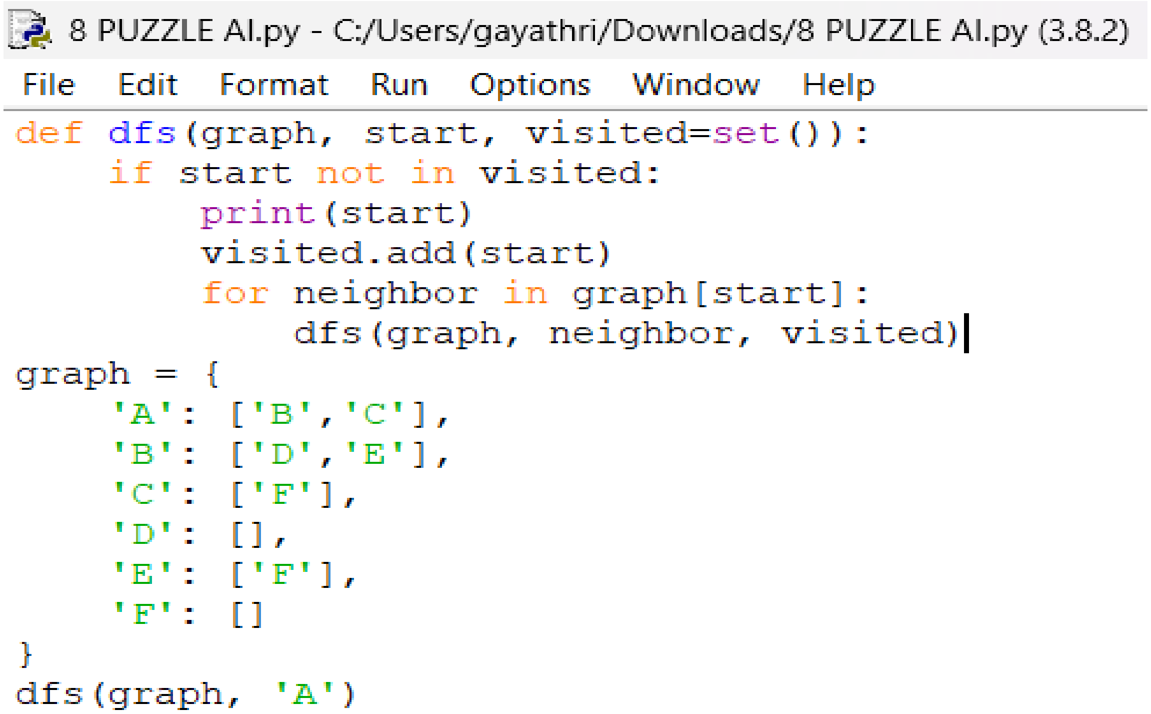
Write the python program to implement DFS.

AIM

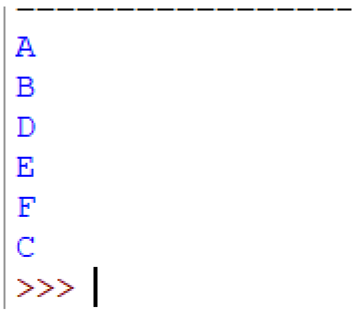
To implement **Depth-First Search (DFS)** algorithm in Python for traversing a graph starting from a given node.

ALGORITHM

1. Represent the graph using an adjacency list.
2. Initialize a **visited set** to keep track of visited nodes.
3. Start at the given start node.
4. If the current node is not visited:
    a. Print the node and mark it as visited.
    b. Recursively visit all unvisited neighbors of the current node.
5. Continue recursion until all reachable nodes are visited.


8 PUZZLE AI.py - C:/Users/gayathri/Downloads/8 PUZZLE AI.py (3.8.2)
File
Edit
Format
Run
Options
Window
Help

```python
def dfs(graph, start, visited=set()):
    if start not in visited:
        print(start)
        visited.add(start)
        for neighbor in graph[start]:
            dfs(graph, neighbor, visited)
graph = {
    'A': ['B','C'],
    'B': ['D','E'],
    'C': ['F'],
    'D': [],
    'E': ['F'],
    'F': []
}
dfs(graph, 'A')
```



```
------------------
A
B
D
E
F
C
>>> 
```

RESULT

The program successfully performed **Depth-First Search (DFS)** on the given graph and printed the nodes in **depth-wise order** starting from node 'A'.