

## Final Project Write-Up

This Rust application implements the Six Degrees of Separation concept through detailed graph data analysis, leveraging graph traversal and statistical evaluation of graph characteristics. It utilizes the Breadth-First Search (BFS) and statistical methods including calculations of degree distribution, mean, variance, standard deviation, mode, and graph density.

The dataset, known as euroroad, can be found at: [Euroroad Network](#). This dataset represents the international E-road network, primarily in Europe, consisting of undirected edges where nodes symbolize cities, and edges indicate direct E-road connections between them.

### Key Features:

1. Graph Parsing: Efficient parsing of a provided CSV file to construct the graph in an adjacency list format.
2. Breadth-First Search (BFS): Implementation of BFS to navigate the graph and ascertain the shortest route between any two nodes (starting and target nodes).
3. Statistical Analysis: Capability to compute basic statistical measures such as degree distribution, mean, variance, standard deviation, mode, number of connected components, and the graph's density.
4. Modular Design: The software architecture is divided into three main modules:
  - a. bfs.rs: Contains the BFS algorithm for shortest path determination.
  - b. parser.rs: Handles reading and converting the CSV file into a usable graph format.
  - c. statistics.rs: Facilitates statistical analysis of the graph.
5. Testing: Includes two testing modules:
  - a. test.rs: Verifies the functionality of the statistical analysis.
  - b. graph\_test.rs: Ensures that the graph construction functionality operates correctly.

### Running the Program:

1. Prerequisites: Ensure that Rust and Cargo are installed on your system.
2. Compilation: From the project's root directory, compile the software using the command:  
`cargo build --release`
3. Execution: Run the program with the command: `cargo run`
4. Testing: Execute tests with: `cargo test`

### Input Format:

The application requires a CSV file as input, with each line containing two values that represent an edge in the graph, such as:

node1, node2

node3, node4

node5, node6

...

### Program Output:

1. The application outputs various information:
2. BFS Output: Displays the shortest path between two specified nodes, or indicates the absence of a path.
3. Statistical Output: Includes degree distribution, mean, standard deviation, variance, mode, graph density, and the count of connected components.
4. Example Outputs:
5. Degree Distribution: Details such as {"1076": 1, "1115": 1, "1118": 4, ...}
6. Distance between nodes 316 and 330: 9
7. Mean degree: 2.41396933560477
8. Variance: 1.4129564760613882
9. Standard Deviation: 1.1886784578099276
10. Mode: [2]
11. Number of Connected Components: 26
12. Graph Density: 0.002057944872638338