Section: Properties of Regular Languages

Example

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

Closure Properties

A set is closed over an operation if

\[ \text{L}_1, \text{L}_2 \in \text{class} \]
\[ \text{L}_1 \text{ op } \text{L}_2 = \text{L}_3 \]
\[ \Rightarrow \text{L}_3 \in \text{class} \]
$L = \{x \mid x \text{ is a positive even integer}\}$

$L$ is closed under

- addition? ✓
- multiplication? ✓
- subtraction? ❌
- division? ❌

Closure of Regular Languages

Theorem 4.1 If $L_1$ and $L_2$ are regular languages, then

- $L_1 \cup L_2$
- $L_1 \cap L_2$
- $L_1 L_2$
- $\bar{L}_1$
- $L_1^*$

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
$r_1r_2$ is r.e. denoting $L_1L_2$
$\Rightarrow$ closed under concatenation
$r_1^*$ is r.e. denoting $L_1^*$
$\Rightarrow$ closed under star-closure
complementation:

$L_1$ is reg. lang.

$\Rightarrow \exists \text{ DFA } M \text{ s.t. } L_1 = L(M)$

Construct $M'$ s.t.

Final states in $M$ are nonfinal states in $M'$
nonfinal states in $M$ are final states in $M'$

Note we must start with a DFA
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 $M' = (Q', \Sigma, \delta', (q_0, p_0), F')$

$Q' = Q \times P$

$\delta': \quad \text{d for delta}$

$d'((q_i, p_j), a) = (q_k, p_l)$ if

$d1(q_i, a) = q_k$ and

$d2(p_j, a) = p_l$

$F' = \{(q_i, p_j) \mid q_i \text{ in } F_1, \text{ and } p_j \text{ in } F_2\}$
Example:
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:

$L_1 = \{a^*b^* \cup b^*a^*\}$
$L_2 = \{b^n | n \text{ is even, } n > 0\}$
$L_1/L_2 = \{a^*b^*\}$
Theorem If $L_1$ and $L_2$ are regular, then $L_1/L_2$ is regular.

Proof (sketch)

$\exists$ DFA $M=(Q, \Sigma, \delta, q_0, F)$ 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 $L'_i$)

If $L_1'$ intersect $L_2$ is not the empty set then put $q_i$ in $F'$ in $M'$. 

QED.
Homomorphism

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

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

Example:

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

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

$$h(bc) = 000$$

$$h(ab^*) = 11(00)^*$$
Questions about regular languages:
L is a regular language.

- Given L, Σ, w ∈ Σ*, is w ∈ L?

  Construct a DFA for it, if there is a path from the initial state to a final state, then it is accepted.

- Is L empty?

  If there is a path to any final state, do a DFS (depth first search), then not empty

- Is L infinite?

  there is a dfa for it, determine if there is a loop or not.

- Does L₁ = L₂?

  Construct L₃ = (L₁ intersect Complement(L₂)) union (complement of L₁ intersect L₂)

  If L₃ is empty, then L₁ = L₂.
Identifying Nonregular Languages

If a language $L$ is finite, is $L$ regular?
Yes, put all strings together in a regular expression.

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

- $L_1 = \{a^n b^m | n > 0, m > 0\} = a^*b^*$
- $L_2 = \{a^n b^n | n > 0\}$ not regular
Prove that $L_2 = \{a^n b^n | n > 0\}$ is?

- **Proof:** Suppose $L_2$ is regular.
  $\Rightarrow \exists$ DFA $M$ that recognizes $L_2$

  $M$ has a finite number of states, $k$ states

  Consider a long string: $a^k b^k$ in $L_2$

  Since there are $k$ states and $k$ a's, means there must be some state reached more than once

  that means, there is a loop with one or more a's in the loop along the path

  Means, we could go through that loop an extra time

  That is, we start in the start state to process $a^k b^k$ and reach a final state, but instead we go through the loop of a's an extra time, and still reach the final state.

  contradiction!
  $L$ is not regular

  go through loop extra time:
  $a^k a^k b^k$
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

- $|xy| \leq m$ the loop is in the first $m$ chars
- $|y| \geq 1$ y is the loop
- $xy^iz \in L$ for all $i \geq 0$
To Use the Pumping Lemma to prove L is not regular:

- Proof by Contradiction.
  Assume L is regular.
  ⇒ L satisfies the pumping lemma.
  Choose a long string \( w \) in L, \(|w| \geq m\).
  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!
  ⇒ L is not regular. QED.
Example $L = \{ a^n c b^n | n > 0 \}$

$L$ is not regular.

- **Proof:**
  
  Assume $L$ is regular.
  
  $\Rightarrow$ the pumping lemma holds.
  
  Choose $w = a^m c b^m$

  We want to show there is NO way to partition $w$ into three parts so that the three conditions hold.

  Here is a general partition that handles every case.

  $$x = a^k \quad y = a^j \quad \text{with} \quad j > 0$$

  $$z = a^{m-k-j} c b^m$$
It should be true that xy^iz is in L for all i

find an i where it does not work, i = 0, or i > 1 will not work

i = 2: xyyz = a^{m+j} c b^m \notin L

Contradiction!
L is not regular
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 $w = a^m b^{m+s} c^s$
  
  So the partition is:
  
  $x = a^k, \quad y = a^t, \quad z = a^{m-t-k} b^{m+s} c^s$

  $i = 0$

  $xy^0z = xz = a^k b^{m+s} c^s$

  number of $b$'s is not equal to the number of $a$'s and $c$'s
  
  $xz$ is not in $L$
  
  **Contradiction!**
Example $\Sigma = \{a, b\}$,
$L = \{ w \in \Sigma^* | n_a(w) > n_b(w) \}$

$L$ is not regular.

- Proof:
  Assume $L$ is regular.
  $\Rightarrow$ the pumping lemma holds.
  Choose $w =$
  So the partition is:
Example $L = \{a^3b^n c^{n-3} | n > 3\}$
(shown in detail on handout)
$L$ is not regular.
To Use Closure Properties to prove $L$ is not regular:

- **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 $\Rightarrow L'$ is regular.
  Contradiction!
  $L$ is not regular. QED.
Example \( L = \{a^3b^n c^{n-3} \mid n > 3\} \)

\( L \) 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.