CS 463G SAT Solver Project Roadmap

Phase 0 — Project Setup

- Create Git repo and folder structure
- Add README.md, requirements.txt, and CLI skeleton (run_solver.py)
- Implement simple test using pytest

Phase 1 — Formula Parsing and Loader

- Implement DIMACS parser (parse_dimacs)
- Validate input files and clauses
- Add unit tests for parsing

Phase 2 — Implement DPLL Solver

- Implement DPLL with unit propagation and backtracking
- Add variable selection heuristic
- Verify on small test CNFs

Phase 3 — Implement Heuristic Solvers

- Select two algorithms (e.g., WalkSAT, Genetic Algorithm)
- Implement deterministic seeds and parameter options
- Return best assignment, score, and time

Phase 4 — Experiment Harness

- Build experiment runner (run_experiments.py)
- Run 10 trials for randomized algorithms
- Log results to CSV with time, score, and parameters

Phase 5 — Parameter Sweep & Tuning

- Run grid search over key parameters
- Plot parameter vs score/time trade-offs
- Save tuned parameter defaults

Phase 6 — Data Analysis & Plotting

- Generate plots of solver performance and time
- Create tables comparing average and best results
- Save outputs as PNG/SVG in results/plots/

Phase 7 — Report & Submission

- Compile report.pdf with method summaries and results
- Include data, graphs, and learning outcomes
- Ensure code comments include contribution details