1 YES/NO questions

Circle the correct answer (each question is worth 2pt.) Write SHORT justification.

1. For any function \( f \) from \( A \neq \emptyset \) onto \( A \), \( f \) has property
\[ \forall a \in A \exists b \in A(f(b) = a). \]

Justify: definition of "onto" function.

2. Some infinite sets have the same cardinality.

Justify: \( |N| = |2N| \) and \( N \) (natural numbers) and \( 2N \) (even numbers) are infinite sets.

3. \( \{a,b\} \in \{a,b,\{a,b\}\} \)

Justify: \( \{a,b\} \subseteq \{a,b,\{a,b\}\} \) as \( \{a,b\} \in \{a,b,\{a,b\}\} \)

4. For any function \( R \subseteq A \times A \), \( R^{-1} \) exists.

Justify: Theorem: The inverse function \( R^{-1} \) exists iff \( R \) is 1 – 1 and "onto".

5. A language \( L \) is regular iff \( L = L(r) \) for some \( r \in \Sigma^* \).

Justify: only when \( r \) is a regular expression.

6. \( L^+ = L^* - \{e\} \).

Justify: only when \( e \not\in L \). When \( e \in L \) we get that \( e \in L^+ \) and \( e \not\in L^* - \{e\} \).

7. For any languages \( L_1, L_2, (L_1 \cup L_2) \cup L_1 = L_2 \).

Justify: languages are sets, so it holds only when only when \( L_1 \subseteq L_2 \).

8. \( (\emptyset^* \cap b^*) \cup \emptyset^* \) describes a language with two elements.

Justify: the set \( (\{e\} \cap b^*) \cup \{e\} = \{e\} \cup \{e\} = \{e\} \) has one element.

9. For any automata \( M \), \( L(M) = \emptyset \) iff the set \( F \) of its final states is empty.

Justify: Let \( M \) be such that \( \Sigma = \emptyset, F \neq \emptyset, s \not\in F \), we get \( L(M) = \emptyset \).
10. If \( M = (K, \Sigma, \Delta, s, F) \) is a non-deterministic as defined in the book, then \( M \) is also non-deterministic, as defined in the lecture.

\textbf{Justify:} \( \Sigma \cup \{e\} \subseteq \Sigma^* \)

11. Let \( M \) be a finite state automaton, \( L(M) = \bigcup_{q \in F} \{ w \in \Sigma^* : (s, w)^* \xrightarrow{M} (q, e) \} \).

\textbf{Justify:} \( w \in \bigcup_{g \in F} \{ w \in \Sigma^* : (s, w)^* \xrightarrow{M} (q, e) \} \) iff there is \( q \in F \) such that \( (s, w)^* \xrightarrow{M} (q, e) \) iff \( w \in L \).

12. For any finite automats \( M_1, M_2 \), \( L(M_1) = L(M_2) \) iff \( M_1 \equiv M_2 \).

\textbf{Justify:} definition of automata equivalency.

13. DFA and NDFA recognize the same class of languages.

\textbf{Justify:} theorem proved in class

\section{Two definitions of a non-deterministic automaton}

\textbf{BOOK DEFINITION:} \( M = (K, \Sigma, \Delta, s, F) \) is non-deterministic when \( \Delta \subseteq K \times (\Sigma \cup \{e\}) \times K \)

\textbf{OBSERVE} that \( \Delta \) is always finite because \( K, \Sigma \) are finite sets.

\textbf{LECTURE DEFINITION:} \( M = (K, \Sigma, \Delta, s, F) \) is non-deterministic when \( \Delta \) is finite and \( \Delta \subseteq K \times \Sigma^* \times K \)

\textbf{OBSERVE} that we have to say in this case that \( \Delta \) is finite because \( \Sigma^* \) is infinite.

\textbf{SOLVING PROBLEMS} you can use any of these definitions.

\section{PROBLEMS}

\textbf{PROBLEM 1} Let \( \Sigma = \{a, b\} \). Show that \( (a \cup b)^*a(a \cup b)^* = \Sigma^*a\Sigma^* \).

\textbf{Solution} Observe that \( \mathcal{L}(a \cup b)^* = (\{a\} \cup \{b\})^* = \{a, b\}^* = \Sigma^* \).

Hence \( (a \cup b)^*a(a \cup b)^* = \Sigma^*a\Sigma^* \).

\textbf{PROBLEM 2} Write a regular expression \( r \), such that \( L = \mathcal{L}(r) \) for \( L \) over \( \Sigma = \{a, b\} \) defined as \( L = \{w \in \Sigma^* : w \text{ has no more than three } a's\} \).
Solution

\[ r = b^* \cup b^*ab^* \cup b^*ab^*ab^* \cup b^*ab^*ab^*ab^* \]

**PROBLEM 3** Let \( L \) be a language defined as follows

\[ L = \{ w \in \{a, b\}^* : \text{between any two } a's \text{ in } w \text{ there is an even number of consecutive } b's. \} \]

1. Describe a regular expression \( r \) such that \( L(r) = L \) (Meaning of \( r \) is \( L \)).

**Solution** Remark that 0 is an even number, hence \( a^* \in L \),

\[ r = b^*a^*b^* \cup b^*(a(bb)^*a)^*b^* = (b^*a(bb)^*ab^*)^* \]

2. Construct a finite state automata \( M \), such that \( L(M) = L \).

**Solution**

Components of \( M \) are:

\[ \Sigma = \{a, b\}, K = \{q_0, q_1, q_2, q_3\}, s = q_0, F = \{q_0, q_2, q_3\} \]

\[ \Delta = \{(q_0, b, q_0), (q_0, a, q_3), (q_0, a, q_1), (q_1, bb, q_1), (q_1, a, q_2), (q_2, c, q_0), (q_2, b, q_2), (q_3, b, q_3), (q_3, e, q_0)\} \]

Some elements of \( L(M) \) as defined by the state diagram are:

\[ b, a, aaaa, aabbb, bbbbaba, abbbaba, abbbabbb, ababbaaba, abbabababa, abababababa, .... \]

**PROBLEM 4** Let

\[ M = (K, \Sigma, s, \Delta, F) \]

for \( K = \{q_0\}, s = q_0, \Sigma = \{a, b\}, F = \{q_0\} \) and

\[ \Delta = \{(q_0, aba, q_0), (q_0, ab, q_0)\} \]

1. List some elements of \( L(M) \).

**Solution**

\[ e, ab, abab, ababa, ababaaba, ... \]

2. Write a regular expression for the language accepted by \( M \).

**Solution**

\[ L = (ab \cup aba)^* \]

3. Use the Book Definition to define an automaton \( M' \) such that \( M' \equiv M \) (use the ”STRETCH” technique).

\[ K' = K \cup \{p_1, p_2, p_3\}, \Delta' = \Delta_{\Sigma,e} \cup \{(q_0, a, p_1), (p_1, b, p_2), (p_2, a, q_0), (q_0, b, p_3), (p_3, b, q_0)\} \]

where \( \Delta_{\Sigma,e} \) denotes those elements of \( \Delta \) that involve only elements of \( \Sigma \cup e \).
PROBLEM 5 (20pts)

For $M$ defined as follows

$$M = (K, \Sigma, s, \Delta, F)$$

for $K = \{q_0, q_1, q_2\}$, $s = q_0$

$\Sigma = \{a, b\}$, $F = \{q_1, q_2\}$ and

$$\Delta = \{(q_0, ab, q_1), (q_0, e, q_1), (q_0, b, q_2), (q_1, a, q_1), (q_2, bb, q_2), (q_1, e, q_2)\}$$

Write a regular expression describing $L(M)$.

$$aba^*(bb)^* \cup a^*(bb)^* \cup b(bb)^*$$

Write 5 steps of the general method of transformation the NDFA $M$, into an equivalent deterministic $M'$.

Reminder 1: $E(q) = \{p \in K : (q, e) \xrightarrow{M} (p, e)\}$ and

$$\delta(Q, \sigma) = \bigcup \{E(p) : \exists q \in Q, (q, \sigma, p) \in \Delta\}.$$ 

Reminder 2: The above definitions apply to the Book definition of non-deterministic automata.

The proper DIAGRAM of new (book definition - use "stretch" method) $M$ is:

$K' = K \cup \{p_1, p_2\}$,

$\Delta' = \Delta_{\Sigma \cup e} \cup \{(g_0, a, p_1), (p_1, b, q_1), (q_2, b, p_2), (p_2, b, q_2)\}$

where $\Delta_{\Sigma \cup e}$ denotes those elements of $\Delta$ that involve only elements of $\Sigma \cup e$.

Solution: apply definition to $M'$ defined above.