

# Project 2 Readme Team BJ

Version 1 9/11/24

A single copy of this template should be filled out and submitted with each project submission, regardless of the number of students on the team. It should have the name `readme_<teamname>` Also change the title of this template to "Project x Readme Team xxx"

1	Team Name: <b>BJ</b>																		
2	Team members names and netids: <b>Brynn Johnson – bjohns33</b>																		
3	Overall project attempted, with sub-projects: <b>Tracing NTM Behavior</b>																		
4	Overall success of the project: <b>I would say this is fairly successful – I got all of my correct outputs needed.</b>																		
5	Approximately total time (in hours) to complete: <b>5 hours</b>																		
6	Link to github repository: <a href="https://github.com/bjohns77/Project2-TOC"><b>https://github.com/bjohns77/Project2-TOC</b></a>																		
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.																		
<table border="1"><thead><tr><th>File/folder Name</th><th>File Contents and Use</th></tr></thead><tbody><tr><td colspan="2">Code Files</td></tr><tr><td><b>./src/ntm_tracer.py</b></td><td><b>This contains all of the trace and print code.</b></td></tr><tr><td colspan="2">Test Files</td></tr><tr><td><b>./inputs/aplus.csv</b></td><td><b>Given file for the a<sup>+</sup> NTM</b></td></tr><tr><td><b>./inputs/aplus_bplus.csv</b></td><td><b>A csv file I created for the a<sup>+</sup>b<sup>+</sup> NTM for testing reject states</b></td></tr><tr><td><b>./inputs/composite.csv</b></td><td><b>Given file for testing</b></td></tr><tr><td colspan="2">Output Files</td></tr><tr><td><b>./outputs/terminal_screenshot.jpg</b></td><td><b>Screenshots of the terminal</b></td></tr></tbody></table>		File/folder Name	File Contents and Use	Code Files		<b>./src/ntm_tracer.py</b>	<b>This contains all of the trace and print code.</b>	Test Files		<b>./inputs/aplus.csv</b>	<b>Given file for the a<sup>+</sup> NTM</b>	<b>./inputs/aplus_bplus.csv</b>	<b>A csv file I created for the a<sup>+</sup>b<sup>+</sup> NTM for testing reject states</b>	<b>./inputs/composite.csv</b>	<b>Given file for testing</b>	Output Files		<b>./outputs/terminal_screenshot.jpg</b>	<b>Screenshots of the terminal</b>
File/folder Name	File Contents and Use																		
Code Files																			
<b>./src/ntm_tracer.py</b>	<b>This contains all of the trace and print code.</b>																		
Test Files																			
<b>./inputs/aplus.csv</b>	<b>Given file for the a<sup>+</sup> NTM</b>																		
<b>./inputs/aplus_bplus.csv</b>	<b>A csv file I created for the a<sup>+</sup>b<sup>+</sup> NTM for testing reject states</b>																		
<b>./inputs/composite.csv</b>	<b>Given file for testing</b>																		
Output Files																			
<b>./outputs/terminal_screenshot.jpg</b>	<b>Screenshots of the terminal</b>																		

	Plots (as needed) – Not required for this project
8	Programming languages used, and associated libraries: <b>Python</b>
9	Key data structures (for each sub-project): <b>Tree:</b> <ul style="list-style-type: none"><li>- List of lists</li></ul> <b>Dictionary:</b> <ul style="list-style-type: none"><li>- This was the transition function that was automatically created by the program</li></ul>
10	General operation of code: <ul style="list-style-type: none"><li>- This NTM trace code uses a breadth first search to implement a non-deterministic search for the accept state based on the transition functions. If there is no accept state, there it continues or rejects the value. If it accepts, it will print out the path that is taken. If the entire machine rejects, it prints out the number of states seen.</li></ul>
	What test cases you used/added, why you used them, what did they tell you about the correctness of your code. I added the testing file aplus_bplus.csv. This allowed me to do testing on rejection states: “aaaa” meant that there were no b’s so it was rejected. “aaaab” was accepted. “aaaba” was rejected.  I used the aplus.csv testing file, with the default test in order to test my NTM and printing correctness.
12	How you managed the code development I did most of the code in one sitting. I created the NTM trace first and then the printing function. I tackled each section step by step, printing things out to debug.
13	Detailed discussion of results: This program resulted in a printing out the path that was taken by the NTM to reach the accepting. If there was no way to reach an accepting state, it would reject and print out a rejecting message with the number of states reached.
14	How team was organized: I was working alone, so I was in charge of writing all the code and doing the write-up.
15	What you might do differently if you did the project again. I would give myself more time for the project, starting it sooner. I would also collaborate with others to bounce ideas off of them. I worked completely alone this time.
16	Any additional material: Instead of syncing my fork, I decided to copy all the code and directories from the provided repository into my own repository. This is an exact copy of the repository; it was just not forked.

