Midterm Exam

COSE215: Theory of Computation
2024 Spring

Instructor: Jihyeok Park

April 24, 2024. 13:30-14:45

- **If you are not good at English, please write your answers in Korean.**
  (영어가 익숙하지 않은 경우, 답안을 한글로 작성해 주세요.)
- **Write answers in good handwriting.**
  If we cannot recognize your answers, you will not get any points.
  (글씨를 알아보기 힘들면 점수를 드릴 수 없습니다. 답안을 읽기 좋게 작성해주세요.)
- **Write your answers in the boxes provided.**
  (답안을 제공된 박스 안에 작성해 주세요.)

Student ID	
Student Name	

Question:	1	2	3	4	5	6	7	Total
Points:	15	15	15	20	15	10	10	100
Score:								

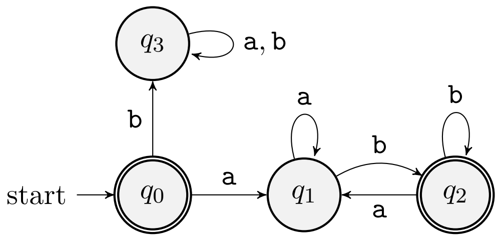
1. 15 points A **deterministic finite automaton (DFA)** D is 5-tuple:

$$D = (Q, \Sigma, \delta, q_0, F)$$

- Q is a finite set of **states**
- Σ is a finite set of **symbols**
- $\delta : Q \times \Sigma \to Q$ is the **transition function**
- $q_0 \in Q$ is the **initial state**
- $F \subseteq Q$ is the set of **final states**

Consider the following DFA D described with a **transition diagram**:

$D =$



(a) 6 points Fill in the blanks in the **transition table** of D.

(Note that $\to$ indicates the **initial state**, and $*$ indicates a **final state**.)

	$q_i \in Q$	a	b
☐	q_0	q_1	q_3
☐	q_1	q_1	q_2
☐	q_2	☐	☐
☐	q_3	☐	☐

(b) 6 points The **extended transition function** $\delta^* : Q \times \Sigma^* \to Q$ of D is defined as:

- **(Basis Case)** $\delta^*(q, \epsilon) = q$
- **(Induction Case)** $\delta^*(q, aw) = \delta^*(\delta(q, a), w)$ where $a \in \Sigma$, $w \in \Sigma^*$

Fill in the blanks in the results of the **extended transition function** δ^* of D.

$$\delta^*(q_1, \epsilon) = \square$$
$$\delta^*(q_0, \texttt{bab}) = \square$$
$$\delta^*(q_2, \texttt{abab}) = \square$$

(c) 3 points Describe the **language** accepted by D:

$L(D) =$ ☐

2. 15 points Design a **DFA** using a **transition diagram** that accepts the following languages.

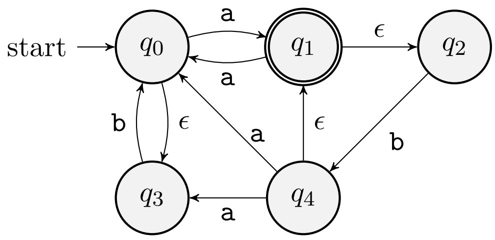
(a) 5 points $L = \{\texttt{a}^n \mid n \not\equiv 0 \pmod 3\}$

(b) 5 points $L = \{w \in \{\texttt{0}, \texttt{1}\}^* \mid w \text{ is any string except } \texttt{1} \text{ or } \texttt{10}\}$

(c) 5 points $L = \{w \in \{\texttt{a}, \texttt{b}\}^* \mid w \text{ has at least two } \texttt{a}\text{'s and even number of } \texttt{b}\text{'s}\}$

3. [15 points] Consider the following **ϵ-nondeterministic finite automaton (ϵ-NFA)** N^ϵ:

$$N^\epsilon = (Q = \{q_0, q_1, q_2, q_3, q_4\}, \Sigma = \{\mathtt{a}, \mathtt{b}\}, \delta, q_0, F = \{q_1\})$$



(a) [6 points] Construct a **DFA** D using a **transition table** such that $L(D) = L(N^\epsilon)$ via the **subset construction**.

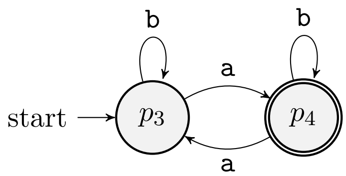
(Note that $\rightarrow$ indicates the **initial state**, and $*$ indicates a **final state**.)

$D =$

	$P = \mathcal{P}(Q)$	a	b
☐	$p_0 = \{$ ☐ $\}$	☐	☐
☐	$p_1 = \{$ ☐ $\}$	☐	☐
☐	$p_2 = \{$ ☐ $\}$	☐	☐

(b) [6 points] Consider the following DFA $D' = (P' = \{p_3, p_4\}, \Sigma, \delta', p_3, \{p_4\})$:

$D' =$



Fill in the table in the left side using the **table-filling algorithm** for the states of D and D', and define the **equivalence classes** $(P \cup P')/_{\equiv}$ in the right side using the result of the algorithm.

	p_0	p_1	p_2	p_3
p_1	☐			
p_2	☐	☐		
p_3	☐	☐	☐	
p_4	☐	☐	☐	☐

$(P \cup P')/_{\equiv} = \{$

$\{$ ☐ $\}$,

$\{$ ☐ $\}$,

$\}$

(c) [3 points] Explain why D and D' are **equivalent** using the result of the algorithm:

☐

4. [20 points] Write **regular expressions (REs)** that represent the following languages.

(a) [5 points] $L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ has exactly two } \mathtt{a}\text{'s}\}$.

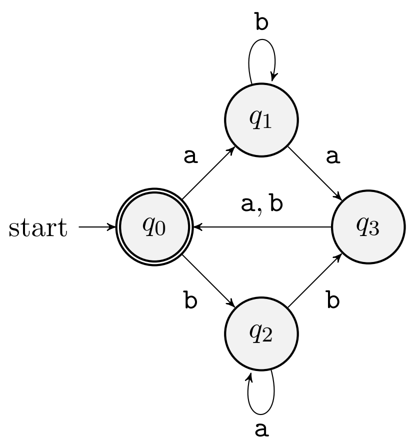
$R =$ [ ]

(b) [5 points] $L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* \mid w \text{ has } \mathtt{bab} \text{ as a substring}\}$.

(Note that a **substring** is a contiguous sequence of characters within a string. For example, $\mathtt{b\underline{bab}b}, \mathtt{\underline{bab}bba}, \mathtt{baa\underline{bab}} \in L$ but $\mathtt{bb}, \mathtt{baab}, \mathtt{abaaba} \notin L$.)

$R =$ [ ]

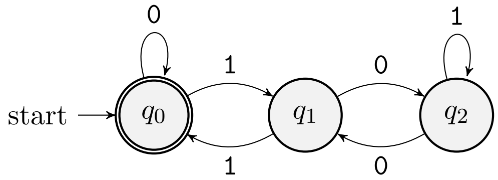
(c) [5 points] $L \subseteq \{\mathtt{a}, \mathtt{b}\}^*$ is the language of the following DFA:



$R =$ [ ]

(d) [5 points] $L \subseteq \{\mathtt{a}, \mathtt{b}\}^*$ is the language such that $h(L)$ is the language of the following DFA where $h : \{\mathtt{a}, \mathtt{b}\} \to \{\mathtt{0}, \mathtt{1}\}^*$ is a homomorphism with $h(\mathtt{a}) = \mathtt{101}$ and $h(\mathtt{b}) = \mathtt{11}$.

(Hint: construct a DFA accepting L and convert it to an equivalent regular expression.)



$R =$ [ ]

5. 15 points Fill in the blanks in the following **proofs** showing that each language is **NOT regular** using the **pumping lemma** for regular languages.

(a) 7 points $L = \{w \in \{\texttt{a}, \texttt{b}\}^* \mid w \text{ has the same number of } \texttt{a}\text{'s and } \texttt{b}\text{'s}\}$.

1. Assume that any positive integer n is given. (i.e., $n \geq 1$)
2. Pick a word $L \ni w =$ [ ] .
3. $|w| =$ [ ] $\geq n$.
4. Assume that any split $w = xyz$ satisfying ① $|y| > 0$ and ② $|xy| \leq n$ is given.
5. Let $i =$ [ ] . We need to show that ¬③ $xy^iz \notin L$:

[ ]

(b) 8 points The language $L = \{\texttt{a}^i\texttt{b}^j \mid i \neq j\}$ is **NOT regular**.

Prove that the following language L' is also **NOT regular** using the **closure properties** of regular languages with the fact that L is not regular.

$$L' = \{w \in \{\texttt{a}, \texttt{b}\}^* \mid \text{the number of } \texttt{a}\text{'s is not equal to the number of } \texttt{b}\text{'s}\}$$

[ ]

6. [10 points] Design **context-free grammars** that represent the following languages.

(a) [5 points] $L = \{w \in \{\texttt{a}, \texttt{b}, \texttt{c}\}^* \mid N_{\texttt{a}}(w) = N_{\texttt{b}}(w)\}$

(b) [5 points] $L = \{\texttt{a}^i\texttt{b}^j\texttt{c}^k \mid i, j, k \geq 0 \wedge i + 2k = j\}$

7. [10 points] Design an **unambiguous context-free grammar** for the following language.

$$L = \{w \in \{\texttt{0}, \texttt{1}, \texttt{+}, \texttt{*}\}^* \mid w \text{ is an arithmetic expression evaluated to an odd number}\}$$

Only single-digit numbers (`0` or `1`) are allowed, and multiple digits (e.g., `110`) are not allowed. The `*` operator has **higher precedence** than the `+` operator, and they are **left-associative**. For example, `1+0*1` is in L because it is evaluated to an odd number `1+(0*1)` $= 1$.

This is the last page.
I hope that your tests went well!