3.2
Constructing DFAs
How do we design a DFA $M$ for a given language $L$? That is $L(M) = L$.

- DFA is a like a program that has fixed amount of memory independent of input size.
- The memory of a DFA is encoded in its states
- The state/memory must capture enough information from the input seen so far that it is sufficient for the suffix that is yet to be seen (note that DFA cannot go back)
Example I: Basic languages

Assume $\Sigma = \{0, 1\}$.
$L = \emptyset, \quad L = \Sigma^*, \quad L = \{\epsilon\}, \quad L = \{0\}$. 
Assume \( \Sigma = \{0, 1\} \).

\[ L = \{w \in \{0, 1\}^* \mid |w| \text{ is divisible by } 5 \} \]
Assume $\Sigma = \{0, 1\}$.

$L = \{w \in \{0, 1\}^* \mid w \text{ ends with } 01\}$
Assume $\Sigma = \{0, 1\}$.

$L = \{w \in \{0, 1\}^* \mid w \text{ contains } 001 \text{ as substring}\}$
Assume $\Sigma = \{0, 1\}$.

$L = \{w \in \{0, 1\}^* \mid w \text{ contains 001 or 010 as substring}\}$
Example VI: Has a 1 exactly k positions from end

Assume $\Sigma = \{0, 1\}$.

$L = \{w \mid w \text{ has a 1 } k \text{ positions from the end}\}$. 
DFA Construction: Example

\[ L = \{ \text{Binary numbers congruent to 0 mod 5} \} \]

Example:

1. \[1101011_2 = 107_{10} = 2 \mod 5,\]
2. \[1010_2 = 10 = 0 \mod 5\]

Key observation:

\[ \text{val}(w) \mod 5 = a \text{ implies} \]

\[ \begin{align*}
\text{val}(w0) \mod 5 &= (\text{val}(w) \cdot 2) \mod 5 = 2a \mod 5 \\
\text{val}(w1) \mod 5 &= (\text{val}(w) \cdot 2 + 1) \mod 5 = (2a + 1) \mod 5
\end{align*} \]
DFA Construction: Example

$L = \{ \text{Binary numbers congruent to } 0 \mod 5 \}$

Example:
1. $1101011_2 = 107_{10} = 2 \mod 5$,
2. $1010_2 = 10 = 0 \mod 5$

**Key observation:**
$val(w) \mod 5 = a$ implies

\[
val(w_0) \mod 5 = (val(w) \times 2) \mod 5 = 2a \mod 5
\]

\[
val(w_1) \mod 5 = (val(w) \times 2 + 1) \mod 5 = (2a + 1) \mod 5
\]
THE END
...
(for now)