## Code implementation:

**Answer:**

```
tree = {
    'A': ['B', 'C'],
    'B': ['D', 'E'],
    'C': ['F', 'G'],
    'D': [],
    'E': [],
    'F': [],
    'G': [],
}

from collections import deque

def bfs(graph, start, goal):


    queue = deque([start])
    visited = set([start])
    parent = {}

    while queue:
        node = queue.popleft()

        if node == goal:
            return reconstruct_path(parent, start, goal)

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

    return None
```