Date:
EX: 02

Depth First search

Aim:

To implement Depth first search using python.

Algorithm:

1. start
2. Intialise graph as an adjacency list. (dictionary).
3. Define DFS function:
   - input: graph, node, visited set
4. check if the node is in the visited set.
   - if not, proceed to the next step
5. Mark the node as visited.
6. print the node.
7. Iterate through each neighbour of the node:
   - If the neighbour is not visited, recursively call DFS for the neighbor
8. Take the starting node as input from the user.
9. Initialize an empty set to keep track of visited nodes.
10. Call the DFS function with graph, starting node & visited set.
11. Stop.

Program:

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

def dfs(graph, node, visited):
    if node not in visited:
        print(node, end = ' ')
        visited.add(node)
        for neighbor in graph[node]:
            dfs(graph, neighbor, visited)

visited = set()
start_node = input("Enter a letter")
print("DFS starting from node", start_node, ":")
dfs(graph, start_node, visited).
```

output:

```
Enter a letter : A
DFS starting from node A :
A B D E F C
```

Result:

Python program for Depth first search is implemented successfully.