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: Lonely_Garcia
2	Individual name: Rene Alzina
3	Individual netid: ralzina
4	Other team members names and netids: N/A
5	Link to github repository: https://github.com/ralzina/Project2-TOC
6	Overall project attempted, with sub-projects: NTM Tracer
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/ntm_tracer.py	Implements the tracing of NTM tree and handles the reconstruction of the accepting path or the longest rejected path
Test Files	
Input/*	These files have the different cases tested with the NTM: 0^n1^n, composite numbers, and a+
Output Files	
Stdout	The program prints everything into stdout so there are no specific

	<table><tr><td></td><td>output files</td></tr><tr><td colspan="2">Plots (as needed)</td></tr><tr><td>N/A</td><td>N/A</td></tr></table>
8	Individual Student time (in hours) to complete: 4
9	Your specific activities and responsibilities I was the only one on my team, so I took care of developing the entire code I first looked at the provided GitHub repository to understand what each function was meant to do and understand the behavior of the code. Then, I realized I needed to add the functionality for ntm_tracer.py. From there, I implemented the requested function that uses a tree as a list of lists and at each level gets valid transitions from the current state and tape character and then evaluates where to go next while managing the nondeterminism as well. After that, I continued to the reconstruction of the path, where I realized it would be much easier if I knew at each level, what was my previous state that brought me to the current level. So, I went back to add this extra functionality to the past function so that when I had to reconstruct the path, it would be much easier.
10	What was personally learned (topic, programming, algorithms) The results were very impressive because it proves how NTMs can solve any problem that a modern computer can solve, and by implementing it in code, I have gained a much better understanding of how NTMs work. I also learned a lot about data structures and algorithms because I had to use a tree as a list of lists to represent a breadth first search which was a great application of algorithms.
11	How team was organized, and what might be improved. I was the only one in the team. The project could be improved by having valid test cases and by having a valid starting GitHub repository earlier.
12	Any additional material: I created my own test cases since the ones given were wrong, so I had to modify the test cases to work.