Project 2 Teamwork Team JL

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: JL
2	Individual name Julia Lizak
3	Individual netid jlizak
4	Other team members names and netids
5	Link to github repository: https://github.com/julializak/Project2-TOC
6	Overall project attempted, with sub-projects: NTM Trace
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	
ntm_tracer.py	Main implementation, had the TODO statements, where I wrote majority of my code
entrypoint.py	Command line Had to add the main function in there when I was calling to test it
Test Files	
aplus.csv	Given. NTM test for a language
composite.csv	Given.NTM for composite checking
equal_01s.csv	Given. Strings with equal numbers of 0 and 1s
Output Files	
No output files but here are some screenshots	

```
∨ TERMINAL

● jlizak@student10:~/Project1-TOC$ python3 -m src.entrypoint input/aplus.csv aaa --max_depth 10
  Tracing NTM: a plus on input 'aaa'
  String accepted in 4 transitions.
  Config: aa q3 a_
  Tree depth: 4
  Total transitions: 7
    q1 aaa
   a q1 aa
   a q2 aa
   aa q1 a
   aa q2 a
   aaa q1 _
   aaa q2 _
   aa q3 a_
  Avg nondeterminism: 1.750
○ jlizak@student10:~/Project1-TOC$ ▮
```

```
∨ TERMINAL

● jlizak@student10:~/Project1-TOC$ python3 -m src.entrypoint input/composite.csv 111111 --max_depth 20
  Tracing NTM: Composite_tester on input '111111'
  Execution stopped after 20 steps.
  Tree depth: 20
  Total transitions: 72
  Avg nondeterminism: 1.059
    q1 111111
   $ q2 11111
   $x q3 1111
   $xx q3 111
   $ q4 x1111
   $xxx q3 11
   $x q4 x111
    q4 $x1111
   $xxxx q3 1
   $xx q4 x11
   $ q4 xx111
   _ q5s x1111
   $xxxxx q3 _
   $xxx q4 x1
   $x q4 xx11
    q4 $xx111
   _x q5s 1111
   $xx q4 xx1
   $ q4 xxx11
   _ q5s xx111
   _ q6 x$111
   $x q4 xxx1
    q4 $xxx11
   _x q5s x111
    q6 _x$111
   $ q4 xxxx1
   _ q5s xxx11
   _xx q5s 111
   _ q7 x$111
    q4 $xxxx1
   _x q5s xx11
   _x q6 x$11
   __ q5x $111
   _ q5s xxxx1
   _xx q5s x11
   _ q6 xx$11
   __$ q5x 111
   _x q5s xxx1
   _xxx q5s 11
    q6 _xx$11
   __ q6 $x11
   _xx q5s xx1
   _xx q6 x$1
   _ q7 xx$11
   _ q6 _$x11
   _xxx q5s x1
   _x q6 xx$1
   __ q5x x$11
```

```
         _xx q5s x11
        _ q6 xx$11
        __$ q5x 111
        _x q5s xxx1
        _xxx q5s 11
         q6 _xx$11
        __ q6 $x11
        _xx q5s xx1
        _xx q6 x$1
        _ q7 xx$11
        _ q6 _$x11
        _xxx q5s x1
        _x q6 xx$1
        __ q5x x$11
        __ q7 $x11
        _xxxx q5s 1
        _ q6 xxx$1
        ___ q5x $11
        ___ q5s x11
        _xxx q6 x$
         q6 _xxx$1
        ___$ q5x 11
        ___x q5s 11
        _xx q6 xx$
        _ q7 xxx$1
        ___ q6 $x1
        ___ q6 x$1
        _x q6 xxx$
        __ q5x xx$1
        __ q6 _$x1
        __ q6 _x$1
        _ q6 xxxx$
        ___ q5x x$1
        ___ q7 $x1
        ___ q7 x$1
         q6 _xxxx$
        ____ q5x $1
        ____ q5s x1
        ____ q5x $1
○ jlizak@student10:~/Project1-TOC$ ▮
```

```
∨ TERMINAL
● jlizak@student10:~/Project1-TOC$ python3 -m src.entrypoint input/equal_01s.csv 0101 --max_depth 20
  Tracing NTM: {w | w has the same number of 0's and 1's} Nondeterministic on input '0101'
  String accepted in 13 transitions.
  Config: _xxx_ qacc
  Tree depth: 13
  Total transitions: 21
   q0 0101
   _ q1 101
   q3 _x01
   _ q4 x01
   _ q5 x01
   _x q4 01
   _x q5 01
   _xx q1 1
   _x q3 xx
   _ q3 xxx
   q3 _xxx
   _ q4 xxx
   _ q5 xxx
   _x q4 xx
   _x q5 xx
   _xx q4 x
   _xx q5 x
   _xxx q4 _
   _xxx q5 _
   _xx q3 x_
   _xxx_ qacc _
  Avg nondeterminism: 1.105
○ jlizak@student10:~/Project1-TOC$ ▮
```

8	Individual Student time (in hours) to complete: 7-8
9	Your specific activities and responsibilities Everything
10	What was personally learned (topic, programming, algorithms) Better practice of how nondeterministic TM's operate and how to code them, as well as

	using BFS. More practice with Python and debugging and code organization, especially after being given kind of a template to start with. Turning formal TM definitions into code and things that can be executed.
11	How team was organized, and what might be improved. Done by myself
12	Any additional material: