Example

\[ L = \{a^nba^n \mid n > 0\} \]

Closure Properties

A set is closed over an operation if

\[ L_1, L_2 \in \text{class} \]
\[ L_1 \text{ op } L_2 = L_3 \]
\[ \Rightarrow L_3 \in \text{class} \]

Example

\[ L = \{x \mid x \text{ is a positive even integer}\} \]

L is closed under

- addition?
- multiplication?
- subtraction?
- division?

Closure of Regular Languages

**Theorem 4.1** If \( L_1 \) and \( L_2 \) are regular languages, then

\[ L_1 \cup L_2 \]
\[ L_1 \cap L_2 \]
\[ L_1L_2 \]
\[ L_1 \hat{\cup} \]
\[ L_1^* \]

are regular languages.
Proof (sketch)

$L_1$ and $L_2$ are regular languages
$\Rightarrow \exists$ reg. expr. $r_1$ and $r_2$ s.t.
$L_1 = L(r_1)$ and $L_2 = L(r_2)$
$r_1 + r_2$ is r.e. denoting $L_1 \cup L_2$
$\Rightarrow$ closed under union
$r_1 r_2$ is r.e. denoting $L_1 L_2$
$\Rightarrow$ closed under concatenation
$r_1^*$ is r.e. denoting $L_1^*$
$\Rightarrow$ closed under star-closure

complementation:
$L_1$ is reg. lang.
$\Rightarrow \exists$ DFA $M$ s.t. $L_1 = L(M)$
Construct $M'$ s.t.

intersection:
$L_1$ and $L_2$ are reg. lang.
$\Rightarrow \exists$ DFA $M_1$ and $M_2$ s.t.
$L_1 = L(M_1)$ and $L_2 = L(M_2)$
$M_1 = (Q, \Sigma, \delta_1, q_0, F_1)$
$M_2 = (P, \Sigma, \delta_2, p_0, F_2)$
Construct $M' = (Q', \Sigma, \delta', (q_0, p_0), F')$
$Q' = \delta'$:
Regular languages are closed under

- reversal \( L^R \)
- difference \( L_1 - L_2 \)
- right quotient \( L_1 / L_2 \)
- homomorphism \( h(L) \)

**Right quotient**

Def: \( L_1 / L_2 = \{ x | xy \in L_1 \text{ for some } y \in L_2 \} \)

Example:

\[
\begin{align*}
L_1 &= \{ a^* b^* \cup b^* a^* \} \\
L_2 &= \{ b^n | n \text{ is even, } n > 0 \} \\
L_1 / L_2 &=
\end{align*}
\]

**Theorem** If \( L_1 \) and \( L_2 \) are regular, then \( L_1 / L_2 \) is regular.

**Proof** (sketch)

\[\exists \text{ DFA } M = (Q, \Sigma, \delta, q_0, F) \text{ s.t. } L_1 = L(M).\]

Construct DFA \( M' = (Q, \Sigma, \delta, q_0, F') \)

For each state \( i \) do

Make \( i \) the start state (representing \( L_1' \))

QED.
Homomorphism

Def. Let $\Sigma, \Gamma$ be alphabets. A homomorphism is a function

$$h : \Sigma \rightarrow \Gamma^*$$

Example:

$\Sigma = \{a, b, c\}, \Gamma = \{0, 1\}$

- $h(a) = 11$
- $h(b) = 00$
- $h(c) = 0$

$h(bc) =$

$h(ab^*) =$

Questions about regular languages:

L is a regular language.

- Given L, $\Sigma$, $w \in \Sigma^*$, is $w \in L$?

- Is L empty?

- Is L infinite?

- Does $L_1 = L_2$?
Ch. 4.3 - Identifying Nonregular Languages

If a language $L$ is finite, is $L$ regular?

If $L$ is infinite, is $L$ regular?

- $L_1 = \{a^n b^m | n > 0, m > 0\} = \epsilon$
- $L_2 = \{a^n b^n | n > 0\}$

Prove that $L_2 = \{a^n b^n | n > 0\}$ is?
**Pumping Lemma:** Let \( L \) be an infinite regular language. \( \exists \) a constant \( m > 0 \) such that any \( w \in L \) with \(|w| \geq m\) can be decomposed into three parts as \( w = xyz \) with

\[
\begin{align*}
|xy| & \leq m \\
|y| & \geq 1 \\
x y^i z & \in L \quad \text{for all } i \geq 0
\end{align*}
\]

**Meaning:** Every long string in \( L \) (the constant \( m \) above corresponds to the finite number of states in \( M \) in the previous proof) can be partitioned into three parts such that the middle part can be “pumped” resulting in strings that must be in \( L \).

**To Use the Pumping Lemma to prove \( L \) is not regular:**

- **Proof by Contradiction.**
  
  Assume \( L \) is regular.
  
  \( \Rightarrow \) \( L \) satisfies the pumping lemma.

  Choose a long string \( w \) in \( L \), \(|w| \geq m\). (The choice of the string is crucial. Must pick a string that will yield a contradiction).

  Show that there is NO division of \( w \) into \( xyz \) (must consider all possible divisions) such that \(|xy| \leq m\), \(|y| \geq 1\) and \( xy^i z \in L \ \forall \ i \geq 0\).

  The pumping lemma does not hold. Contradiction!

  \( \Rightarrow \) \( L \) is not regular. QED.

**Example** \( L = \{a^n b^n | n > 0 \} \)

\( L \) is not regular.

- **Proof:**
  
  Assume \( L \) is regular.

  \( \Rightarrow \) the pumping lemma holds.

  Choose \( w = \) where \( m \) is the constant in the pumping lemma. (Note that \( w \) must be chosen such that \(|w| \geq m\).)

  The only way to partition \( w \) into three parts, \( w = xyz \), is such that \( x \) contains 0 or more \( a \)'s, \( y \) contains 1 or more \( a \)'s, and \( z \) contains 0 or more \( a \)'s concatenated with \( c b^m \). This is because of the restrictions \(|xy| \leq m\) and \(|y| > 0\). So the partition is:

  It should be true that \( xy^i z \in L \) for all \( i \geq 0\).
Example \( L = \{ a^n b^{n+s} c^s | n, s > 0 \} \)

L is not regular.

- **Proof:**
  Assume \( L \) is regular.
  \( \Rightarrow \) the pumping lemma holds.
  Choose \( w = \)
  The only way to partition \( w \) into three parts, \( w = xyz \), is such that \( x \) contains 0 or more \( a \)'s, \( y \) contains 1 or more \( a \)'s, and \( z \) contains 0 or more \( a \)'s concatenated with the rest of the string \( b^{m+s} c^s \).
  This is because of the restrictions \( |xy| \leq m \) and \( |y| > 0 \). So the partition is:

Example \( \Sigma = \{ a, b \}, L = \{ w \in \Sigma^* \mid n_a(w) > n_b(w) \} \)

L is not regular.

- **Proof:**
  Assume \( L \) is regular.
  \( \Rightarrow \) the pumping lemma holds.
  Choose \( w = \)
  So the partition is:
Example  \( L = \{a^3b^n c^{n-3} | n > 3 \} \)

\( L \) is not regular.

- **Proof:**
  Assume \( L \) is regular. \( \Rightarrow \) the pumping lemma holds.

  Choose \( w = a^3b^m c^{m-3} \) where \( m \) is the constant in the pumping lemma. There are three ways to partition \( w \) into three parts, \( w = xyz \).

  1) \( y \) contains only \( a \)'s
  2) \( y \) contains only \( b \)'s
  3) \( y \) contains \( a \)'s and \( b \)'s

  We must show that each of these possible partitions lead to a contradiction. (Then, there would be no way to divide \( w \) into three parts s.t. the pumping lemma contraints were true).

  **Case 1:** (\( y \) contains only \( a \)'s). Then \( x \) contains 0 to 2 \( a \)'s, \( y \) contains 1 to 3 \( a \)'s, and \( z \) contains 0 to 2 \( a \)'s concatenated with the rest of the string \( b^m c^{m-3} \), such that there are exactly 3 \( a \)'s. So the partition is:

  \[
  x = a^k \quad y = a^j \quad z = a^{3-k-j} b^m c^{m-3}
  \]

  where \( k \geq 0, j > 0, \) and \( k + j \leq 3 \) for some constants \( k \) and \( j \).

  It should be true that \( xy^i z \in L \) for all \( i \geq 0 \).

  \( xy^2 z = (x)(y)(y)(z) = (a^k)(a^j)(a^j)(a^{3-j-k}b^m c^{m-3}) = a^{3+j} b^m c^{m-3} \not\in L \) since \( j > 0 \), there are too many \( a \)'s. Contradiction!

  **Case 2:** (\( y \) contains only \( b \)'s) Then \( x \) contains 3 \( a \)'s followed by 0 or more \( b \)'s, \( y \) contains 1 to \( m - 3 \) \( b \)'s, and \( z \) contains 3 to \( m - 3 \) \( b \)'s concatenated with the rest of the string \( c^{m-3} \). So the partition is:

  \[
  x = a^3 b^k \quad y = b^j \quad z = b^{m-k-j} c^{m-3}
  \]

  where \( k \geq 0, j > 0, \) and \( k + j \leq m - 3 \) for some constants \( k \) and \( j \).

  It should be true that \( xy^i z \in L \) for all \( i \geq 0 \).

  \( xy^2 z = a^3 b^{m-j} c^{m-3} \not\in L \) since \( j > 0 \), there are too few \( b \)'s. Contradiction!

  **Case 3:** (\( y \) contains \( a \)'s and \( b \)'s) Then \( x \) contains 0 to 2 \( a \)'s, \( y \) contains 1 to 3 \( a \)'s, and 1 to \( m - 3 \) \( b \)'s, \( z \) contains 3 to \( m - 1 \) \( b \)'s concatenated with the rest of the string \( c^{m-3} \). So the partition is:

  \[
  x = a^{3-k} \quad y = a^k b^j \quad z = b^{m-j} c^{m-3}
  \]

  where \( 3 \geq k > 0, \) and \( m - 3 \geq j > 0 \) for some constants \( k \) and \( j \).

  It should be true that \( xy^i z \in L \) for all \( i \geq 0 \).

  \( xy^2 z = a^3 b^j a^k b^m c^{m-3} \not\in L \) since \( j, k > 0 \), there are \( b \)'s before \( a \)'s. Contradiction!

  \( \Rightarrow \) There is no partition of \( w \).

  \( \Rightarrow \) \( L \) is not regular!. QED.
To Use Closure Properties to prove L is not regular:

Using closure properties of regular languages, construct a language that should be regular, but for which you have already shown is not regular. Contradiction!

- **Proof Outline:**
  Assume L is regular.
  Apply closure properties to L and other regular languages, constructing L’ that you know is not regular.
  closure properties ⇒ L’ is regular.
  Contradiction!
  L is not regular. QED.

**Example** \( L = \{a^{3b^n}c^{n-3} | n > 3 \} \)

L is not regular.

- **Proof:** (proof by contradiction)
  Assume L is regular.
  Define a homomorphism \( h : \Sigma \rightarrow \Sigma^* \)
  \( h(a) = a \quad h(b) = a \quad h(c) = b \)
  \( h(L) = \)
Example: $L = \{a^n b^m a^m | m \geq 0, n \geq 0\}$

$L$ is not regular.

- **Proof:** (proof by contradiction)
  
  Assume $L$ is regular.

Example: $L_1 = \{a^n b^n a^n | n > 0\}$

$L_1$ is not regular.

- **Proof:**
  
  Assume $L_1$ is regular.
  
  Goal is to try to construct $\{a^n b^n | n > 0\}$ which we know is not regular.
  
  Let $L_2 = \{a^*\}$. $L_2$ is regular.
  
  By closure under right quotient, $L_3 = L_1 \setminus L_2 = \{a^n b^n a^p | 0 \leq p \leq n, n > 0\}$ is regular.
  
  By closure under intersection, $L_4 = L_3 \cap \{a^* b^*\} = \{a^n b^n | n > 0\}$ is regular.
  
  Contradiction, already proved $L_4$ is not regular!
  
  Thus, $L_1$ is not regular. QED.