

Tecnológico de Monterrey Campus Santa Fe
Computational Mathematics
Professor Salvador E. Venegas-Andraca
Second Period Project

October 2019

I. Goal. To design and to implement an algorithm capable of accepting an NFA and producing an equivalent DFA.

II. Teams. This project is to be made in pairs. You choose your partner, just make sure that both names are included as a comment on your code file.

III. Deadline. 23:59 hrs, Thursday 24 October 2019.

IV. Project specification

1. Your program will accept as input an NFA described according to the following format: $\{(input_datum, v_i, v_j)\}$. For example, the NFA presented in Fig. (1):

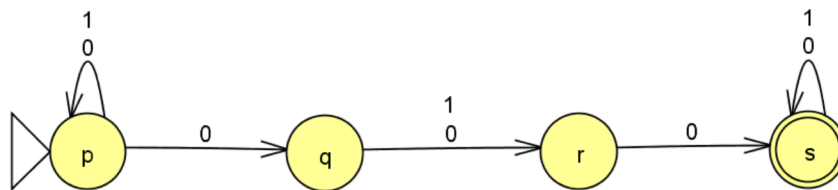


Fig.(1)

would be described as $\{(0,p,p),(0,p,q),(1,p,p),(0,q,r),(1,q,r),(0,r,s),(0,s,s),(1,s,s)\}$

2. The input NFA will be delivered to your program as a text file. Hence, your program must provide users with a mechanism to upload text files. Please use the file **NFA_test_file_Aug_Dec_2019.txt** to test your program.

3. Your program will compute a DFA capable of recognizing the same language the corresponding input NFA does.

4. The DFA produced in the previous step will be stored in a text file following the same format used for NFAs. This text file will be recorded in the same directory the input NFA was uploaded from.

5. Programming language for this course: Python, Anaconda distribution (<https://www.anaconda.com/download>)

V. Project grading

- I will test your algorithm with three different NFAs.
- If your algorithm:
 - does not run (i.e. errors while interpreting your program) then your grade will be 10/100.
 - cannot upload text files then your grade will be 10/100.
 - runs but it does not produce a DFA as required in the project specification then your grade will be 40/100.
 - fulfills the requirements stated in section IV then your grade will be 100/100.