CSE332: Theory of Computation
Homework 1

Student Name: Nguyen Minh Duc
Student ID: 20202026

Problem 1 (10pts). Prove that $(uv)^R = v^R u^R$ for all $u, v \in \Sigma^+$. (Hint: Use induction on the length of v.)

Solution.

Let $n = |v|$ and $P(n)$ be the statement "$\forall u \in \Sigma^+, v \in \Sigma^n : (uv)^R = v^R u^R$".
Base case. $n = 1$, i.e. $v = a,\ a \in \Sigma$. Then, $(uv)^R = (ua)^R = au^R = a^R u^R = v^R u^R$, hence, $P(1)$ is true.
Induction case. Suppose that $P(k)$ is true for $1 \leq k \leq n$ and $n > 0$, i.e. $(uv)^R = v^R u^R$ holds for arbitrary string u and all string v of length up to n over the alphabet Σ. We need to show $P(n+1)$ is also true.
Let $v = xy$, where $x, y \in \Sigma^+$ and $|v| = |x| + |y| = n + 1$. This implies that $|x|, |y| \leq n$. Then,

$$\begin{aligned}
(uv)^R &= (u(xy))^R \\
&= ((ux)y)^R \\
&= y^R (ux)^R \quad (\because \text{ Induction hypothesis for } |y| \leq n) \\
&= y^R \left(x^R u^R\right) \quad (\because \text{ Induction hypothesis for } |x| \leq n) \\
&= \left(y^R x^R\right) u^R \\
&= (xy)^R u^R \quad (\because \text{ Induction hypothesis for } |x|, |y| \leq n) \\
&= v^R u^R,
\end{aligned}$$

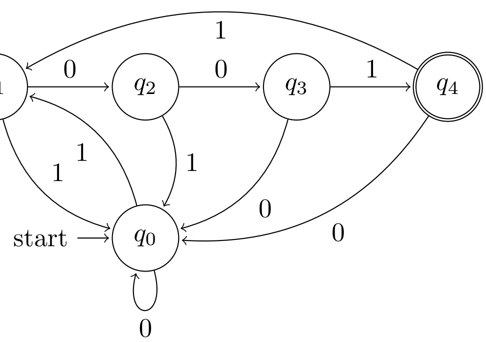
hence, $P(n+1)$ also holds.
Thus, this completes the mathematical strong induction that $P(n)$ is true for all $n > 0$, which implies $(uv)^R = v^R u^R$ for all $u, v \in \Sigma^+$.

Problem 2 (10pts). Consider the following language: $L = \{w \in \{0,1\}^* \mid w \text{ ends with } 1001.\}$.

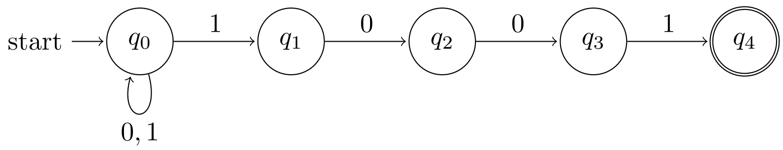
1. (5pts) Design a DFA that accepts L.
2. (5pts) Design an NFA that accepts L.

Solution.

1. (5pts) Design a DFA that accepts L.



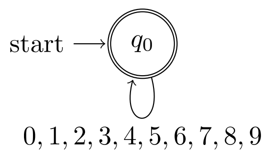
2. (5pts) Design an NFA that accepts L.



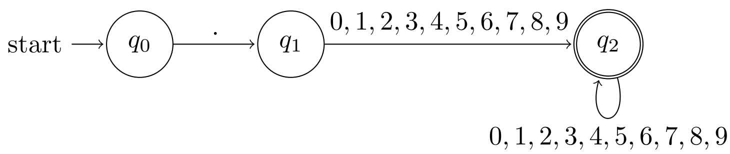
> **Problem 3** (10pts) Design an NFA to recognize the strings that represent real numbers. Assume $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, .\}$. For example, the NFA should accept strings such as "1.0", "12.156", and ".01", but must reject strings such as "0.5.1", "12.", and "3".

Solution.

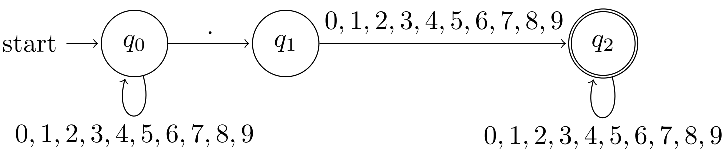
Note that the required real numbers have the form "(digits or empty)(. and digits)".
Let D be the regular expression representing digits, i.e. $D =$ "$(0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9)$".
In regular expression, real numbers could be represented as "$D^*(.D^+)$".
This regular expression consists of two parts "D^*" and "$(.D^+)$" concatenating together. Let's build them one by one.
NFA for "D^*":



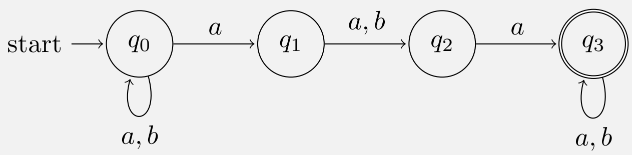
NFA for "$(.D^+)$":



Concatenating the two NFAs together, we obtain the NFA for the real numbers as follow



> **Problem 4** (20pts) Use subset construction to convert the following NFA to a DFA:

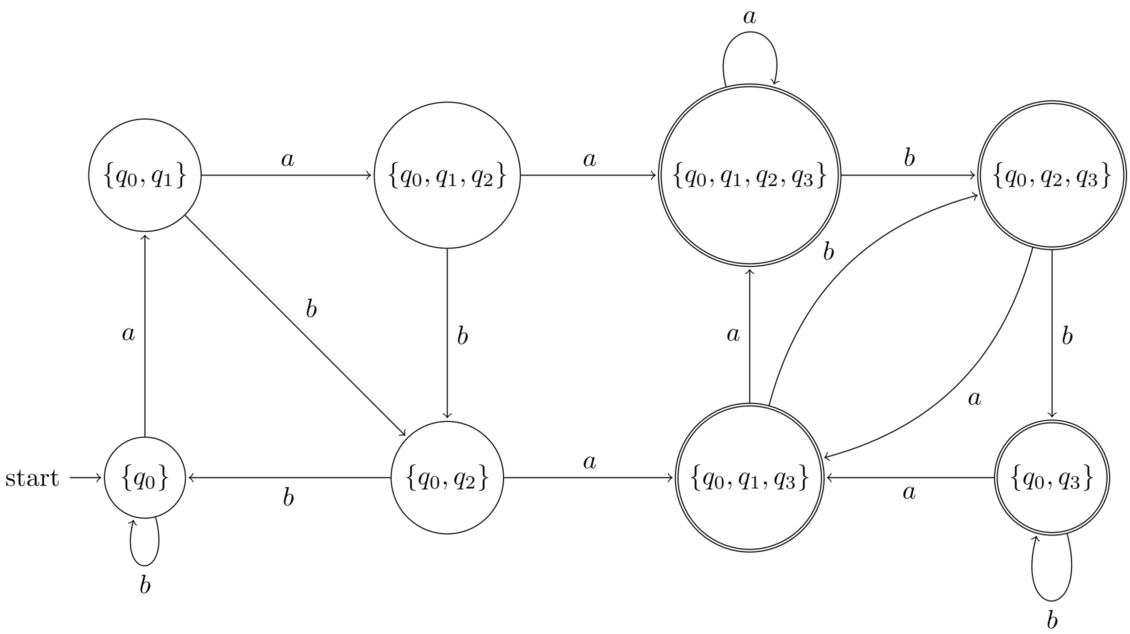


Solution.

The transition table for the DFA is:

	a	b
start $\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1, q_3\}$	$\{q_0\}$
$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_2\}$
$^*\{q_0, q_1, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_2, q_3\}$
$^*\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_2, q_3\}$
$^*\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$
$^*\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$

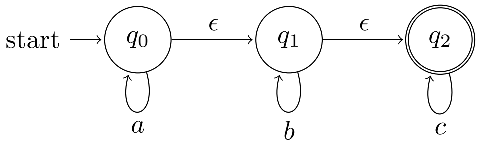
The transition graph for the DFA is:



Problem 5 (10pts) Design an ϵ-NFA that accepts the following language:

$$L = \{a^m b^n c^o \mid m, n, o \geq 0\}$$

Solution.



Problem 6 (20pts) Consider the following transition table of an ϵ-NFA:

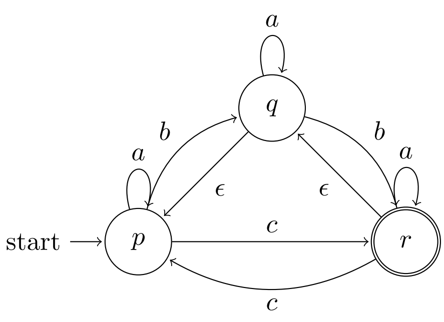
	ϵ	a	b	c
p	$\emptyset$	$\{p\}$	$\{q\}$	$\{r\}$
q	$\{p\}$	$\{q\}$	$\{r\}$	$\emptyset$
r	$\{q\}$	$\{r\}$	$\emptyset$	$\{p\}$

where p is the initial state and r is the final state.

1. (10pts) Compute the ϵ-closure(EClosure) of each state.
2. (10pts) Convert the automaton to a DFA.

Solution.

The transition graph for the ϵ-NFA is:



1. (10pts) Compute the ϵ-closure(EClosure) of each state.

$$\begin{aligned}
\text{EC}\textsc{lose}(\{p\}) &= \{p\} \\
\text{EC}\textsc{lose}(\{q\}) &= \{p, q\} \\
\text{EC}\textsc{lose}(\{r\}) &= \{p, q, r\} \\
\text{EC}\textsc{lose}(\{p, q\}) &= \{p, q\} \\
\text{EC}\textsc{lose}(\{p, r\}) &= \{p, q, r\} \\
\text{EC}\textsc{lose}(\{q, r\}) &= \{p, q, r\} \\
\text{EC}\textsc{lose}(\{p, q, r\}) &= \{p, q, r\}
\end{aligned}$$

2. (10pts) Convert the automaton to a DFA.

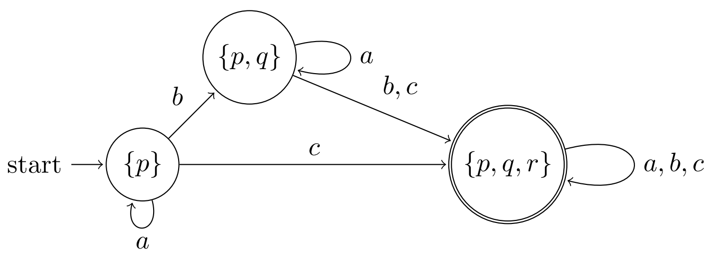
Let $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$ be the ϵ-NFA, and $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$ be the equivalent DFA, then

$$\begin{aligned}
Q_D &= \{S \subseteq Q_E \mid S = \text{EC}\textsc{lose}(S)\} = \{\{p\}, \{p, q\}, \{p, q, r\}\}\,, \\
q_D &= \text{EC}\textsc{lose}(q_0) = \{p\}\,, \\
F_D &= \{S \in Q_D \mid S \cap F_E \neq \emptyset\} = \{p, q, r\}\,,
\end{aligned}$$

and the transition table of D is

	a	b	c
start $\to \{p\}$	$\{p\}$	$\{p, q\}$	$\{p, q, r\}$
$\{p, q\}$	$\{p, q\}$	$\{p, q, r\}$	$\{p, q, r\}$
$^* \{p, q, r\}$	$\{p, q, r\}$	$\{p, q, r\}$	$\{p, q, r\}$

Thus, the equivalent DFA



Problem 7 (15pts, 5pts each) Find regular expressions for the following languages.

1. $L = \{\omega \in \{a, b, c\}^* \mid \omega \text{ has no more than three } a\text{'s}\}$
2. $L = \{\omega \in \{0, 1\}^* \mid \omega \text{ begins and ends with 0 and contains at least one 1}\}$
3. $L = \{\omega \in \{0, 1\}^* \mid \omega \text{ does not contain } 111\}$

Solution.

1. $L = \{\omega \in \{a, b, c\}^* \mid \omega \text{ has no more than three } a\text{'s}\}$

$$(b + c)^* a?(b + c)^* a?(b + c)^* a?(b + c)^*$$

2. $L = \{\omega \in \{0, 1\}^* \mid \omega \text{ begins and ends with 0 and contains at least one 1}\}$

$$0^+ 1(0 + 1)^* 0$$

3. $L = \{\omega \in \{0, 1\}^* \mid \omega \text{ does not contain } 111\}$

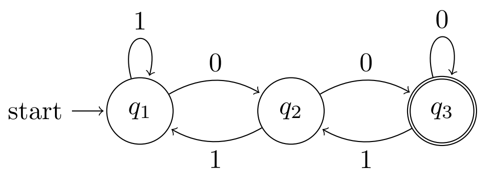
$$(0 + 10 + 110)^*(\epsilon + 1 + 11)$$

Problem 8 (20pts) Consider a DFA represented by a transition table:

	0	1
$\rightarrow q_1$	q_2	q_1
q_2	q_3	q_1
$*q_3$	q_3	q_2

Give all the regular expressions $R_{ij}^{(0)}$,$R_{ij}^{(1)}$,$R_{ij}^{(2)}$. Try to simplify the expressions as much as possible. Think of state q_i as if it were the state with number i.

Solution.



For $k = 0$,

$$
\begin{aligned}
R_{11}^{(0)} &= \epsilon + 1 \\
R_{12}^{(0)} &= 0 \\
R_{13}^{(0)} &= \emptyset \\
R_{21}^{(0)} &= 1 \\
R_{22}^{(0)} &= \epsilon \\
R_{23}^{(0)} &= 0 \\
R_{31}^{(0)} &= \emptyset \\
R_{32}^{(0)} &= 1 \\
R_{33}^{(0)} &= \epsilon + 0
\end{aligned}
$$

For $k = 1$,

$$
\begin{aligned}
R_{11}^{(1)} &= (\epsilon + 1) + (\epsilon + 1)(\epsilon + 1)^*(\epsilon + 1) \\
&= (\epsilon + 1)^+ = 1^* \\
R_{12}^{(1)} &= 0 + (\epsilon + 1)(\epsilon + 1)^* 0 \\
&= (\epsilon + 1)^* 0 = 1^* 0 \\
R_{13}^{(1)} &= \emptyset + (\epsilon + 1)(\epsilon + 1)^* \emptyset \\
&= \emptyset \\
R_{21}^{(1)} &= 1 + 1(\epsilon + 1)^*(\epsilon + 1) \\
&= 1(\epsilon + 1)^* = 1^+ \\
R_{22}^{(1)} &= \epsilon + 1(\epsilon + 1)^* 0 \\
&= \epsilon + 1^+ 0 \\
R_{23}^{(1)} &= 0 + 1(\epsilon + 1)^* \emptyset \\
&= 0 \\
R_{31}^{(1)} &= \emptyset + \emptyset(\epsilon + 1)^*(\epsilon + 1) = \emptyset \\
R_{32}^{(1)} &= 1 + \emptyset(\epsilon + 1)^* 0 = 1 \\
R_{33}^{(1)} &= (\epsilon + 0) + \emptyset(\epsilon + 1)^* \emptyset \\
&= \epsilon + 0
\end{aligned}
$$

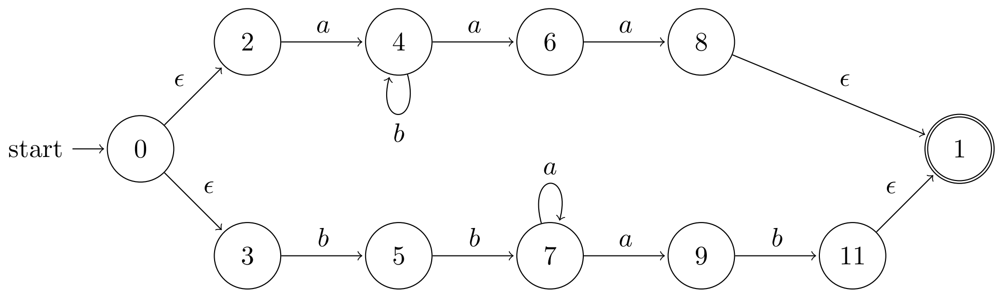
For $k = 2$,

$$
\begin{aligned}
R_{11}^{(2)} &= 1^* + (1^*0)\left(\epsilon + 1^+0\right)^*\left(1^+\right) \\
&= 1^* + (1^*0)\left(1^+0\right)^*\left(1^+\right) \\
&= (1 + 01)^* \\
R_{12}^{(2)} &= 1^*0 + (1^*0)\left(\epsilon + 1^+0\right)^*\left(\epsilon + 1^+0\right) \\
&= 1^*0\left(\epsilon + 1^+0\right)^* \\
&= (1 + 01)^*\,0 \\
R_{13}^{(2)} &= \emptyset + (1^*0)\left(\epsilon + 1^+0\right)^*0 \\
&= (1 + 01)^*\,00 \\
R_{21}^{(2)} &= 1^+ + \left(\epsilon + 1^+0\right)\left(\epsilon + 1^+0\right)^*\left(1^+\right) \\
&= \left(1^+0\right)^*1^+ \\
R_{22}^{(2)} &= \epsilon + 1^+0 + \left(\epsilon + 1^+0\right)\left(\epsilon + 1^+0\right)^*\left(\epsilon + 1^+0\right) \\
&= \left(\epsilon + 1^+0\right)^* \\
&= \left(1^+0\right)^* \\
R_{23}^{(2)} &= 0 + \left(\epsilon + 1^+0\right)\left(\epsilon + 1^+0\right)^*0 \\
&= \left(1^+0\right)^*0 \\
R_{31}^{(2)} &= \emptyset + 1\left(\epsilon + 1^+0\right)^*1^+ \\
&= 1\left(1^+0\right)^*1^+ \\
R_{32}^{(2)} &= 1 + 1\left(\epsilon + 1^+0\right)^*\left(\epsilon + 1^+0\right) \\
&= 1\left(1^+0\right)^* \\
R_{33}^{(2)} &= (\epsilon + 0) + 1\left(\epsilon + 1^+0\right)^*0 \\
&= \epsilon + 0 + 1\left(1^+0\right)^*0
\end{aligned}
$$

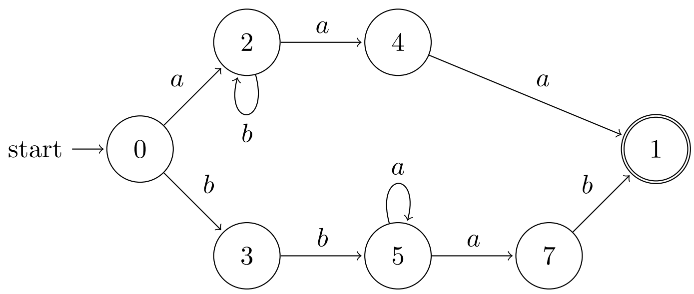
Problem 9 (10pts) Convert the following regular expressions to finite automata (ϵ-NFA):

$$ab^*aa + bba^*ab$$

Solution.



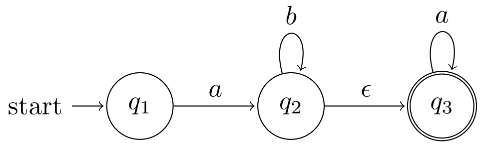
This could be reduced further



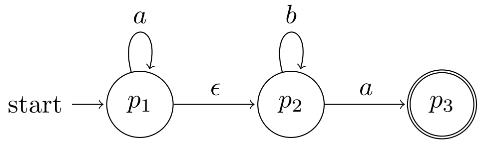
Problem 10 (10pts) Find a ϵ-NFA that accepts language $L(ab^*a^*) \cap L(a^*b^*a)$.

Solution.

The ϵ-NFA for $L(ab^*a^*)$:



The ϵ-NFA for $L(a^*b^*a)$:



Merging the two ϵ-NFAs, we obtain: