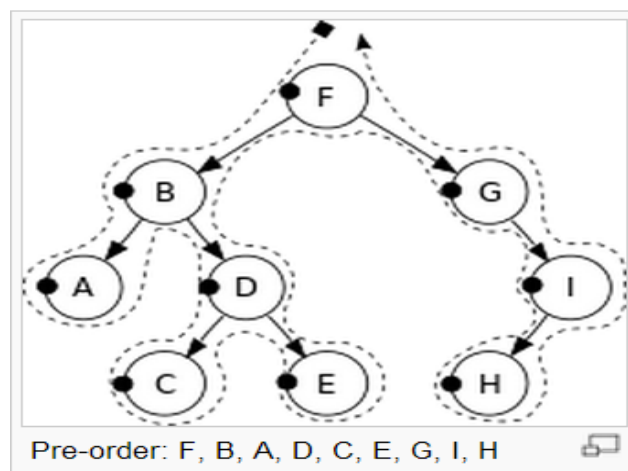
II. Preorder:-

In this traversing order if we seen algorithm this is complicated to keep in mind for finding order, the algorithm is as follow [4]

➤ Traditional Way To find Pre order Traversal

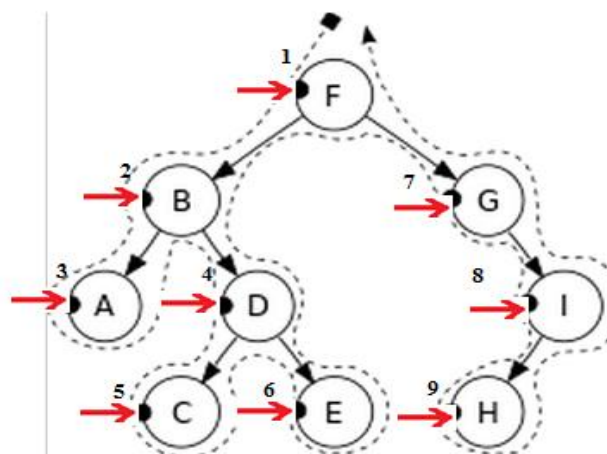
- visit the root,
- visit the left tree,
- visit the right sub tree

According to this algorithm the output of the below tree is as follow



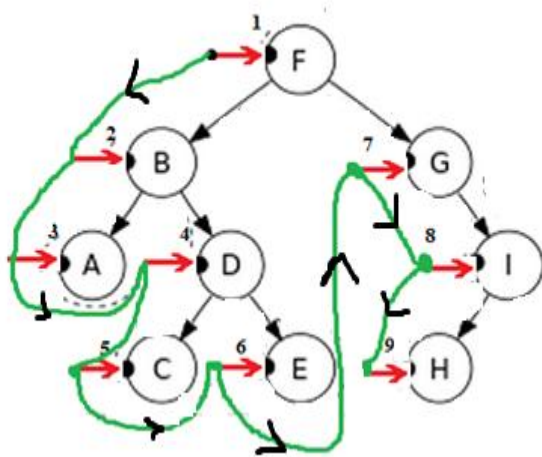
➤ Proposed Method to find Pre Order Traversal

Step 1: Draw Tree & Give Notation () To All Nodes of binary tree



In this tree add  notation to all vertices & apply sequence no from 1 to n for connecting.

Step 2: In this Step Combine all arrow notation start from 1st root node.



Connect all the notation according to ascending order of numbers like (1, 2, 3... n)

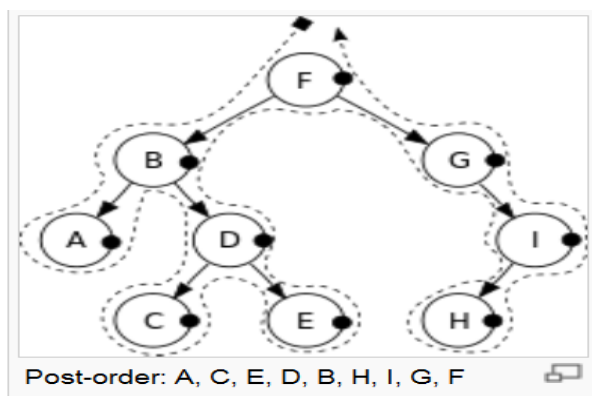
III. Post Order:

In this traversing order if we seen algorithm this is complicated to keep in mind for finding order, the algorithm is as follow

➤ Traditional Way To find Post order Traversal :

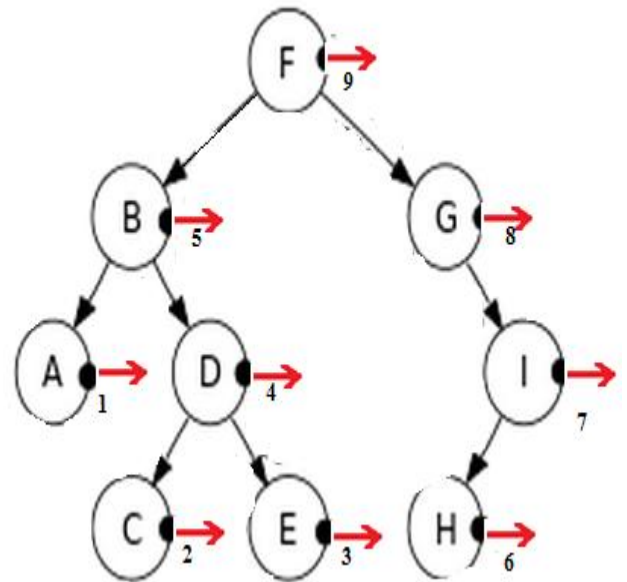
- visit the left,
- visit the root,
- visit the right sub tree

According to algorithm the output of the bellow tree is as follow

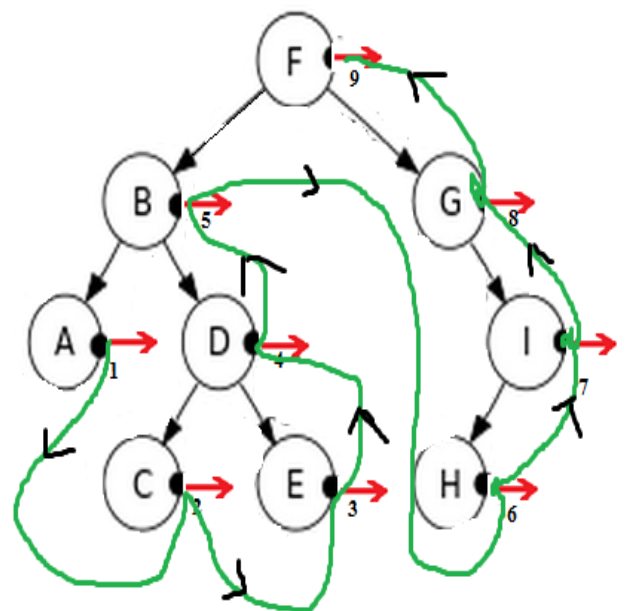


➤ Proposed Method to find Post Order Traversal

Step 1: Draw Tree & Give Notation () To All Nodes of binary tree



Step 2: In this Step Combine all arrow notation start from 1st root node



Connect all the notation according to ascending order of numbers like (1, 2, 3... n)

Final output using graphical notation is: **ACEDBHIGF**

III. CONCLUSION:

This Paper is useful for all students those who want to find infix, postfix, prefix in quick amount of time. This graphical notation method is very effective to memorize as compare to traditional method, because in that we need to memorize complex algorithm & that's why this graphical notation method is useful in mostly competitive examination like GET etc. because it gives fast output.

IV. REFERENCE

- [1] manoj c. lohani, upendra s. aswal and ramesh s. rawat department of computer science, graphic era university, dehradun (india). (received: march 10, 2011; accepted: april 18, 2011)
- [2] Alfred V. Aho, Bell Laboratories, Murray Hill, New Jersey John E. Hopcroft, Cornell University, Ithaca, New York
Jeffrey D. Ullman, Stanford University, Stanford, California
1.7.2001 <http://www.ourstillwaters.org>
- [3] Barnett-2006: Andy E. Barnett, Paper AP16, DIFFTREE: A Macro to Compare Corresponding Files under Two Directory Trees, PharmaSUG 2006
- [4] Alois Panholzer and Helmut Prodinger Discrete Mathematics and Theoretical Computer Science 1, 1997, 247–266
- [5] Arne Anderson and Svante Carlsson Department of Computer Science Lund University, Sweden PREPRINT. Information Processing Letters, 34(1):21-25, 1990.