MiniGit Project Report

# Overview

MiniGit is a lightweight, Git-inspired version control system implemented in C++ to simulate core version control functionalities such as file tracking, committing, branching, merging, and viewing commit history through a command-line interface (CLI)...

# Data Structures

The system employs the following data structures to manage version control operations, aligning with the project's DSA requirements:

Blob: Stores file content with SHA-1 hashing...

Commit Node: Represents a commit with metadata...

Branch Reference: Maps branch names to commit hashes...

Staging Area: Tracks files to be committed...

Log History: Enables traversal of commit history...

# Key Features

MiniGit implements all required features, providing a robust simulation of Git's core functionality:

Initialization (init): Creates a hidden .minigit/ directory...

Add Files (add): Stages files for the next commit...

Commit (commit -m ): Captures a snapshot of staged files...

View Log (log): Traverses the commit history...

Branching (branch): Creates a new branch...

Checkout (checkout): Switches to a specified branch...

Merge (merge): Performs a three-way merge...

Diff Viewer (diff): Implements a basic line-by-line diff viewer...

# Design Decisions

Object-Oriented Design: MiniGit uses separate classes...

Persistence: Stores blobs and commits as files...

Hashing: Relies on a utility function...

Merge Strategy: Implements a three-way merge...

Error Handling: Includes validation...

Modularity: Separates file I/O, hashing, and formatting...

# CLI Usability

MiniGit's CLI mirrors Git's command structure, supporting init, add <filename>, commit -m <message>, log, branch <branch-name>, checkout <branch|hash>, merge <branch-name>, and diff <commit1> <commit2>...

# Limitations

Diff Viewer: The diff implementation shows basic line-by-line changes...

Conflict Resolution: Merge conflicts require manual resolution...

Storage Efficiency: Blobs are stored uncompressed...

Edge Cases: While error handling is robust...

Missing Advanced Features: Features like rebasing...

# Future Improvements

Enhance the diff viewer with a sophisticated algorithm...

Implement automated conflict resolution options...

Add blob compression (e.g., using zlib)...

Introduce a help command to display available commands...

Support advanced Git features like rebase or stash...

Implement garbage collection...

Add more robust validation for edge cases...

# Conclusion

MiniGit successfully implements a lightweight version control system that meets all core and advanced project requirements...