Project 2 Results Discussion
Luca Watson

CODE DISCUSSION:

My code works by first taking command line arguments and using these to generate an object to represent an NTM, complete with a starting state, accept state, reject state, and list of transitions. The code then creates a representation of the starting configuration, with the state and starting string, and adds it to a list to serve as the first level of my tree. The code then loops through each level of the tree, then loops through each configuration in this level, and then loops through all of the NTMs transitions for each configuration to check if each could be taken. If a transition can be taken, as dictated by the state and input symbol described by the transition, a new configuration will be generated from the current configuration and transition description and added to the next level. This continues until the loop is broken by an accept state being reached, or every configuration entering the reject state by an inability to make any other transitions.

Non-determinism is modelled in my code by a single configuration being able to produce multiple other configurations in the next level of the tree if it is able to take multiple transitions. My test cases demonstrate this behavior as they produced trees that have more transitions taken than levels.

I traced my machine's execution by adding a fourth element to each configuration list to represent its parent configuration within the tree, or the configuration that it transitioned from. This makes it much easier to trace the path taken by any given string.

Non-determinism in my machine is calculated according to the description provided - it is the average number of new configurations generated by an average configuration, which is calculated by averaging the number of configurations in each level divided by the number of configurations in the previous level. The non-determinism values output by the machines make sense, as completely deterministic machines all output a determinism value of 1.0, while more non-deterministic machines have values well above 1.

I tested my code by running test scripts with varying levels of determinism. I first started with the example provided and made sure the output matched, then I used relatively short strings for each machine and checked that the input was correct by comparing it to what I got by hand. I then used much longer strings and modified my code to print out

each configuration and transition taken, and verified that the path taken made sense given the machine.

TEST CASES + OUTPUT
Path traces omitted from output column to save space (some got very long)

Machine	Input string	Output
a_equal_b.csv	abbabababbab	Machine name: {w | w has the same number of a's and b's} - deterministic Initial string: abbabababbab Tree depth: 75 Transitions simulated: 74 String rejected in: 75 Degree of non-determinism: 1.0
a_equal_b.csv	abbaaabbabaabb	Machine name: {w | w has the same number of a's and b's} - deterministic Initial string: abbaaabbabaabb Tree depth: 132 Transitions simulated: 131 String accepted in: 132 Degree of non-determinism: 1.0
a_plus.csv	aaa	Machine name: a plus Initial string: aaa Tree depth: 5 Transitions simulated: 10 String accepted in: 5 Degree of non-determinism: 1.2333333333333334

a_plus.csv	aaaaaaaab	Machine name: a plus Initial string: aaaaaaaab Tree depth: 9 Transitions simulated: 23 String rejected in: 9 Degree of non-determinism: 1.1666666666666667
abc_star.csv	aaabbbccccc	Machine name: a*b*c* Nondeterministic Initial string: aaabbbccccc Tree depth: 13 Transitions simulated: 51 String accepted in: 13 Degree of non-determinism: 1.185897435897436
abc_star.csv	aaabbbcccccbc	Machine name: a*b*c* Nondeterministic Initial string: aaabbbcccccbc Tree depth: 12 Transitions simulated: 50 String rejected in: 12 Degree of non-determinism: 1.2638888888888888
equal_01s_DTM.csv	0101010101010101010101010001110	Machine name: {w | w has the same number of 0's and 1's} Deterministic Initial string: 0101010101010101010101010001110 Tree depth: 552

		Transitions simulated: 551 String rejected in: 552 Degree of non-determinism: 1.0
equal_01s_DTM.csv	01010011	Machine name: {w	w has the same number of 0's and 1's} Deterministic Initial string: 01010011 Tree depth: 44 Transitions simulated: 43 String accepted in: 44 Degree of non-determinism: 1.0
equal_01s.csv	01010101010101010101010 101010011	Machine name: {w	w has the same number of 0's and 1's} Nondeterministic Initial string: 010101010101010101010101010011 Tree depth: 484 Transitions simulated: 723 String accepted in: 484 Degree of non-determinism: 1.0165289256198347
equal_01s.csv	01010101010101010	Machine name: {w	w has the same number of 0's and 1's} Nondeterministic Initial string: 01010101010101010 Tree depth: 146 Transitions simulated: 217 String rejected in: 146 Degree of non-determinism: 1.0273972602739727