Strings and Sets:

- A string over $\Sigma$ is any finite-length sequence of elements of $\Sigma$
- The set of all strings over alphabet $\Sigma$ is denoted as $\Sigma^*$
- Operators over set:
  - set complement, union, intersection, etc.
  - set concatenation $AB$, power of set $A^n$, $A^*$, $A^+$

Right-linear grammars (regular grammars) can define regular language

The set of all strings accepted by deterministic finite automaton (DFA) is denoted as $A = L(M)$

Grammars and machine models are related: Chomsky hierarchy

A subset $A \subseteq \Sigma^*$ is said to be a regular set if $A = L(M)$ for some finite automaton $M$. 
Strings and Sets:

- A string over $\Sigma$ is any finite-length sequence of elements of $\Sigma$
- The set of all strings over alphabet $\Sigma$ is denoted as $\Sigma^*$
- Operators over set:
  - set complement, union, intersection, etc.
  - set concatenation $AB$, power of set $A^n$, $A^*$, $A^+$

Right-linear grammars (regular grammar) can define regular language
Strings and Sets:

- A string over $\Sigma$ is any finite-length sequence of elements of $\Sigma$
- The set of all strings over alphabet $\Sigma$ is denoted as $\Sigma^*$
- Operators over set:
  - set complement, union, intersection, etc.
  - set concatenation $AB$, power of set $A^n$, $A^*$, $A^+$

- Right-linear grammars (regular grammar) can define regular language
- The set of all strings accepted by deterministic finite automaton (DFA) is denoted as $A = L(M)$
Strings and Sets:

- A string over $\Sigma$ is any finite-length sequence of elements of $\Sigma$
- The set of all strings over alphabet $\Sigma$ is denoted as $\Sigma^*$
- Operators over set:
  - set complement, union, intersection, etc.
  - set concatenation $AB$, power of set $A^n$, $A^*$, $A^+$

Right-linear grammars (regular grammar) can define regular language

- The set of all strings accepted by deterministic finite automaton (DFA) is denoted as $A = L(M)$

Grammars and machine models are related: Chomsky hierarchy
Strings and Sets:

- A string over $\Sigma$ is any finite-length sequence of elements of $\Sigma$
- The set of all strings over alphabet $\Sigma$ is denoted as $\Sigma^*$
- Operators over set:
  - set complement, union, intersection, etc.
  - set concatenation $AB$, power of set $A^n$, $A^*$, $A^+$

- Right-linear grammars (regular grammar) can define regular language

- The set of all strings accepted by deterministic finite automaton (DFA) is denoted as $A = L(M)$

- Grammars and machine models are related: Chomsky hierarchy
  - right-linear grammars $\prec \prec \prec \prec$ finite memory
  - A subset $A \subseteq \Sigma^*$ is said to be a regular set if $A = L(M)$ for some finite automaton $M$. 
We model these abstractly by a mathematical model called a finite automaton

**Definition (Deterministic finite automaton)**

Formally, a deterministic finite automaton (DFA) is a structure $M = (Q, \Sigma, \delta, s, F)$, where

- $Q$ is a finite set of states;
- $\Sigma$ is a finite set called input alphabet;
- $\delta : Q \times \Sigma \rightarrow Q$ is the transition function;
- $s \in Q$ is the start state;
- $F$ is a subset of $Q$; elements of $F$ are called accept or final states.
Question 1. Show that the following language is regular
\( \{ awa \mid w \in \{a, b\}^* \} \)
Question 1. Show that the following language is regular
\( \{ awa \mid w \in \{a, b\}^* \} \)
Question 1. Show that the following language is regular
\[ \{ awa \mid w \in \{ a, b \}^* \} \]
Question 1. Show that the following language is regular
\{awaw \mid w \in \{a, b\}^*\}
Question 1. Show that the following language is regular
\[ \{ awa | w \in \{a, b\}^* \} \]

Since we have constructed a DFA for the language, we can claim that, by definition, the language is regular.
Some Closure Properties of Regular Sets

- Closure under intersection: \( A \cap B \)
- Closure under complement: \( \sim A \)
- Closure under union: \( A \cup B = \sim (\sim A \cap \sim B) \)
Some Closure Properties of Regular Sets

- Closure under intersection: $A \cap B$
- Closure under complement: $\sim A$
- Closure under union: $A \cup B = \sim (\sim A \cap \sim B)$

Devise a way to construct new automaton

- intersection: the product construction
Let $M_3 = (Q_3, \Sigma, \delta_3, s_3, F_3)$ where

- $Q_3 = Q_1 \times Q_2 = \{(p, q) \mid p \in Q_1 \land q \in Q_2\}$

- $F_3 = F_1 \times F_2$

- $s_3 = (s_1, s_2)$

- $\delta_3 : Q_3 \times \Sigma \longrightarrow Q_3$ defined by
  $$\delta_3 ((p, q), a) = (\delta_1 (p, a), \delta_2 (q, a))$$

The automaton $M_3$ is called the product of $M_1$ and $M_2$. 

Induction to prove that an automaton accepts a set

Some Closure Properties of Regular Sets

The Product Construction

Closure under intersection

Closure under complement

Closure under union
Question 2. Use the product construction to come up with a deterministic finite automaton (DFA) that accepts all strings from \( \{a, b\}^* \) that contain an even number of \( a \) and that are formed by a repetition of the string \( ab \) (ie \( ababababab \)). Your report should explicitly give all the steps leading to the construction of the proposed DFA.
Question 2. Use the product construction to come up with a deterministic finite automaton (DFA) that accepts all strings from \( \{a, b\}^* \) that contain an even number of \( a \) and that are formed by a repetition of the string \( ab \) (ie \( ababababab \)). Your report should explicitly give all the steps leading to the construction of the proposed DFA.

1. \( L(M_1) = \{x \in \{a, b\}^* | x \) contains even number of \( a \)’s\} \)
Question 2. Use the product construction to come up with a deterministic finite automaton (DFA) that accepts all strings from \( \{a, b\}^* \) that contain an even number of \( a \) and that are formed by a repetition of the string \( ab \) (ie \( ababababab \)). Your report should explicitly give all the steps leading to the construction of the proposed DFA.

1. \( L(M_1) = \{ x \in \{a, b\}^* | x \text{ contains even number of } a's \} \)

2. \( L(M_2) = \{ x \in \{a, b\}^* | x \text{ is formed by repetition of } ab \} \)
Question 2. Use the product construction to come up with a deterministic finite automaton (DFA) that accepts all strings from \( \{a, b\}^* \) that contain an even number of \( a \) and that are formed by a repetition of the string \( ab \) (ie \( ababababab \)). Your report should explicitly give all the steps leading to the construction of the proposed DFA.

1. \( L(M_1) = \{x \in \{a, b\}^* | x \text{ contains even number of } a's\} \)

2. \( L(M_2) = \{x \in \{a, b\}^* | x \text{ is formed by repetition of } ab\} \)

3. \( L(M_3) = L(M_1) \cap L(M_2): \text{ Product construct} \)
1. $L(M_1) = \{ x \in \{a, b\}^* | x \text{ contains even number of } a's \}$

1. $Q_1 = \{ e, o \}$
2. $\sum = \{ a, b \}$
3. $S_1 = e$
4. $F_1 = \{ e \}$

5. $\delta_1$ can be given as table:

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e$</td>
<td>$e$</td>
<td>$e$</td>
</tr>
<tr>
<td>$o$</td>
<td>$e$</td>
<td>$o$</td>
</tr>
<tr>
<td>$o$</td>
<td>$e$</td>
<td>$o$</td>
</tr>
</tbody>
</table>
1. \( L(M_1) = \{ x \in \{ a, b \}^* | x \text{ contains even number of } a's \} \)

1. \( Q_1 = \{ e, o \} \)
2. \( \sum = \{ a, b \} \)
3. \( S_1 = e \)
4. \( F_1 = \{ e \} \)

5. \( \delta_1 \) can be given as table:

\[
\begin{array}{cc}
  & a & b \\
e & e & o \\
o & e & o
\end{array}
\]
2. \( L(M_2) = \{ x \in \{a, b\}^* | x \text{ is formed by repetition of } ab \} \)
2. \( L(M_2) = \{ x \in \{a, b\}^* \mid x \) is formed by repetition of \( ab \} \)
2. $L(M_2) = \{ x \in \{a, b\}^* | x \text{ is formed by repetition of } ab \}$
2. $L(M_2) = \{ x \in \{a, b\}^* | x \text{ is formed by repetition of } ab \}$

1. $Q_2 = \{q_0, q_1, q_2\}$
2. $\Sigma = \{a, b\}$
3. $s_2 = q_0$
4. $F_2 = \{q_0\}$
5. $\delta_2$ can be given as following table:

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>$q_1$</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>$q_2$</td>
<td>$q_0$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>$q_2$</td>
<td>$q_2$</td>
</tr>
</tbody>
</table>
3. \( L(M_3) = L(M_1) \cap L(M_2) \): Product construct

Product construction:

1. \( Q_3 = \{ e, o \} \times \{ q_0, q_1, q_2 \} = \{(e, q_0), (e, q_1), (e, q_3), (o, q_0), (o, q_1), (o, q_2), \} \)
2. \( \Sigma = \{ a, b \} \)
3. \( s_3 = (s_1, s_2) = (e, q_0) \)
4. \( F_3 = F_1 \times F_2 = \{ e \} \times \{ q_0 \} = \{(e, q_0)\} \)
5. \( \delta_3 \) is given as the following table:

\[
\begin{array}{ccc}
(a, b) & a & b \\
(e, q_0) & (o, q_1) & (e, q_2) \\
(e, q_1) & (o, q_2) & (e, q_0) \\
(e, q_3) & (o, q_2) & (e, q_2) \\
(o, q_0) & (e, q_1) & (o, q_2) \\
(o, q_1) & (e, q_2) & (o, q_0) \\
(o, q_2) & (e, q_2) & (o, q_2) \\
\end{array}
\]
3. $L(M_3) = L(M_1) \cap L(M_2)$: Product construct
Some Closure Properties of Regular Sets

- Closure under intersection: $A \cap B$
- Closure under complement: $\sim A$
- Closure under union: $A \cup B = \sim (\sim A \cap \sim B)$

Devise a way to construct new automaton

- intersection: the product construction
- complement: reverse accept and reject states
Question 3. Construct a finite automaton that accepts all binary strings without 111 as substring.

1. \( L(M') = \{ x \in \{0, 1\}^* \mid x \text{ accepts all binary strings with 111 as a substring} \} \)
Question 3. Construct a finite automaton that accepts all binary strings without 111 as substring.

1. \( L(M') = \{ x \in \{0, 1\}^* \mid x \text{ accepts all binary strings with} 111 \text{ as a substring} \} \)

2. \( L(M) = \sim L(M') \): reverse accept and reject states
1. \( L(M') = \{ x \in \{0, 1\}^* | x \text{ accepts all binary strings with } 111 \text{ as a substring} \} \)
1. $L(M') = \{ x \in \{0, 1\}^* | x \text{ accepts all binary strings with 111 as a substring} \}$
2. $L(M) \sim L(M')$: reverse accept and reject states
Exercise: Show that $L^2 = \{aw_1aaw_2a : w_1, w_2 \in \{a, b\}^*\}$ is regular.