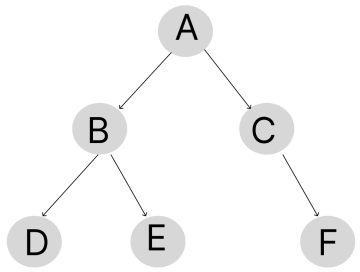
# IMPLEMENTATION OF BFS

In [ ]:
```
graph ={
    'A':['B','C'],
    'B':['D','E'],
    'C':['F'],
    'D':[],
    'E':[],
    'F':[]
}
```

In [ ]:
```
from IPython.display import Image
Image(filename="Graph01.png" ,width=200,height=200)
```

Out[ ]:



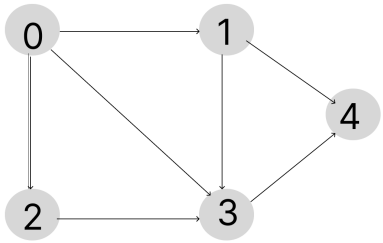

In [ ]:
```
from collections import deque

def bfs(graph, start):
    visited = set()
    queue = deque([start])
    result = []

    while queue:
        node = queue.popleft()
        if node not in visited:
            visited.add(node)
            result.append(node)
            queue.extend(graph[node])

    return result


print("BFS:", bfs(graph, 'A'))
```

```
BFS: ['A', 'B', 'C', 'D', 'E', 'F']
```

In [ ]:
```
Graph = {
    0: [1, 2, 3],
    1: [4, 3],
    2: [3],
    3: [4],
    4: []
}
```

In [ ]:
```
from IPython.display import Image
Image(filename="Graph02.png" ,width=200,height=200)
```

Out[ ]:




In [ ]:
```
from collections import deque

def bfs(graph, start):
    visited = set()
    queue = deque([start])
    result = []

    while queue:
        node = queue.popleft()
        if node not in visited:
            visited.add(node)
            result.append(node)
            queue.extend(graph[node])

    return result



print("BFS:", bfs(Graph, 0))
```

```
BFS: [0, 1, 2, 3, 4]
```

# AI LAB :[PC-CS(AM)593]

## SUPRATIM NAG_AIML/22/57