In [4]:
```
graph = {
    '0': ['1','3','4'],
    '1': ['2'],
    '2': [],
    '3': ['5'],
    '4': ['5'],
    '5': []
}
vis=set()
def dfs(vis,graph,node):
    if node not in vis:
        print(node, end = " ")
        vis.add(node)
        for adj in graph[node]:
            dfs(vis,graph,adj)
print("Following is the Depth-First Search")
dfs(vis, graph, '0')


visited = []
queue = []

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)

print("\nFollowing is the Breadth-First Search")
bfs(visited, graph, '0')
```

```
Following is the Depth-First Search
0 1 2 3 5 4
Following is the Breadth-First Search
0 1 3 4 2 5
```