DFS

Aim: The aim of Depth first search (DFS) is to explore all vertices of a graph systematically by going along each branch before backtracking

code:

```
def dfs(graph, start):
    stack = [start]
    visited = set()
    while stack:
        vertex = stack.pop()
        if vertex not in visited:
            print(vertex, end=" ")
            visited.add(vertex)
            for neighbor in graph[vertex]:
                if neighbor not in visited:
```

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

Algorithm

* Start at the source node
* Keep a track on visited not to avoid revisiting them
* Loop until the stack is empty.
* pop a node from the stack (Last in first out)
* Explore adjacent nodes
* if neighbor not visited push it onto the stack.
* Repeat until the stack is empty
* Stop.

Output:

```
dfs(graph, 'A')
```

A C F B E D.