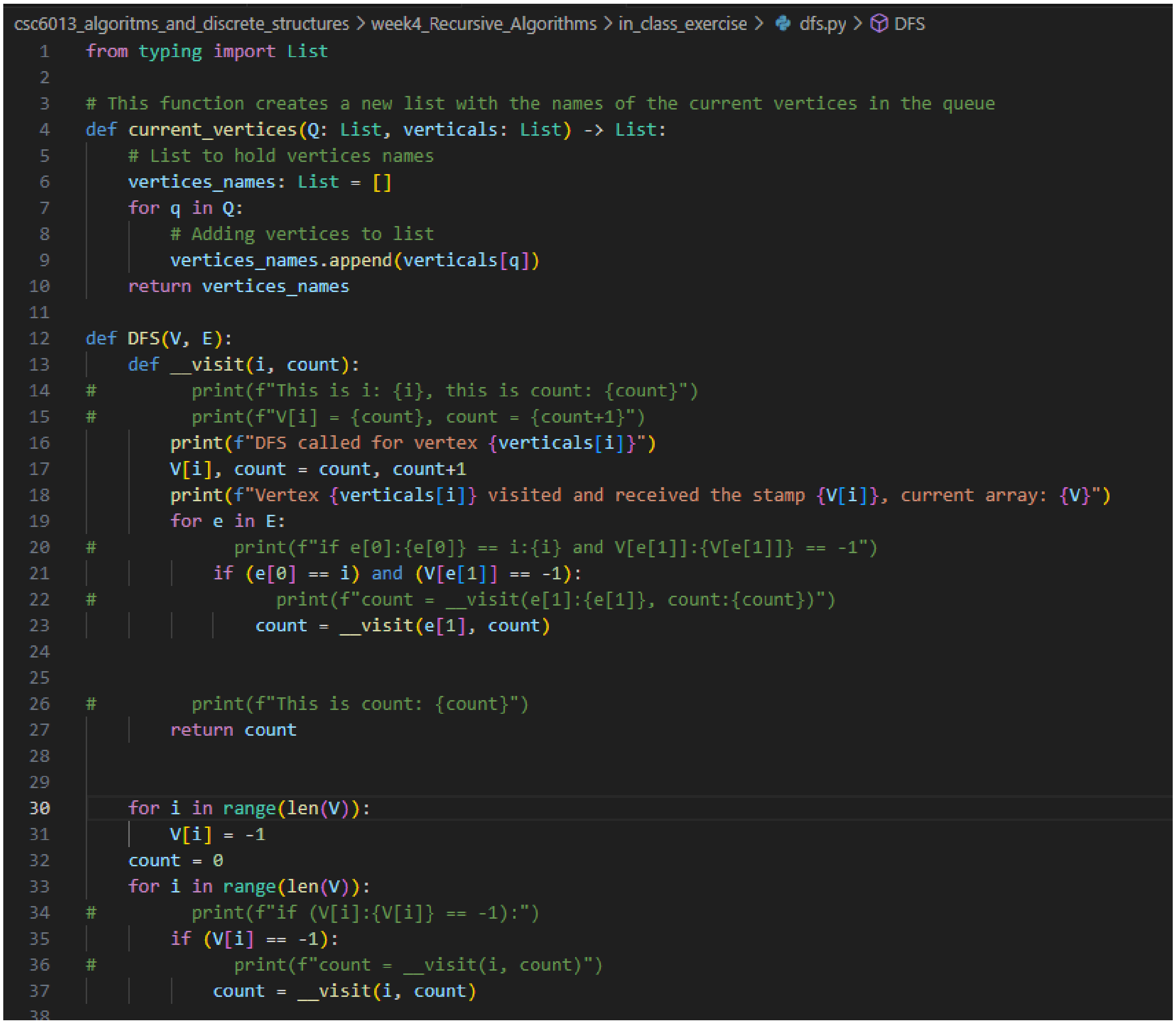

csc6013_algoritms_and_discrete_structures > week4_Recursive_Algorithms > in_class_exercise > dfs.py > DFS

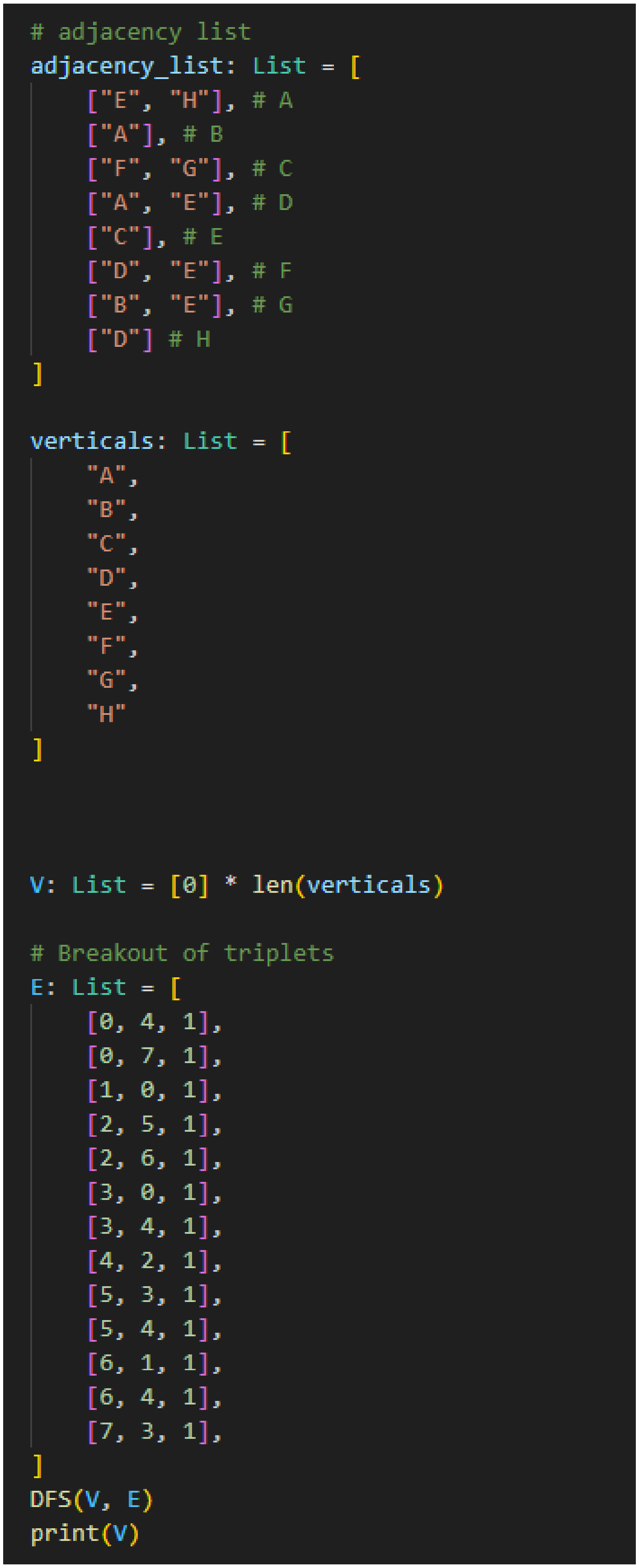
```python
from typing import List

# This function creates a new list with the names of the current vertices in the queue
def current_vertices(Q: List, verticals: List) -> List:
    # List to hold vertices names
    vertices_names: List = []
    for q in Q:
        # Adding vertices to list
        vertices_names.append(verticals[q])
    return vertices_names

def DFS(V, E):
    def __visit(i, count):
#         print(f"This is i: {i}, this is count: {count}")
#         print(f"V[i] = {count}, count = {count+1}")
        print(f"DFS called for vertex {verticals[i]}")
        V[i], count = count, count+1
        print(f"Vertex {verticals[i]} visited and received the stamp {V[i]}, current array: {V}")
        for e in E:
#             print(f"if e[0]:{e[0]} == i:{i} and V[e[1]]:{V[e[1]]} == -1")
            if (e[0] == i) and (V[e[1]] == -1):
#                 print(f"count = __visit(e[1]:{e[1]}, count:{count})")
                count = __visit(e[1], count)


#         print(f"This is count: {count}")
        return count


    for i in range(len(V)):
        V[i] = -1
    count = 0
    for i in range(len(V)):
#         print(f"if (V[i]:{V[i]} == -1):")
        if (V[i] == -1):
#             print(f"count = __visit(i, count)")
            count = __visit(i, count)
```



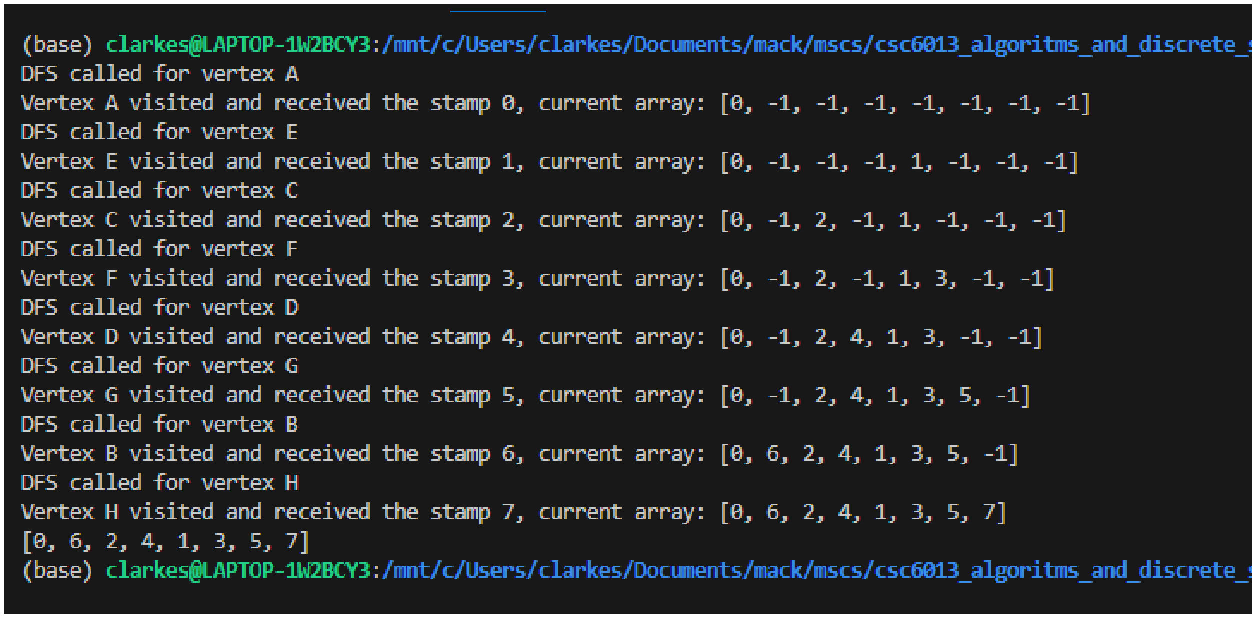
```python
# adjacency list
adjacency_list: List = [
    ["E", "H"], # A
    ["A"], # B
    ["F", "G"], # C
    ["A", "E"], # D
    ["C"], # E
    ["D", "E"], # F
    ["B", "E"], # G
    ["D"] # H
]

verticals: List = [
    "A",
    "B",
    "C",
    "D",
    "E",
    "F",
    "G",
    "H"
]



V: List = [0] * len(verticals)

# Breakout of triplets
E: List = [
    [0, 4, 1],
    [0, 7, 1],
    [1, 0, 1],
    [2, 5, 1],
    [2, 6, 1],
    [3, 0, 1],
    [3, 4, 1],
    [4, 2, 1],
    [5, 3, 1],
    [5, 4, 1],
    [6, 1, 1],
    [6, 4, 1],
    [7, 3, 1],
]
DFS(V, E)
print(V)
```



```
(base) clarkes@LAPTOP-1W2BCY3:/mnt/c/Users/clarkes/Documents/mack/mscs/csc6013_algoritms_and_discrete_
DFS called for vertex A
Vertex A visited and received the stamp 0, current array: [0, -1, -1, -1, -1, -1, -1, -1]
DFS called for vertex E
Vertex E visited and received the stamp 1, current array: [0, -1, -1, -1, 1, -1, -1, -1]
DFS called for vertex C
Vertex C visited and received the stamp 2, current array: [0, -1, 2, -1, 1, -1, -1, -1]
DFS called for vertex F
Vertex F visited and received the stamp 3, current array: [0, -1, 2, -1, 1, 3, -1, -1]
DFS called for vertex D
Vertex D visited and received the stamp 4, current array: [0, -1, 2, 4, 1, 3, -1, -1]
DFS called for vertex G
Vertex G visited and received the stamp 5, current array: [0, -1, 2, 4, 1, 3, 5, -1]
DFS called for vertex B
Vertex B visited and received the stamp 6, current array: [0, 6, 2, 4, 1, 3, 5, -1]
DFS called for vertex H
Vertex H visited and received the stamp 7, current array: [0, 6, 2, 4, 1, 3, 5, 7]
[0, 6, 2, 4, 1, 3, 5, 7]
(base) clarkes@LAPTOP-1W2BCY3:/mnt/c/Users/clarkes/Documents/mack/mscs/csc6013_algoritms_and_discrete_
```