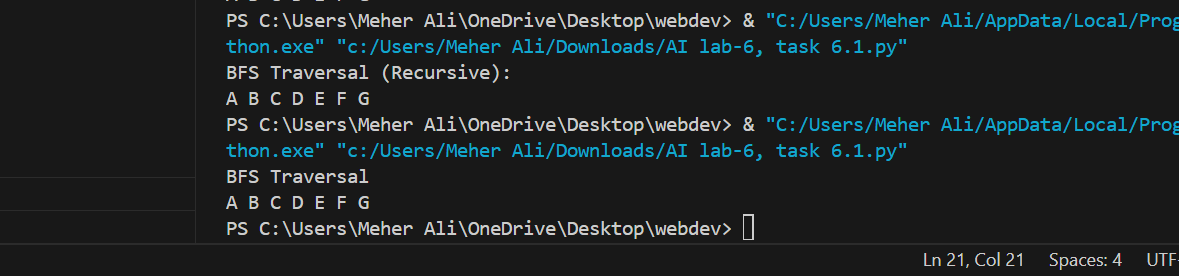
**LAB-6**

**TASK.1**

The given program implements recursive Breadth-First Search (BFS) traversal on a graph represented as an adjacency list (dictionary). The function bfs\_recursive() takes three parameters: the graph, the current level of nodes (level\_nodes), and a visited set. It prints nodes level by level, adding unvisited neighbors to the next level and recursively calling itself until no nodes remain. The program defines a sample graph and initiates BFS from node "A", printing the traversal order.



**TASK.2**

The program implements a graph data structure using the Graph and Node classes and performs Breadth-First Search (BFS) using a queue. The Node class represents individual graph nodes, storing a value and a list of neighbors. The Graph class allows adding nodes (add\_node()) and edges (add\_edge()). The bfs\_with\_queue() method performs BFS by processing nodes level by level, using a queue to track unvisited neighbors. The program constructs a sample graph and initiates BFS traversal from node "A", printing the nodes in the order they are visited.

