



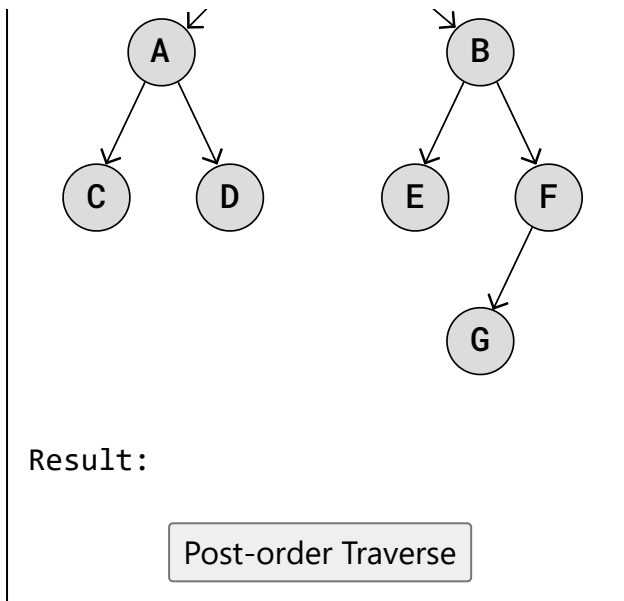
DSA Post-order Traversal

[< Previous](#)[Next >](#)

Post-order Traversal of Binary Trees

Post-order Traversal is a type of Depth First Search, where each node is visited in a certain order. Read more about Binary Tree traversals in general [here](#).

Doing a Post-order Traversal on a Binary Tree can be visualized like this:



Post-order Traversal works by recursively doing a Post-order Traversal of the left subtree and the right subtree, followed by a visit to the root node. It is used for deleting a tree, post-fix notation of an expression tree, etc.

What makes this traversal "post" is that visiting a node is done "after" the left and right child nodes are called recursively.

This is how the code for Post-order Traversal looks like:

Example

Python:

```
1 def postOrderTraversal(node):
2     if node is None:
3         return
4     postOrderTraversal(node.left)
5     postOrderTraversal(node.right)
6     print(node.data, end=", ")
```

[Try it Yourself »](#)

[Tutorials ▼](#)[References ▼](#)[Exercises ▼](#)[Sign In](#)[SS](#) [JAVASCRIPT](#) [SQL](#) [PYTHON](#) [JAVA](#) [PHP](#) [HOW TO](#) [W3.CSS](#) [C](#) [C](#)

and then the letter 'C' is printed (line 6).

This means that C is visited, or printed, "after" its left and right child nodes are traversed, that is why it is called "post" order traversal.

The `postOrderTraversal()` function continues to propagate back to previous recursive function calls, so the next node to be printed is 'D', then 'A'.

The function continues to propagate back and printing nodes until all nodes are printed, or visited.

[< Previous](#)[Sign in to track progress](#)[Next >](#)

Full access

All Courses
All Certificates

\$499 Save 75%
~~\$1,995~~

COLOR PICKER



[Tutorials ▼](#)[References ▼](#)[Exercises ▼](#)[Sign In](#)[SS](#)[JAVASCRIPT](#)[SQL](#)[PYTHON](#)[JAVA](#)[PHP](#)[HOW TO](#)[W3.CSS](#)[C](#)[C](#)[PLUS](#)[SPACES](#)[GET CERTIFIED](#)[FOR TEACHERS](#)[FOR BUSINESS](#)[CONTACT US](#)

Top Tutorials

- [HTML Tutorial](#)
- [CSS Tutorial](#)
- [JavaScript Tutorial](#)
- [How To Tutorial](#)
- [SQL Tutorial](#)
- [Python Tutorial](#)
- [W3.CSS Tutorial](#)
- [Bootstrap Tutorial](#)
- [PHP Tutorial](#)
- [Java Tutorial](#)
- [C++ Tutorial](#)
- [jQuery Tutorial](#)

Top References

- [HTML Reference](#)
- [CSS Reference](#)
- [JavaScript Reference](#)
- [SQL Reference](#)
- [Python Reference](#)
- [W3.CSS Reference](#)
- [Bootstrap Reference](#)
- [PHP Reference](#)
- [HTML Colors](#)
- [Java Reference](#)
- [AngularJS Reference](#)
- [jQuery Reference](#)

Top Examples

- [HTML Examples](#)
- [CSS Examples](#)
- [JavaScript Examples](#)
- [How To Examples](#)
- [SQL Examples](#)
- [Python Examples](#)
- [W3.CSS Examples](#)
- [Bootstrap Examples](#)

Get Certified

- [HTML Certificate](#)
- [CSS Certificate](#)
- [JavaScript Certificate](#)
- [Front End Certificate](#)
- [SQL Certificate](#)
- [Python Certificate](#)
- [PHP Certificate](#)
- [jQuery Certificate](#)

[Tutorials ▼](#)[References ▼](#)[Exercises ▼](#)[Sign In](#)[≡](#) [SS](#) [JAVASCRIPT](#) [SQL](#) [PYTHON](#) [JAVA](#) [PHP](#) [HOW TO](#) [W3.CSS](#) [C](#) [C](#)[FORUM](#) [ABOUT](#) [ACADEMY](#)

W3Schools is optimized for learning and training. Examples might be simplified to improve reading and learning.

Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness

of all content. While using W3Schools, you agree to have read and accepted our [terms of use](#), [cookies](#) and [privacy policy](#).

[Copyright 1999-2026](#) by Refsnes Data. All Rights Reserved. [W3Schools is Powered by W3.CSS](#).