Department of Computer Science & Software Engineering

Comp5361 Discrete Structures and Formal Languages

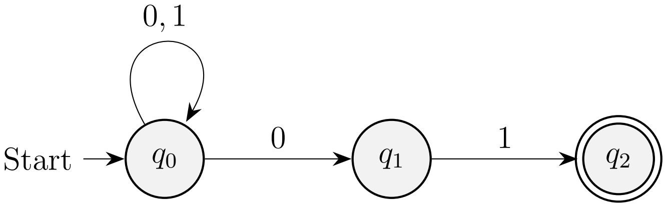
Winter 2021

Programming Assignment 3. Due date: April 14.

1. Use the `PySimpleAutomata`-software to write a program that takes as input a transition table for a DFA or NFA A, and outputs the transition diagram for A. For example, the input could be the automaton $A = (\{q_0, q_1, q_2\}, \{0,1\}, \delta, q_0, \{q_2\})$ that has transition table

δ	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\varnothing$	$\{q_2\}$
$\star q_2$	$\varnothing$	$\varnothing$

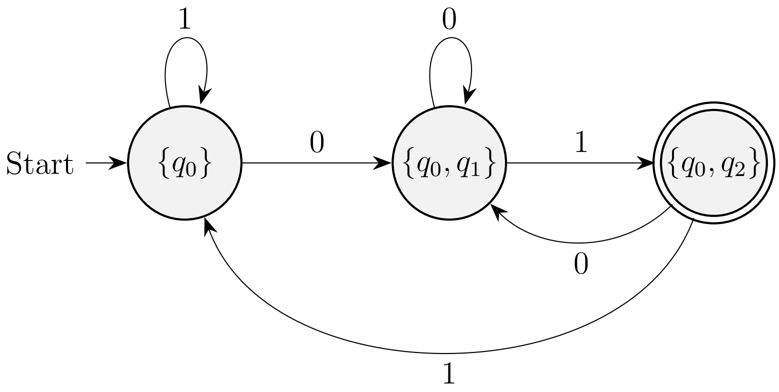
The automaton A as a transition diagram:



2. Write a Python program that takes as input the transition table for an NFA A (no ϵ-transitions), such as the one above, and outputs the transition table and diagram for a DFA B, such that $L(A) = L(B)$. For example, the NFA above should result in output

	0	1
$\varnothing$	$\varnothing$	$\varnothing$
$\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_1\}$	$\varnothing$	$\{q_2\}$
$\star\{q_2\}$	$\varnothing$	$\varnothing$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$\star\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\star\{q_1, q_2\}$	$\varnothing$	$\{q_2\}$
$\star\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$

and



Input/Output format

The input/output format is up to the student. You may take and show the information in any format you want. The input/output streams is up to the student as well. For example, you may read from `sys.stdin` or from a file.

Use of libraries and some built-in python functions

- Any other function, whether built-in or not, that directly simplifies the assignment is not allowed.

What to submit

Using the Moodle system, submit a .zip file that includes:

1. Source code.
2. A .pdf file, explaining your format and how to run your code in few lines.

The format to be followed for submission file is :

`<yourname>_passign3_<student-id>.zip`

How to prepare for the demo

You will have to demonstrate your program in a one to one zoom meeting (the date of which will be announced) and your program will be tested using other test cases.

1. Correctness of the program : 20%
2. Valid input and output cases passed : 40%
3. Comprehension of the written program : 30%
4. Understanding of the problem statement 10%