LANGUAGES THAT ARE and ARE NOT REGULAR

**Pumping Lemma**

We know that there are \( t \) languages over an alphabet \( \Sigma \), and there are only \( 2^\Sigma \) regular languages.

So we have \( t \leq 2^\Sigma \).

Languages that **ARE NOT REGULAR**

Let's look at some simple examples of languages that might **not be regular**.

Regular languages have a simple, repetitive structure:

\( (u, *, *) \) \( \sim \) periodic

**Det**: \( \text{Reg lang} \equiv \text{Reg expressions} \)
REGULAR / NOT REGULAR LANGUAGES

1. \( L_1 = a^* b^* \) is regular because it is defined by a regular expression.

2. \( L_2 = \{a^n b^n : n \geq 0\} \) is not regular and we will prove it in few minutes, using PUMPING LEMMA.

Intuitively: we can't construct a FINITE AUTOMATA \( M \) for \( L_2 \) because it would need to have a MEMORY to count and compare \# of a's and \# of b's.
We will have a new class of automata: **Push Down Automata** to accommodate the "memory" problem.

1. \[ L_3 = a^* \] is **REGULAR**
   \[ L_4 = \{a^m : m \geq 0 \} \] is regular
   \[ L_4 = L_3 \]

2. \[ L_5 = \{a^m : m \text{ prime} \} \] is **not** regular
to be proved by Pumping Lemma

3. \[ L_6 = \{a^m : m \text{ even} \} \] is **REGULAR**

\[ L = (aa)^* \]
$L_7 = \{ \omega \in \{a,b\}^* : \omega \text{ has an equal number of } a\text{'s and } b\text{'s} \}$

**Proof:**
Assume $L_7$ regular, so $L = L_7 \cap L_1 = L_7 \cap a^* b^*$ must be regular.

Thus, regular languages are closed under $\cap$.

But $L = L_7 \cap a^* b^* = \{ a^n b^n : n > 0 \} = L_2$.

Contradiction with fact that $\{ a^n b^n : n \geq 0 \}$ is not regular.
1. \( L_8 = \{ \text{ww}^R : w \in \{a,b\}^* \} \)  
   Proof by Pumping Lemma

2. \( L_9 = \{ \text{ww} : w \in \{a,b\}^* \} \)
\( L_{9'} = \{ \text{w} \in \{a,b\}^* \} \)
   Proof by Pumping Lemma

3. \( L_{10} = \{ \text{ww} \in \{a,b\}^* : \text{w stands for w in which each occurrence of a is replaced by } b \text{ and vice versa} \} \)
   Proof by Pumping Lemma
Let \( L' = \{ xy : x \in L \land y \in L \text{ regular} \} \)
\( L \) is over \( \Sigma \), \( x, y \in \Sigma^* \)

Proof:
\[ L_{11} = L \cdot \bar{L} \]
\[ \bar{L} = \{ \epsilon \} \cup L \]
\[ L \text{ is regular, so is } L' \text{ (proved theorem) and regular languages are closed under concatenation (same theorem).} \]

Remember: THEOREM:
The class of regular languages is closed under \( \cup, \cap, \cdot, \circ \).
Example

$L = \{a, aa, ab, ba, aba, ha\}$

$L_{12} = \{w \in L \cap \{a, b\}^* : w^R \in L\}$

$S = \{a, b, aab\}$

General: $L$ is regular

$L = \{a, aa, ab, ba, aba, ha\}$

$L_{12} = \{a, b, aab\}$

$L_{12} = \emptyset$
\[ L_{12} = \{ \omega \in \Sigma^* : \omega \in L \land \omega^R \in L \} \land L \text{ is regular over } \Sigma \]

**Observe**

\[ L_{12} = L \cap L^R \]

\[ \text{for } L^R = \{ \omega^R : \omega \in L \} \]

and \( L^R \) is \text{regular} \hspace{1cm} \text{proof}

\( L \)-reg, so \( L = L(M) \), \( M = (K, \Sigma, \Delta, s, F) \)

we define \( M' = (K', \Sigma, \Delta', s', F') \) as follows:

\[ k' = K \cup \{ s' \}, \quad s' = 2, \quad 2' \in K \]

\[ F' = \{ s' \} \]

\[ \Delta' = \{(2, e, 2) : (p, s, 2) \in \Delta \} \cup \{(s', e, 2) : \exists \in F' \} \]

all transitions REVERSED
\[ L_{13} = \{ w^R : w \in L^R \text{ is regular} \} = L^R \text{ for } L \text{ regular} \]

Picture proof: \( L_{\text{REGULAR}} M = (K, \Sigma, S, \Delta_m, F) \)

\[ L = L(M) \]

\[ \Delta_m^{-1} = \{ (q, \sigma, p) : (p, \delta, q) \in \Delta_m \} \cup \{ (s', e, 2) : \exists e \forall \sigma \in \Sigma \} \]
Theorem

Any finite language is regular.

Proof. Let \( L = \{ \omega_1, \omega_2, \ldots, \omega_m \} \subseteq \Sigma^* \) with \( n \geq 0 \).

Construct \( M \), such that

\[
L(M) = L = L_{\omega_1} \cup L_{\omega_2} \cup \ldots \cup L_{\omega_m} = \{ \omega_1 \} \cup \{ \omega_2 \} \cup \ldots \cup \{ \omega_m \}
\]

We construct

\[
M \xrightarrow{\omega_i} S_0 \xrightarrow{\omega_i} S_0 \cup \ldots \cup \omega_i \\
M = M_{\omega_1} \cup M_{\omega_2} \cup \ldots \cup M_{\omega_m}
\]