

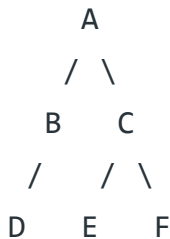


Related Articles

Breadth First Search

BFS stands for Breadth First Search is a vertex based technique for finding a shortest path in graph. It uses a Queue data structure which follows first in first out. In BFS, one vertex is selected at a time when it is visited and marked then its adjacent are visited and stored in the queue. It is slower than DFS.

Ex-



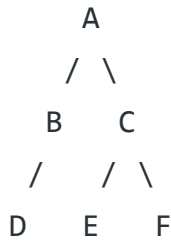
Output is:

A, B, C, D, E, F

Depth First Search

DFS stands for Depth First Search is a edge based technique. It uses the Stack data structure, performs two stages, first visited vertices are pushed into stack and second if there is no vertices then visited vertices are popped.

—



Output is:

A, B, D, C, E, F

BFS vs DFS

S.NO

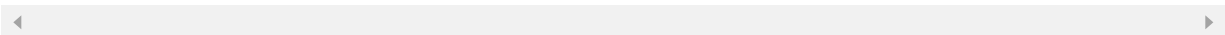
1. BFS stands for Breadth First Search.
2. BFS(Breadth First Search) uses Queue data structure for finding the shortest path.
3. BFS can be used to find single source shortest path in an unweighted graph because

DFS

- DFS stands for Depth First Search.
- DFS(Depth First Search) uses Stack data structure.
- In DFS, we might traverse through more edges to reach a destination vertex from a

with minimum number of edges from a source vertex.

- | | | |
|----|--|---|
| 3. | BFS is more suitable for searching vertices which are closer to the given source. | DFS is more suitable when there are solutions away from source. |
| 4. | BFS considers all neighbors first and therefore not suitable for decision making trees used in games or puzzles. | DFS is more suitable for game or puzzle problems. We make a decision, then explore all paths through this decision. And if this decision leads to win situation, we stop. |
| 5. | The Time complexity of BFS is $O(V + E)$ when Adjacency List is used and $O(V^2)$ when Adjacency Matrix is used, where V stands for vertices and E stands for edges. | The Time complexity of DFS is also $O(V + E)$ when Adjacency List is used and $O(V^2)$ when Adjacency Matrix is used, where V stands for vertices and E stands for edges. |



Please also see [BFS vs DFS for Binary Tree](#) for the differences for a Binary Tree Traversal.

Attention reader! Don't stop learning now. Get hold of all the important DSA concepts with the [DSA Self Paced Course](#) at a student-friendly price and become industry ready.

To complete your preparation from learning a language to DS Algo and many more

In case you wish to attend live classes with industry experts, please refer [Geeks Classes Live](#) and [Geeks Classes Live USA](#)

Like 0

Previous

Next

RECOMMENDED ARTICLES

Page : 1 2 3

01 **BFS vs DFS for Binary Tree**
22, Feb 16

05 **Breadth First Search or BFS for a Graph**
20, Mar 12

02 **Difference between Local File System (LFS) and Distributed File System (DFS)**
02, Jul 20

06 **0-1 BFS (Shortest Path in a Binary Weight Graph)**
23, Apr 17

03 **Minimum number of edges between two vertices of a graph using DFS**
29, Jan 19

07 **BFS for Disconnected Graph**
17, May 17

04 **Calculate number of nodes between two vertices in an acyclic Graph by DFS method**
08, Sep 20

08 **Print all paths from a given source to a destination using BFS**
12, Aug 17

Article Contributed By :



MKS075

@MKS075

Vote for difficulty

Current difficulty : [Easy](#)

Easy

Normal

Medium

Hard

Expert

Improved By : [MahammadRafeeShaik](#), [ashushrma378](#), [tusharsingh2](#)

Article Tags : [BFS](#), [DFS](#), [Algorithms](#), [Difference Between](#)

Practice Tags : [DFS](#), [BFS](#), [Algorithms](#)

Improve Article

Report Issue

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments



5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

Got It !

Company

About Us
Careers
Privacy Policy
Contact Us
Copyright Policy

Practice

Courses
Company-wise
Topic-wise
How to begin?

Learn

Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

Contribute

Write an Article
Write Interview Experience
Internships
Videos

@geeksforgeeks , Some rights reserved