



Dr. D. Y. PATIL  
EDUCATIONAL FEDERATION

## Assignment No-5.

Title - Tree - TBT.

Objectives -

- To understand how to make inorder traversal without using stack and recursion.

• Problem Statement.

Convert given binary tree into threaded binary tree.

• Outcome -

Students will be able to analyze advantages of TBT.

Software and hardware Requirements -

Operating System recommended - 64 bit Open

Source linux or its derivative

programming tools recommended : c++ programming tool like G++/GCC

Theory

Explain detail what is TBT with ex?

- A TBT stands for threaded binary tree

- A TBT is a binary tree in which every null pointer is replaced with a pointer to its inorder successor or predecessor

- There are 2 types in a threaded binary tree.



Dr. D. Y. PATIL  
EDUCATIONAL FEDERATION

1) Single - Threaded Binary Tree - each node points to its inorder successor

2) Double - Threaded Binary Tree - each node points to both its inorder successor and predecessor.

\* properties

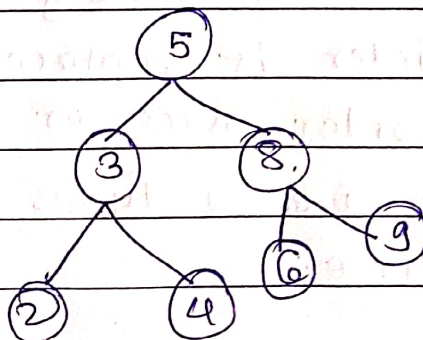
→ In a single-threaded tree, a node with no right child points to its inorder successor and a node with no left child points to its inorder predecessor.

Advantages of TBT.

- Threaded Binary Tree eliminates the need for stack space during traversal.
- Inorder traversal can be done without using recursion or a stack, reducing time and space complexity.

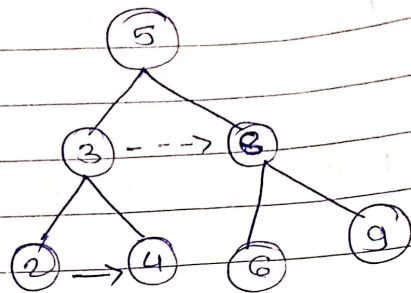
ex.

Binary Tree





After making it a single-threaded Binary Tree.



Difference of a BT and TBT.

Binary Tree

Threaded Binary Tree

1) Binary Tree have a lot of wasted space

Threaded binary tree do not a wasted space,

2) The null pointers remain null.

Threaded Binary trees The null pointers are used as Thread.

3) we can't use null pointer so it is as wastage memory

we can use the null pointers which is a efficient way to use computers memory





Dr. D. Y. PATIL  
EDUCATIONAL FEDERATION

4)	Traverse is not easy and not memory efficient	Traverse is easy. completed without using stack or recursive function.
----	---	--

5)	Less complex than Threaded binary Tree.	Structure is complex
----	---	----------------------

#### Algorithm

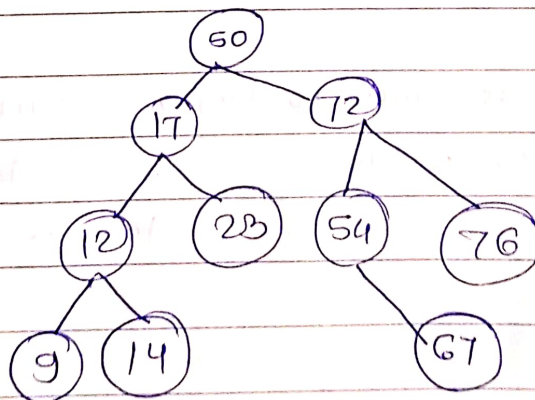
- postorder Algorithm.

Step 1 - First, we traverse the left subelement repeatedly

Step 2 - Now, in the second stage, we traverse the right subelement repeatedly

Step 3 - visit the root node.

ex.



(5) (8)



Dr. D. Y. PATIL  
EDUCATIONAL FEDERATION

IL  
ION

Postorder - 9, 14, 12, 19, 23, 17, 67, 54, 76, 72, 50.

Conclusion - Thus, we have studied and implemented TBT and understood how make inorder traversal faster without using stack and recursion.