**Project Proposal: The ChatroLeague Chase**

**1. Project Title:**

The ChatroLeague Chase

**2. Project Overview:**

The ChatroLeague Chase is a C programming project focused on implementing and optimizing graph-based shortest path algorithms to simulate a chase scenario. The objective is to develop an efficient algorithm that determines the shortest route between multiple locations where intelligence reports indicate the presence of a target (the ChatroLeague Leader). The program will utilize graphs to represent a network of interconnected locations and implement algorithms such as Dijkstra’s or A\* to find the optimal paths.

**3. Objectives:**

- Develop a C-based program that models locations and paths as a weighted graph.  
- Implement shortest path algorithms (Dijkstra’s/A\* or other suitable methods).  
- Provide input flexibility through manual input and file-based input options.  
- Output the shortest path efficiently while maintaining a simple and clear interface.  
- Optimize the algorithm to handle large datasets efficiently.  
- Use Git for version control and ensure modular, well-documented code.

**4. Scope:**

- The program will allow users to input location data manually or via a file.  
- The graph structure will support weighted edges to account for varying travel costs between locations.  
- The shortest path between designated locations will be computed and displayed.  
- The program will be developed primarily for a command-line interface (CLI) in Windows and GitHub Codespaces environments.  
- Future expansions may include additional heuristics, real-time tracking simulations, or interactive visualization.

**5. Technical Requirements:**

- Language: C  
- Development Environment: Visual Studio Code (Windows) and GitHub Codespaces  
- Version Control: Git  
- Data Structures: Graphs (Adjacency List/Matrix), Priority Queues (for Dijkstra/A\*)  
- Algorithms: Dijkstra’s Algorithm, A\* Algorithm (if heuristics are incorporated)  
- Input/Output Handling: Manual input and file-based input/output with options to toggle.

**6. Expected Challenges:**

- Efficiently handling large graphs with numerous locations and paths.  
- Implementing an optimized priority queue for fast shortest path computation.  
- Managing dynamic input while ensuring program robustness.  
- Testing different datasets to validate algorithm performance and correctness.

**7. Timeline:**

|  |  |  |
| --- | --- | --- |
| Phase | Task | Estimated Duration |
| Phase 1 | Research & Planning | 1 week |
| Phase 2 | Graph Implementation | 2 weeks |
| Phase 3 | Shortest Path Algorithm Implementation | 2 weeks |
| Phase 4 | Input/Output Handling | 1 week |
| Phase 5 | Optimization & Testing | 2 weeks |
| Phase 6 | Documentation & Finalization | 1 week |

**8. Conclusion:**

The ChatroLeague Chase aims to provide an efficient and well-structured implementation of graph-based pathfinding algorithms in C. The project will serve as both a learning experience and a practical demonstration of algorithmic efficiency in real-world problem-solving scenarios. By leveraging Git for version control and optimizing for performance, this project will contribute to a deeper understanding of graph theory and algorithm design.