```python
from collections import deque

def bfs(graph, start):
    visited = []                # List to keep track of visited nodes
    queue = deque([start])      # Initialize queue with the starting node

    print("BFS Traversal Order:", end=" ")

    while queue:
        node = queue.popleft()  # Remove the first node from queue
        if node not in visited:
            print(node, end=" ")
            visited.append(node)

            # Add unvisited neighbors to the queue
            for neighbour in graph[node]:
                if neighbour not in visited:
                    queue.append(neighbour)
    print()

# Example Graph (Adjacency List)
graph = {
    'A': ['B', 'C'],
    'B': ['D', 'E'],
    'C': ['F'],
    'D': [],
    'E': ['F'],
    'F': []
}

# Start BFS from node 'A'
bfs(graph, 'A')
```

BFS Traversal Order: A B C D E F