ChunQian Moy

CSCE 428: Project Report

**Part 1**

For part 1 my program will first remove all the white spaces on the file and fetch all information needed from the text file and store it as variable. States and alphabet will store in array of string, final states will store in an array on integer, 1 or accepting state and 0 for non-accepting state according to the index of the states, start state will be store in an int variable which is the index of the starting state. Then, Delta will be store in a 2D vector with y-axis indicating the state and x-axis indicating the alphabet, and the vector store the next state when user enter an input. For example, when the dfa is in state 1 which is index 0 in the state array and input alphabet 0 which is index 0 in the alphabet array and the next state will be state 2 and therefore the vector [0][0] will store the name of state2. I have a verify function that will verify the index number with the parameter of the name of the state.

After, the fetching of information is done, the program will start to check for membership by first getting the x and y index for the vector by using the verify function. The verify function will first verify the input alphabet with the respective x index and the index of the start state for the y index. Then, the program will keep on looping with the next input until the end of the input and check whether the current state is a final state. If yes, then it will print a “Yes” and it’s not a final state then it will Print a “No”. Moreover, my program also check for empty string and invalid input, if it’s an invalid input it will print a “No”. For empty string, it will check the start state, if the start state is a final state then print a yes and vice versa.

Potential Problem

The main problem for this program is the formatting of the txt file.

* There should always be a ‘#’ after the end of the line for start state and accepting states.
* There shouldn’t be anything written after input\_alphabet, start\_state and delta.
* The last state on the file cannot be an accepting state, for example

If it’s not a valid file, it will not throw any exception.

**Part 2**

For part2, the fetching part works almost the same as part 1 except all the information are stored in vector, there’s separate function to retrieve different information such accepting state, dfa, starting state and etc. After the fetching of information, my program will then call a function check that needs all the information of both dfa as parameter to check for equivalence.

In the check function, it will create a new dfa which the intersection of the two dfa, the program will do a cross product on the accepting states, states, delta and create the intersection for both of the dfa.