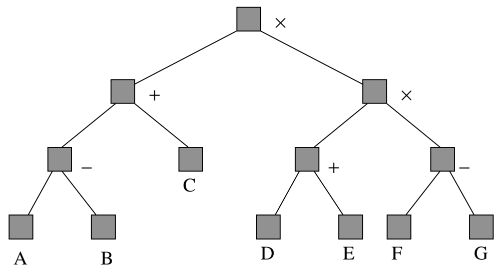
Tree Traversals

- To perform a traversal of a data structure, we use a method of visiting every node in some predetermined order
- Traversals can be used
  - to test data structures for equality
  - to display a data structure
  - to construct a data structure of a give size
  - to copy a data structure

Depth-First Traversals

- There are 3 depth-first traversals
  - inorder
  - postorder
  - preorder
- For example, consider the expression tree:

Example: Expression Tree



Depth-First Traversals

- Inorder traversal (LVR - Left, Visit root, Right)

  A – B + C × D + E × F – G

- Postorder traversal (LRV - Left, Right, Visit root)

  A B – C + D E + F G – × ×

- Preorder traversal (VLR - Visit root, Left, Right)

  × + – A B C × + D E – F G

Depth-First Traversals

- The parenthesised Inorder traversal

  ((A – B) + C) × ((D + E) × (F – G))

  This is the infix expression corresponding to the expression tree

- Postorder traversal gives a postfix expression
- Preorder traversal gives a prefix expression

Depth-First Traversals

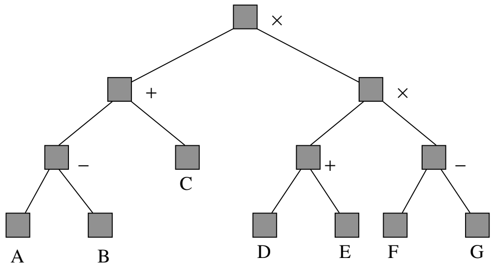
- Recursive definition of inorder traversal
- LVR - Left, Visit root, Right

  Given a binary tree *T*

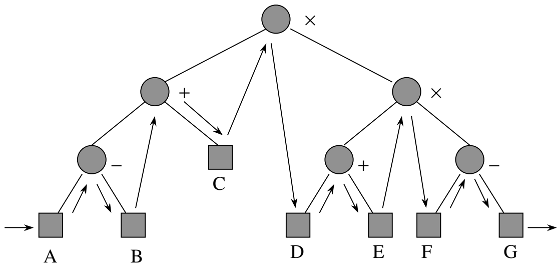
  if *T* is not empty
  - perform an inorder traversal of *Left*(*T*)
  - visit the root of *T*
  - perform an inorder traversal of *Right*(*T*)

e.g. Print binary tree in ascending order

Example: Inorder Traversal



Example: Inorder Traversal

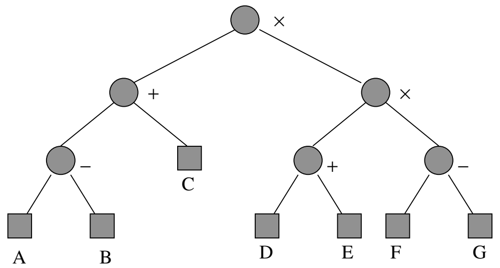


Depth-First Traversals

- Recursive definition of postorder traversal
- LRV - Left, Right, Visit root

Given a binary tree *T*
if *T* is not empty
perform an postorder traversal of *Left*(*T*)
perform an postorder traversal of *Right*(*T*)
visit the root of *T*

Example: Postorder Traversal

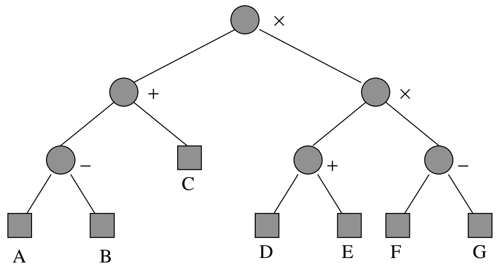


Depth-First Traversals

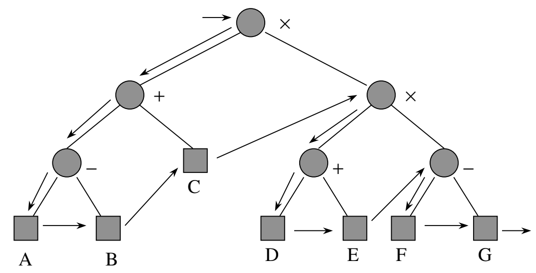
- Recursive definition of preorder traversal
- VLR - Visit root, Left, Right

Given a binary tree *T*
if *T* is not empty
visit the root of *T*
perform an preorder traversal of *Left*(*T*)
perform pre traversal of *Right*(*T*)

Example: Preorder Traversal



Example: Preorder Traversal



Example of inorder traversal
Print binary tree in ascending order

```
public void inorder(TreeNode tNode) {
    if(tNode !=null)
      {
        inorder(tNode.getLeft());

        KeyedItem nodeItem = (KeyedItem)tNode.getItem();
        System.out.println(nodeItem.getKey());

        inorder(tNode.getRight());
    }
}
```