# **Breadth First Search or BFS for a Graph**

* Difficulty Level : [Easy](https://www.geeksforgeeks.org/easy/)
* Last Updated : 27 Jun, 2022

[Breadth-First Traversal (or Search)](http://en.wikipedia.org/wiki/Breadth-first_search) for a graph is similar to Breadth-First Traversal of a tree (See method 2 of [this post](https://www.geeksforgeeks.org/level-order-tree-traversal/)). The only catch here is, that, unlike trees, graphs may contain cycles, so we may come to the same node again. To avoid processing a node more than once, we use a boolean visited array. For simplicity, it is assumed that all vertices are reachable from the starting vertex. BFS uses a queue data structure for traversal.

For example, in the following graph, we start traversal from vertex 2. When we come to vertex 0, we look for all adjacent vertices of it. 2 is also an adjacent vertex of 0. If we don’t mark visited vertices, then 2 will be processed again and it will become a non-terminating process.



[Click here for the Complete Course!](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?vC=1)

[](https://practice.geeksforgeeks.org/courses/complete-interview-preparation?vC=1)

There can be multiple BFS traversals for a graph. Different BFS traversals for the above graph :

2, 3, 0, 1

2, 0, 3, 1