Date 02/03/2024

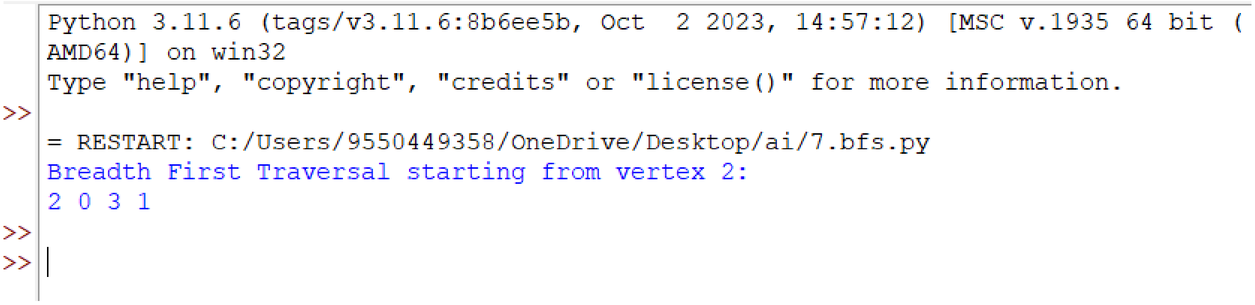
7.Aim: Write the python program to implement BFS.

Program:

```
from collections import defaultdict, deque


class Graph:
    def __init__(self):
        self.graph = defaultdict(list)


    def add_edge(self, u, v):
        self.graph[u].append(v)


    def bfs(self, start):
        visited = [False] * (max(self.graph) + 1)
        queue = deque()


        queue.append(start)
        visited[start] = True


        while queue:
            node = queue.popleft()
            print(node, end=" ")


            for neighbor in self.graph[node]:
                if not visited[neighbor]:
                    queue.append(neighbor)
                    visited[neighbor] = True


# Example usage:
```

```
if __name__ == "__main__":
    graph = Graph()
    graph.add_edge(0, 1)
    graph.add_edge(0, 2)
    graph.add_edge(1, 2)
    graph.add_edge(2, 0)
    graph.add_edge(2, 3)
    graph.add_edge(3, 3)

    print("Breadth First Traversal starting from vertex 2:")
    graph.bfs(2)
```

Output:



```
   Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct  2 2023, 14:57:12) [MSC v.1935 64 bit (
   AMD64)] on win32
   Type "help", "copyright", "credits" or "license()" for more information.
>>
   = RESTART: C:/Users/9550449358/OneDrive/Desktop/ai/7.bfs.py
   Breadth First Traversal starting from vertex 2:
   2 0 3 1
>>
>>
```

Result: The given program has been executed successfully