\( A = \{a, b\} \quad C = \{x, y\} \)
\( B = \{0, 1\} \)
\[(A \times B) \times C = \{(a, 0, x), \ldots\} \quad 3\]
\[A \times (B \times C) = \{(a, (0, y)), \ldots\} \quad 3\]
\[A \times B \times C = \{(a, 0, x), \ldots\} \quad 3\]

**alphabet:** finite set of symbols \( \Sigma \)

**string:** finite sequence of symbols (empty string too)

(can think of as a function from \( \{0, \ldots, l-1\} \) to \( \Sigma \))

\( \{c\} \quad w_c \) cat
\( w(0) = c \quad w_0 = c \quad w_1 = a \quad w_2 = \varepsilon \)
\( w(1) = a \quad w(2) = \varepsilon \)

\( l = \) length of string = \(|w|\)

**string concatenation:** \( w, v \) are strings over some alphabet \( \Sigma \)
$$w_0v = w_0, \ldots, w_{m-1} \; v_0 \ldots v_{m-1}$$

A string $u$ is a substring of $w$ iff $w = xuy$ for some strings $x, y$.

A string $u$ is a prefix of $w$ iff $w = uxv$ for some strings $x, y$.

A string $u$ is a suffix of $w$ iff $w = xuv$ for some strings $x, y$.

$w^R =$ reverse of $w = w_{m-1} \ldots w_0$

Language = set of strings

Concatenation of languages: $L_1 \circ L_2 = L_1L_2$

$$= \{ w \mid w = uv \text{ where } u \in L_1 \text{ and } v \in L_2 \}$$

Ex: $\Sigma = \{ a, b \}$

$L_1 = \{ a, b \}$

$L_2 = \{ a, b \}$

$L_1 \circ L_2 = \{ aa, ab, ba, bb \}$
\[ L_3 = \{ e, a, ab \} \]
\[ L_4 = \{ b, e \} \]
\[ L_3 L_4 = \{ e, b, a, ab, abb \} \]

Kleene star: \[ L^* = \{ w \mid w = w_1 w_2 \ldots w_k \text{ s.t. } k \geq 0 \text{ and } w_1, w_2, \ldots, w_k \in L \} \]

\[ L_3^* = \{ e, a, aa, aaa, \ldots \} \]
\[ aababaab \in L_3^* \]
\[ \epsilon \in L_3^* \]
\[ aabb \not\in L_3^* \]

\[ L_3^* = \{ w \mid w \text{ doesn't start with } b \text{ and doesn't have } 2 \text{ } b \text{'s in a row } \text{ or more} \} \]

Prove \[ S = \{ w \mid w \text{ has each } b \text{ preceded by } a \} \]

Prove \[ S = L_3^* \]
Proof: $L_3^* \subseteq S$ : Suppose $w \in L_3^*$

Suppose $w_k = b$, then

$w$ must have been $x_1 x_2 \cdots x_k a b y_1 \cdots y_k$

so $b @ w_k$ is preceded by an $a$

$\therefore w \in S$

$S \subseteq L_3^*$: Suppose $w \in S$

Let $\{x_1, \ldots, x_k\}$ be the positions of the $b$'s in $w$. Then let

$\{y_1, \ldots, y_k\}$ be the $a$'s between each $b$

$= \{x_1, x_2 - x_1, x_3 - x_2, \ldots, x_k - x_{k-1}\}$

Now $w = a^a_1 a a_2 \cdots a_a_1 a a_2 \cdots a_a_1 a b \cdots$

$y_i - 1$ times $y_i - 1$ times

$\Sigma^* = \text{set of all strings over } \Sigma$
Any decision problem is a language!

Let $P$ = problem of, given $x, y, z$, determining if $x \cdot y = z$ as a language, thus might be the set of strings

$L = \{1 \cdot 1 \cdot 1, 10 \cdot 10 \cdot 100, \ldots\}$

$s \notin L$ since $1 \cdot 2 \neq 5$

Circuit-EQUIV is problem of, given 2 combinational circuits, determining if the output is always the same

\[ a \rightarrow \text{D} \quad \text{is equiv to} \quad b \rightarrow \text{D} \]

\[ b \quad \text{c} \rightarrow \text{D} \quad \text{is equiv to} \quad c \rightarrow \text{D} \]
as a language, we use encoding of circuits

problem becomes \{ w - v | w and v are encodings of equivalent circuits \}

\[ a \land b \quad \quad \quad \quad a \land b \quad (a \land b) \lor (\neg a \land b) \]

\[ a \land b \quad \quad \quad \quad a \land b \quad (a \land b) \lor (\neg a \land b) \]