

User Guide

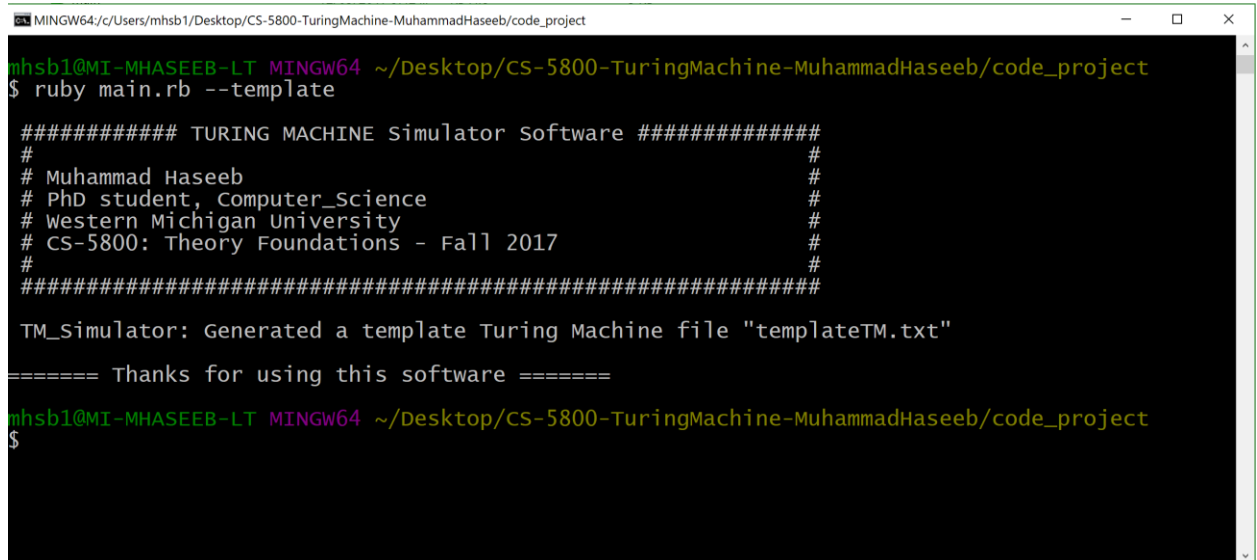
Introduction

This document contains the user guide for the **Turing Machine Simulator** software.

Generating a Template Turing Machine

To generate a template Turing machine file:

- Invoke the main.rb script with Ruby with -t or --template switch



```

MINGW64/c/Users/mhsb1/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$ ruby main.rb --template

##### TURING MACHINE Simulator Software #####
#
# Muhammad Haseeb
# PhD student, Computer_Science
# Western Michigan University
# CS-5800: Theory Foundations - Fall 2017
#
#####

TM_Simulator: Generated a template Turing Machine file "templateTM.txt"

===== Thanks for using this software =====

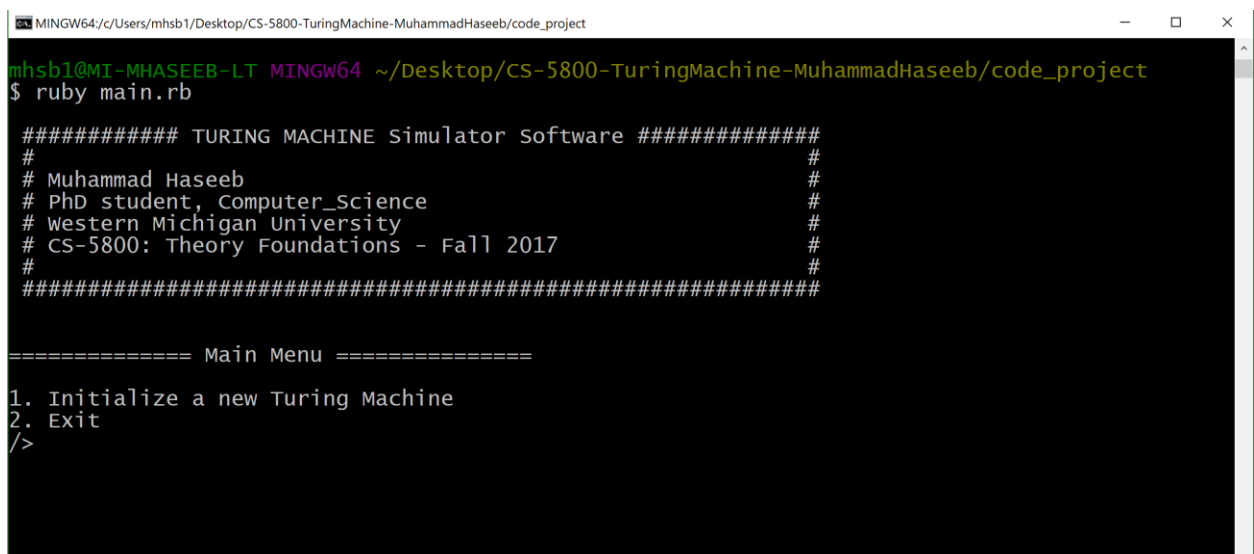
mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$

```

NOTE: In order to avoid any inconvenience, always generate a new Template TM file and modify it according to your TM.

Using the software

- Invoke the main.rb script using Ruby without any switch



```

MINGW64/c/Users/mhsb1/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$ ruby main.rb

##### TURING MACHINE Simulator Software #####
#
# Muhammad Haseeb
# PhD student, Computer_Science
# Western Michigan University
# CS-5800: Theory Foundations - Fall 2017
#
#####

===== Main Menu =====

1. Initialize a new Turing Machine
2. Exit
/>

```

- From the main menu, enter 1 to go to Initializing a Turing machine sub-menu and enter the path to TM file and press enter.

```

MINGW64/c/Users/mhsb1/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$ ruby main.rb

##### TURING MACHINE Simulator Software #####
#
# Muhammad Haseeb
# PhD student, Computer Science
# Western Michigan University
# CS-5800: Theory Foundations - Fall 2017
#
#####

===== Main Menu =====

1. Initialize a new Turing Machine
2. Exit
/> 1

Enter the relative or absolute path to TM input file OR -1 to back to main menu
Note: Only use the forward slashes. example: ./path/to/TM_file.txt
/> ./examples/languageAcceptor/LA1.txt

```

- Now enter any string you want your machine to operate on, once the results output, it will ask you if you want to enter any more strings for the same machine, put in your answer and it will do so, entering No, N, n, NO will take you back to main menu

```

MINGW64/c/Users/mhsb1/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project

Enter the string for tape (T): /> abaabbabbbabbbbaaa
Number of extra blanks "B"s to append at end of string /> 2

==== Simulating the Language Acceptor Turing Machine on input string: abaabbabbbabbbbaaaBBBBBBBBBBBBBBBBBBBBB ====
Tape:  BabaabbabbbabbbbaaaBBBBBBBBBBBBBBBBBBBBBBBB
      ^
Head>  [q3]

==== RESULTS: The string is ACCEPTED by the L(TM) ====

Do you want to enter another string for the same Turing Machine? Y/N? /> y
Enter the string for tape (T): /> ababbbabbbab
Number of extra blanks "B"s to append at end of string /> 2

==== Simulating the Language Acceptor Turing Machine on input string: ababbbabbbabBBBBBBBBBBBBBBBBBBBBB ====
Tape:  BababbbabbbabBBBBBBBBBBBBBBBBBBBBBBBB
      ^
Head>  [q1]

==== RESULTS: The string is NOT ACCEPTED by the L(TM) ====

Do you want to enter another string for the same Turing Machine? Y/N? /> 2
Do you want to enter another string for the same Turing Machine? Y/N? /> n

===== Main Menu =====

1. Initialize a new Turing Machine
2. Exit
/>

```

- Enter 2 at main menu to quit software

```
Do you want to enter another string for the same Turing Machine? Y/N? /> n

===== Main Menu =====
1. Initialize a new Turing Machine
2. Exit
/> 2

===== Thanks for using this software =====

mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$
```

Turing Machine Input File format

The Turing Machine input file looks like this:

```
MINGW64/c/Users/mhsb1/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$ cat templateTM.txt
# Auto-generated template for Turing Machine
# Generated at: 2017-12-06 13:44:59 -0500 Eastern Standard Time

# Turing Machine Configuration
# options: standard, LanguageAcceptor
Machine=standard

# Set of States and Markers separated by comma
Q=q0,q1,q2

# Set of Input Alphabets separated by comma
E=a,b

# Initial State
I=q0

# Set of Final states separated by comma
# Required for only Language Acceptors. Ignored otherwise
F=q2

# Transition Table
# Separated by pipe
S|B|a|b
q0|q1,B,R|nil,nil,nil|nil,nil,nil
q1|q2,B,L|q1,b,R|q1,a,R
q2|nil,nil,nil|q2,a,L|q2,b,L

mhsb1@MI-MHASEEB-LT MINGW64 ~/Desktop/CS-5800-TuringMachine-MuhammadHaseeb/code_project
$
```

The file has following options:

- **Machine:** The variation of the Turing machine in question [valid options: standard, LanguageAcceptor]
- **Q:** Set of states in the machine separated by commas
- **E:** Set of input alphabets and any markers, don't put the B (blank) in here. The software automatically adds B to this set.
- **I:** The initial state of the machine
- **F:** The final state of machine. This is only required for Language Acceptors else it will be ignored

- **Transition Table:**

- **Rows:** The transition table rows should be pipe | separated and in the order with the E set, for example if $E=a,b,X,Y$ then the rows of T-Table should be $S|a|b|X|Y$.
- **Columns:** The columns should be in order with the set of states Q. For instance, if $Q=q_0,q_1,q_2,q_f$ the columns must be according to that format, the indexing of the transition table is according to that.