Q8. Traverse a general tree (rooted) in all siblings together (bfs) manner and write the procedure for it. Discuss the space complexity of tree representation first.

We are assuming that the structure of the node of the tree is as follows.

```
Node:
    data: data present in the node
    children: a dynamic array which stores the links of child nodes of this node
```

Since, every node is storing only the data and the links to its children, the total space used for representing the tree will be n + e, where

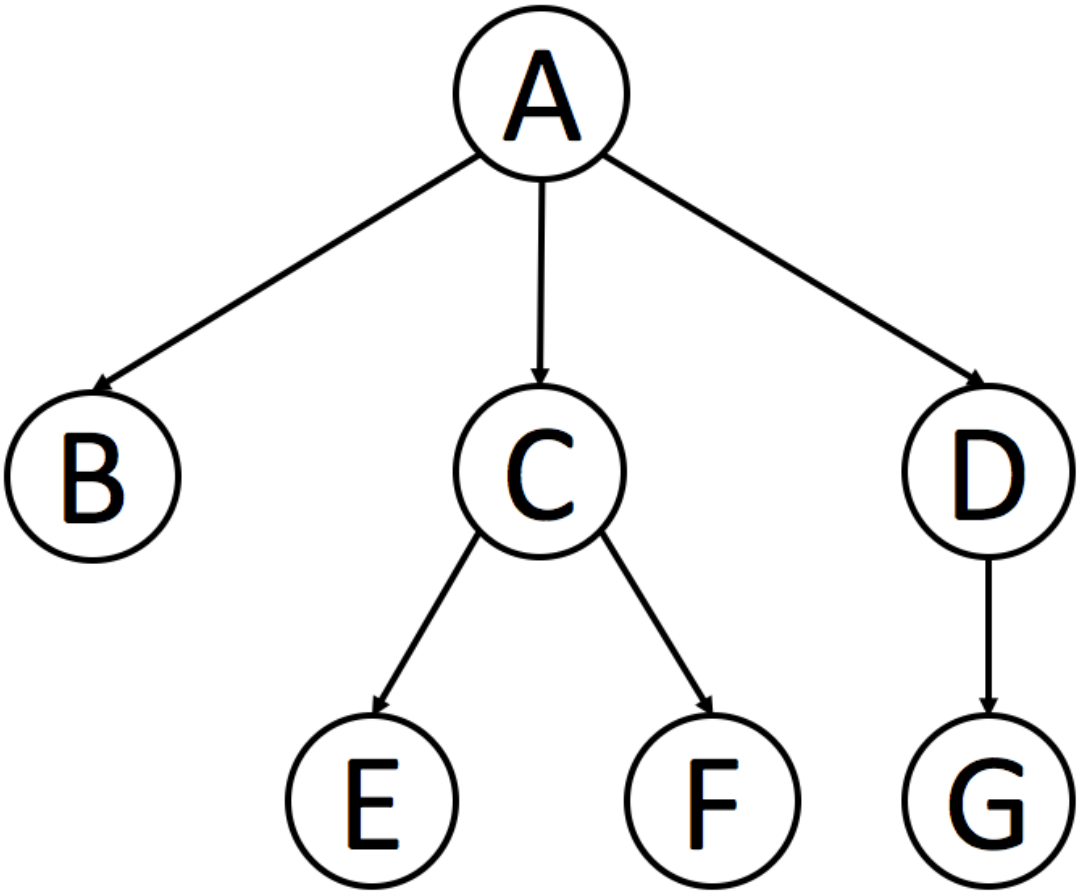
- n: number of nodes
- e: number of edges

Also, in a tree, e = n - 1. Hence the space complexity will be O(n) which is also equal to O(e).

The pseudo code for traversing the tree in BFS manner is as follows.

```
# root: Root of the tree

BFS(root):
    IF root == NULL:
        RETURN

    Q = Create new Queue
    Q.enqueue(root)

    WHILE Q IS NOT EMPTY:
        top = Q.dequeue()
        PRINT top.data

        IF top.children IS NOT EMPTY:
            FOR child in top.children:
                Q.enqueue(child)
```

Example: Let the given general tree be as follows.



The output of BFS traversal will be as follows.

```
A
B
C
D
E
F
G
```