In [1]:

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

visited = []
queue = []
visited_dfs = set()
def bfs(visited, graph, node):
    visited.append(node)
    queue.append(node)

    while queue:
        m = queue.pop(0)
        print (m, end = " ")

        for neighbour in graph[m]:
            if neighbour not in visited:
                visited.append(neighbour)
                queue.append(neighbour)

def dfs(visited_dfs, graph, node): #function for dfs
    if node not in visited_dfs:
        print (node)
        visited_dfs.add(node)

        for neighbour in graph[node]:
            dfs(visited_dfs, graph, neighbour)

print("Following is the Breadth-First Search")
bfs(visited, graph, 'A')

print("\nFollowing is the Depth-First Search")
dfs(visited_dfs, graph, 'A')
```

```
Following is the Breadth-First Search
A B C D E F
Following is the Depth-First Search
A
B
D
E
C
F
```

In [ ]: