Sipser: Chapter 1.1
Do not draw your sword to kill a fly.

-Korean Proverb
Finite State Automata
7-Tuple TM

\( Q \) is the set of states \((q_0, q_1, q_2, \ldots q_n)\)

\( q_0 \) is the initial state

\( \Gamma \) is the tape alphabet \(( \_ \in \Gamma, \Sigma \subseteq \Gamma)\)

\( \delta \) is the transition function \((Q \times \Gamma \rightarrow \Gamma \times \{L, R\} \times Q)\)

\( q_{\text{accept}} \) is the accept state(s)

\( q_{\text{reject}} \) is the reject state(s)

\( \Sigma \) is the input alphabet \(( \_ \not\in \Sigma )\)
7-Tuple TM

\( Q \) is the set of states \((q_0, q_1, q_2, \ldots q_n)\)

\( q_0 \in Q \) is the initial state

\( \Gamma \) is the tape alphabet \((\_ \in \Gamma, \Sigma \subseteq \Gamma)\)

\( \delta \) is the transition function \((Q \times \Gamma \rightarrow \Gamma \times \{L, R\} \times Q)\)

\( q_{\text{accept}} \in Q \) is the accept state(s)

\( q_{\text{reject}} \in Q \) is the reject state(s)

\( \Sigma \) is the input alphabet \((\_ \not\in \Sigma)\)
$\delta$ is the transition function $(Q \times \Gamma \rightarrow X \times \{X, R\} \times Q)$
$\delta$ is the transition function $(Q \times \Gamma \rightarrow \times \times \{L, R\} \times Q)$
TM that decides $L = \{ w \mid w \text{ ends with } 1 \}$
TM that decides $L = \{ w \mid w \text{ ends with } 1 \}$
FSA that decides \( L = \{ w \mid w \ \text{ends with 1} \} \)
Turing Machine
Finite State Automata
FSA that decides $L = \{ w \mid w \text{ ends with } 1 \}$
5-Tuple FSA

\( Q \) is the set of states \((q_0, q_1, q_2, \ldots q_n)\)

\( q_0 \in Q \) is the initial state

\( \Sigma \) is the input alphabet \((\_ \not\in \Sigma)\)

\( \delta \) is the transition function \((Q \times \Sigma \rightarrow Q)\)

\( q_{\text{accept}} \in Q \) is the accept state(s)
FSA that decides $L = \{ w \mid w \text{ starts with } 1 \}$
\( L = \{ 0 \{ 1, 2 \} \} \)
\[ L = \{ 0 \{1, 2\} \} \]
$L = \{ 0 \{1, 2\} \}$
$L = \{ 0 \{1, 2\} 2^n \mid n \geq 0 \}$
\[ L = \{ \{0, 1, 2\}^n \mid \text{where the digits sum to a product of 3} \ (n \geq 0) \} \]
L = \{ \{0, 1, 2\}^n \mid \text{where the digits sum to a product of 3} \} (n \geq 0)
\[ L_1 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]

\[ L_2 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]
\[ L_1 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]

\[ L_2 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]
\[ L_1 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]
\[ L_2 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]
\[ L_1 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \} \]
$L_2 = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \}$
If a language can be decided by an FSA...

It is called *REGULAR*. 
FSA that decides $L = \{ w \mid w \text{ ends with } 1 \}$
L = \{ \{0, 1, 2\}^n \mid \text{where the digits sum to a product of 3} \ (n \geq 0) \}\
L = \{ 0 \{1, 2\}^n \mid \text{where } n \geq 0 \}
Regular Languages, closed under...

1.) **Union** $L_{AUB} = \{ x \mid x \in L_A \text{ or } x \in L_B \}$
$L_{AUB} = \{ x \mid x \in L_A \text{ or } x \in L_B \}$
\[ L_{A\cup B} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \]
\[ L_{A\cup B} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \]
\[ L_{\text{AUB}} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \]
\[ L_1 \cup L_2 \]
L_1 \cup L_2
Regular Languages, closed under...

1.) **Union**  \( L_{A \cup B} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \)

2.) **Concatenation**  
\[
L_{A \cdot B} = \{ xy \mid x \in L_A \text{ and } y \in L_B \}
\]

3.) **Star**  \( L_A^* = \{ x_1x_2\ldots x_n \mid x_i \in L_A \} \)
\[ L = \{ \{0, 1, 2\}^n \mid \text{where the digits sum to a product of 3} \ (n \geq 0) \} \]
L = \{ \{0, 1, 2\}^* \mid \text{where the digits sum to } \} \\
\text{a product of 3}
L = \{ [0 \ 1 \ 2]^* \mid \text{where the digits sum to a product of 3}\}
\[ L = \{ (0 \ 1 \ 2)^* \mid \text{where the digits sum to} \ a \text{ product of } 3 \} \]

012
012012
012012012
012012012
\[ L = \{ w \mid w \text{ is of the form } (01)^*2 \} \]
L = \{ w \mid w \text{ is of the form } (01)^*2 \}\}
\[ L = \{ w \mid w \text{ is of the form } (01)^*2 \} \]
\[ \{ w \mid w \text{ is of the form } (01)^{2} \} \]

\((M \mid m) \text{ onroe}
[a-z]^{*}@[a-z]^{*}.com
[0-9]^{3}[-][0-9]^{3}[-][0-9]^{4} \]
Regular Language

Finite State Automaton

Regular Expression
L = \{ w \mid w \text{ is of the form } (01)^* 2 \}
Regular Languages, closed under...

1.) **Union**  \( L_{A \cup B} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \)

2.) **Concatenation**  
\[
L_{A \cdot B} = \{ xy \mid x \in L_A \text{ and } y \in L_B \}
\]

3.) **Star**  
\[ L_A^* = \{ x_1x_2\ldots x_n \mid x_i \in L_A \} \]
Regular Languages, closed under...

1.) **Union**  \( L_{A \cup B} = \{ x \mid x \in L_A \text{ or } x \in L_B \} \)

2.) **Concatenation**  
\[
L_A \cdot B = \{ xy \mid x \in L_A \text{ and } y \in L_B \}
\]

3.) **Star**  \( L_A^* = \{ x_1x_2\ldots x_n \mid x_i \in L_A \} \)
$\Sigma = \{0, 1\}$
\[ \Sigma = \{ a, b \} \]
$L_1 \cdot L_b = A \text{ string ending in 1 followed by a string ending in b}$
\[ \Sigma = \{0, 1\} \]

\[ \Sigma = \{a, b\} \]
\[ \Sigma = \{0, 1\} \]

\[ \Sigma = \{0, b\} \]