# Breadth-First Search (BFS) using Queue

This code performs Breadth-First Search (BFS) traversal on a graph using a Queue data structure.  
It explores all nodes level by level starting from the root node.

## Explanation:

1. The Queue class is defined with two main methods:  
 - enqueue(): adds an element to the queue.  
 - dequeue(): removes the first element from the queue.  
   
2. The graph is represented as a dictionary, where keys are nodes and values are lists of adjacent nodes.  
  
3. The algorithm:  
 - Starts at node 'A' and enqueues it.  
 - Dequeues a node, visits it, and enqueues all its unvisited neighbors.  
 - Repeats this process until the queue is empty or the goal node is found.  
  
4. If the goal node is reached, it prints the traversal path up to that point.  
 If not, it prints “Error.”  
  
BFS uses the \*\*FIFO (First In, First Out)\*\* principle, meaning it explores closer nodes first before going deeper.

## Example Output:

Enter your Goal : H  
Output: ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H']