Project 1 Readme Team GC

1	Team Name: GC
2	Team members names and netids: Giancarlos Reyes GReyes2
3	Overall project attempted, with sub-projects: SAT: Bruteforce
4	Overall success of the project: Succesful
5	Approximately total time (in hours) to complete: 4 Hours
6	Link to github repository: https://github.com/GReyes87/Project1-TOC.git
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). Add more rows as necessary.

File/folder Name	File Contents and Use
Code Files	
src	sat_GC.py
Test Files	
input	cnffile_GC_check.cnf
Output Files	
results	data_GC.csv
Plots (as needed)	
results	sat_result_plot_GC.png

8	Programming languages used, and associated libraries: Python,itertools,csv,time,os,json, typing, matplotlib
9	Key data structures (for each sub-project): List, list of list, Dictionaries, Tuples, CSV file rows, Matplotlib scatter plot arrays.
10	General operation of code (for each subproject): The brute force SAT solver loads each problem from a .cnf file, tries all possible true/false combinations for the variables, checks if any assignment satisfies all clauses,

	and records the result to a CSV file, which is then used to make a plot showing how problem size affects solver performance.
11	What test cases you used/added, why you used them, what did they tell you about the correctness of your code: I used the provided cnf test cases in the input folder to check if my brute force solver worked. I chose this because it allowed me to easily see if it matched the output that was given to us.
12	How you managed the code development: My first thought with the project was to layout my idea and thought process on paper. After doing this, I followed my guideline step by step ensuring I did not leave any important details out.
13	Detailed discussion of results: The results show that unsatisfiable cases took longer because the solver had to try every possible assignment, while satisfiable ones finished faster once a solution was found. The graph backs this up showing red dots (unsatisfiable) are higher, showing longer times, and green dots (satisfiable) are lower. This makes sense for a brute-force SAT solver and matches what we expect from these types of problems.
14	How team was organized : I was by myself
15	What you might do differently if you did the project again: I think I would just start earlier and ask questions to the TA in a timely manner.
16	Any additional material: n/a