Name: Saleem Talha

Section: BSCS-VII-C

Roll Number: 210973

DFS & BFS implementation in the Social Network Graph

Introduction

This is the basic explanation and report for our first assignment in which we had to implement **DFS** via recursion and **BFS** via iteration, we were told to take at least 6 nodes and implement both algorithms in python

For the nodes I have chosen myself and 5 of my friends and I have defined the relations (edges) as well

The program gives you a menu from where you can add people, make connections, apply bfs and dfs with starting point of your choosing and see the network as well

Graph Representation

Graph representation is done by using adjacency list. We had a choice between adjacency list and matrixes and I chose adjacency list

Algorithm implementation

Depth first search (DFS)

DFS is implemented recursively with the use of stack

1. Visited set preserves node that have been explored

2. This recursive process will expand a node, set it with the visits flag and thus visit all unvisited nodes connected to the node.

3. Still, the order in which objects are traversed is stored in the result list

Breadth-First Search (BFS)

BFS is implemented iteratively using a queue (specifically, a `deque` for built in function usage)

1. A visited set contain nodes that has been visit in a set of nodes of visited set

2. A queue regulates how the nodes are to be visited.

3. A path of the algorithm is to look through nodes on a level-by-level basis, pushing unvisited neighbors to the queue.

4. The traversal order is stored in a result list that was implemented as a dynamic array.

Conclusion

This implementation effectively uses graph theory principles in order to establish a basic social network analysis application. The problem shows how DFS and BFS can be applied in a real-life situation when selecting an appropriate data structure and algorithm. This simplicity makes it possible to test and experiment with the implemented algorithms using the interactive interface.

Outputs

Menu

```
--- Social Network Analysis ---
1. Add Person
2. Add Connection
3. Run DFS
4. Run BFS
5. Print Network
6. Exit
```

Nodes (People) and Edges (Relations)

```
--- Social Network Analysis ---
1. Add Person
2. Add Connection
3. Run DFS
4. Run BFS
5. Print Network
6. Exit
Enter your choice (1-6): 5
Talha: Anila, Annie, Saad
Anila: Mubashir, Annie
Saad:
Sadia: Saad, Anila
Annie: Talha, Saad
Mubashir: Talha, Anila, Saad
```

DFS

```
--- Social Network Analysis ---
1. Add Person
2. Add Connection
3. Run DFS
4. Run BFS
5. Print Network
6. Exit
Enter your choice (1-6): 3
Enter starting person for DFS: Talha
DFS result: ['Talha', 'Anila', 'Mubashir', 'Saad', 'Annie']
```

BFS

```
--- Social Network Analysis ---
1. Add Person
2. Add Connection
3. Run DFS
4. Run BFS
5. Print Network
6. Exit
Enter your choice (1-6): 4
Enter starting person for BFS: Talha
BFS result: ['Talha', 'Anila', 'Annie', 'Saad', 'Mubashir']
```