Project TeamworkTemplate

Version 1 9/11/24

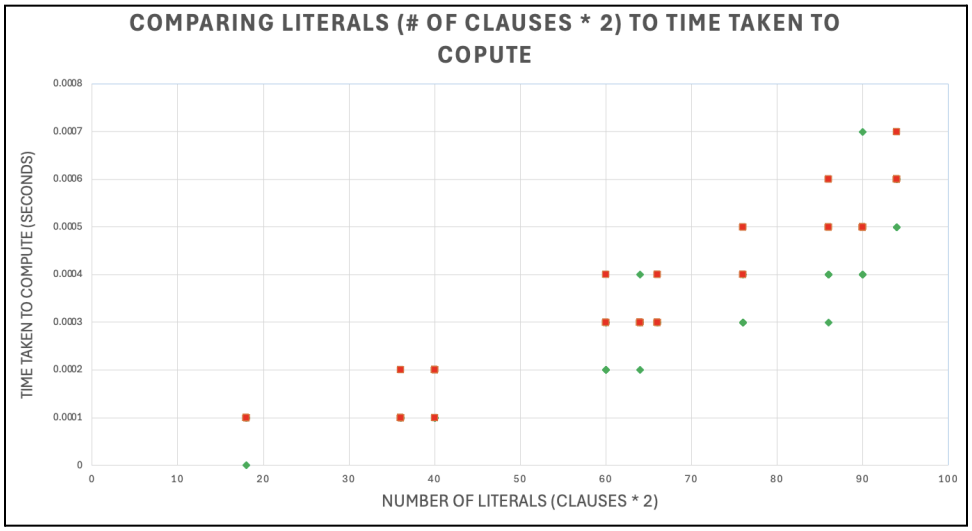
A ***separate copy*** of this template should be filled out and submitted by each student, regardless of the number of students on the team. Also change the title of this template to “Project x Teamwork <team> - <netid>”

1	**Team Name:** oheldrin
2	**Individual name:** Olivia Heldring
3	**Individual netid:** oheldrin
4	**Other team members names and netids:** NA
5	**Link to github repository:** https://github.com/oliviaheldring/TheoryProject_oheldrin
6	**Overall project attempted, with sub-projects:** Implementing a polynomial time 2-SAT solver with the DPLL algorithm
7	List of included files (if you have many files of a certain type, such as test files of different sizes, list just the folder): (Add more rows as necessary)

File/folder Name	File Contents and Use
Code Files	
twoSATcode_oheldrin.py	This contains all of my functions, including my main function. This file reads in the input data, processes each problem, and lists out key information like Problem Number, Number of Clauses, Number of variables, Number of Literals, the Clauses themselves, Satisfiability, and Time taken.
Test Files	
input_file_oheldrin	This was in Canvas and given by Kogge. It contains 100 2-SAT problems. (I did a few of these problems by hand to ensure my code matched the solutions I came up with)
Output Files	
output_file_oheldrin.txt	The results of my code are printed to the terminal and

	stored in this output file. In this file, you will find the following information about all 100 problems: Problem Number, Number of Clauses, Number of variables, Number of Literals, the Clauses themselves, Satisfiability, and Time taken.

Plots (as needed)



- **This graph compares the number of literals (clauses * 2) to the time it takes to compute. As we can see, the time taken grows exponentially as the number of literals grows. The red squares represent unsatisfiable, and the green diamonds represent satisfiable.**

Other	In my repo, I also attached my **readme**, my **teamwork file**, and a file about all my data represented as a clear table.

8	**Individual Student time (in hours) to complete:** 12
9	**Your specific activities and responsibilities:** I worked alone, so everything submitted was completed entirely by me.
10	**What was personally learned (topic, programming, algorithms)** I learned a lot about 2-SAT problems and the DPLL algorithm. 1. <u>CNF Representation</u>: I learned how to represent conjunctive normal form (CNF)

	in Python and process this format for SAT solving. 2. Optimizations: Implementing unit propagation and pure literal elimination improved my understanding of simplifying problems before guessing solutions, making the algorithm more efficient. 3. DPLL Algorithm: I practiced recursive backtracking, where guesses are made, tested, and adjusted. 4. Input Parsing and File Handling: Processing CNF files reminded me how to handle and parse complex input data. 5. Performance Awareness: Measuring runtime gave me insights into computational efficiency, especially with SAT problems.
11	**How team was organized, and what might be improved.** My project was individual, so my team wasn't organized in any way.