Project Report: Friendship Network Simulation and Analysis

Arman Singhal
240184

1. Network Generation Logic

Each student was assigned a set of features:
- Popularity: Randomly generated using an exponential distribution skewed towards 0, normalized to the [0, 1] range.
- Group Memberships: Each student was randomly assigned to one of 5 class groups, sport groups and area groups.

Edge Creation:
- A pairwise similarity score was computed based on shared group membership.
- This score is adjusted using a harmonic mean of the students' popularity, promoting connections between mutually popular students.
- Sigmoid function was applied to the probability to normalize it to the [0, 1] range.
- A secondary mechanism added additional connections based on mutual friends.

Edge Weights:
- Weights were normalized and mapped to the range [1, 10], where a lower weight indicates a stronger friendship.
- Only edges with weights below 2.25 were considered as valid friendships.

2. Number and Size of Friendship Groups

Using Depth-First Search (DFS), the number and size of connected components were analyzed where connections were considered only if the edge weight was less than the defined threshold (2.25).

Results of current run:
- Number of Connected Components: 4
- Maximum Component Size: 410
- Minimum Component Size: 185

This suggests the presence of few large friendship groups and few smaller groups.

3. Sample Shortest Paths

Both Dijkstra's Algorithm and the A* Search Algorithm were implemented to find shortest paths between student pairs.

Both Dijkstra's and A* algorithms generally returned identical path distances and routes but sometimes Dijkstra's outperformed A* like on the current run.

Example Output:

```
Distance between 241 and 796 (Dijkstra): 8.96, Path: [241, 741, 595, 837, 796]
Distance between 241 and 796 (A*): 10.00, Path: [241, 796]
```

This indicates that while A* is theoretically faster with a good heuristic, the current heuristic was not strong enough to outperform Dijkstra in this context.

4. Bridge Observations

Using DFS with Tarjan's algorithm, all bridge nodes were identified.

Results:

- Number of Bridge Nodes: 2
- After removing these nodes:
  - New Number of Connected Components: 7
  - Maximum Component Size: 212
  - Minimum Component Size: 1

This analysis highlights the existence of a few key students who serve as bridges between otherwise disconnected groups.

5. Reflections and Patterns

- The network did not exactly represent networks as most real life ones have a big connected group and few very small ones while this had few groups of roughly the same size.
- The numbers and weights can be tuned to better represent real life networks but this model still gave a good understanding on how graphs work and the different algorithms associated with it.