ARTIFICIAL INTELLIGENCE LAB EXPERIMENTs

PROGRAM 01 : Breadth First Search

```
graph = {
    'A' : ['B', 'C'],
    'B' : ['D'],
    'C' : ['E'],
    'D' : ['F'],
    'E' : ['F'],
    'F' : []
}
start = input("Enter the start node : ")

def breadthFirstSearch(graph):
    visited = []
    queue = [start]

    while queue:
        node = queue.pop(0)

        if node not in visited:
            visited.append(node)
            neighbours = graph[node]

            for neighbour in neighbours:
                queue.append(neighbour)
    return visited

print(f"BFS Traversal : {breadthFirstSearch(graph)}")
```

OUTPUT :

```
C:\Users\hayst\Documents\Coding-Programming\Artificial Intelligence
Enter the start node : A
BFS Traversal : ['A', 'B', 'C', 'D', 'E', 'F']
```

```
Enter the start node : C
BFS Traversal : ['C', 'E', 'F']
```

PROGRAM 02 : Goal Search

```
graph = {
    'A' : ['B', 'C'],
    'B' : ['D'],
    'C' : ['E'],
    'D' : ['F'],
    'E' : ['F'],
    'F' : []
}
start = input("Enter the start node : ")
goal = input("Enter the goal node  : ")

def goalSearch(graph):
    visited = []
    queue = [start]

    if start == goal:
        print("Start node itself is a goal Node")
        visited.append(start)
        return visited
    else:
        visited.append(start)
        while queue:
            node = queue.pop(0)
            for neighbour in graph[node]:
                if neighbour not in visited:
                    visited.append(neighbour)
                    queue.append(neighbour)

                if neighbour == goal:
                    return visited
        return "Connecting path doesnt exist !"

print(f"GS Traversal : {goalSearch(graph)}")
```

OUTPUT :

```
C:\Users\hayst\Documents\Coding-Programming\Artificial Intelligence\
Enter the start node : A
Enter the goal node  : A
Start node itself is a goal Node
GS Traversal : ['A']
```

```
Enter the start node : A
Enter the goal node  : D
GS Traversal : ['A', 'B', 'C', 'D']
```

```
Enter the start node : A
Enter the goal node  : H
GS Traversal : Connecting path doesnt exist !
```