**Queue and BFS**[Report Issue](https://github.com/LeetCode-Feedback/LeetCode-Feedback/issues)

One common application of Breadth-first Search (BFS) is to find the shortest path from the root node to the target node. In this article, we provide an example to explain how queue is applied in a BFS algorithm step by step.

An Example

Here we provide an example to show how BFS is used to find the shortest path between the root node A and the target node G.

A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

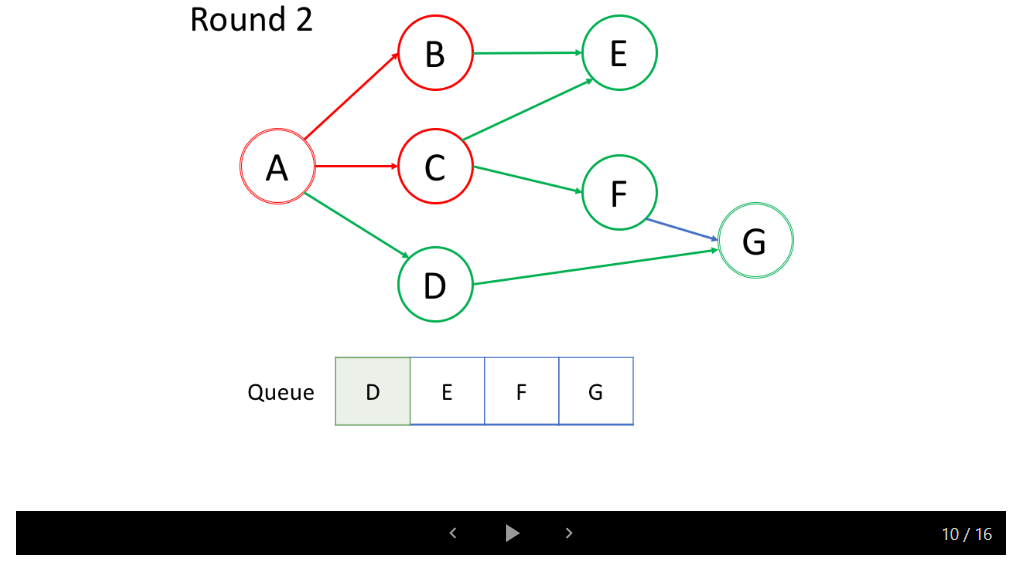
AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.  A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1 / 16

Insights

After watching the animation above, let's answer the following questions:

**1. What is the processing order of the nodes?**

In the first round, we process the root node. In the second round, we process the nodes next to the root node; in the third round, we process the nodes which are two steps from the root node; so on and so forth.

Similar to tree's level-order traversal, the nodes closer to the root node will be traversed earlier.

If a node X is added to the queue in the kth round, the length of the shortest path between the root node and X is exactly k. That is to say, you are already in the shortest path the first time you find the target node.

**2. What is the enqueue and dequeue order of the queue?**

As shown in the animation above, we first enqueue the root node. Then in each round, we process the nodes which are already in the queue one by one and add all their neighbors to the queue. It is worth noting that the newly-added nodes will not be traversed immediately but will be processed in the next round.

The processing order of the nodes is the exact same order as how they were added to the queue, which is First-in-First-out (FIFO). That's why we use a queue in BFS.