

Lecture 9 – The Pumping Lemma for Regular Languages

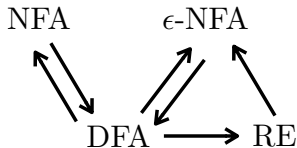
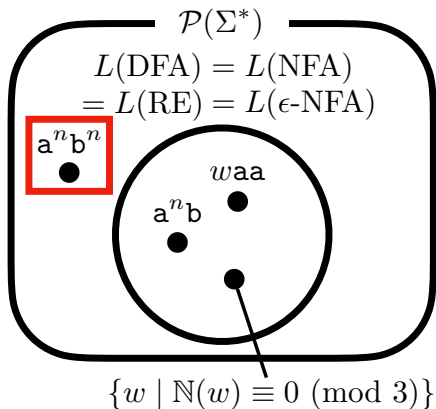
COSE215: Theory of Computation

Jihyeok Park



2024 Spring

- Not all languages are regular: e.g., $L = \{a^n b^n \mid n \geq 0\}$.



- How to prove that a language is **NOT** regular? **Pumping Lemma!**

1. Pumping Lemma for Regular Languages

Intuition

Pumping Lemma

Proof of Pumping Lemma

Application: Proving Languages are Not Regular

2. Examples

Example 1: $L = \{a^n b^n \mid n \geq 0\}$

Example 2: $L = \{ww^R \mid w \in \{a, b\}^*\}$

Example 3: $L = \{a^l b^m c^n \mid l + m \leq n\}$

Example 4: $L = \{a^{n^2} \mid n \geq 0\}$

Example 5: $L = \{a^n b^k c^{n+k} \mid n, k \geq 0\}$

1. Pumping Lemma for Regular Languages

Intuition

Pumping Lemma

Proof of Pumping Lemma

Application: Proving Languages are Not Regular

2. Examples

Example 1: $L = \{a^n b^n \mid n \geq 0\}$

Example 2: $L = \{ww^R \mid w \in \{a, b\}^*\}$

Example 3: $L = \{a^l b^m c^n \mid l + m \leq n\}$

Example 4: $L = \{a^{n^2} \mid n \geq 0\}$

Example 5: $L = \{a^n b^k c^{n+k} \mid n, k \geq 0\}$

Let's consider a **regular language** L .

Then, there exists a **regular expression** R such that $L(R) = L$.

Then, there are two possibilities:

- 1 R does **not contain** the **Kleene star** operator $(^*)$.

$abc \qquad ab|(bc\epsilon)|d \qquad (a|bcd)\epsilon|\emptyset bca$

Then, the language L should be **finite**.

- 2 R **contains** at least one Kleene star operator $(^*)$.

$$R_1 R_2^* R_3$$

Roughly speaking, we can **repeat** the middle part R_2 as many times as we want (including 0 times).

The **Pumping Lemma** formally captures this intuition.

Lemma (Pumping Lemma for Regular Languages)

For a given regular language L , **there exists** a *positive integer* n such that **for all** $w \in L$, if $|w| \geq n$, **there exists** $w = xyz$ such that

- ① $|y| > 0$
- ② $|xy| \leq n$
- ③ $\forall i \geq 0. xy^iz \in L$

$A = L \text{ is regular}$



$B = \exists n > 0. \forall w \in L. |w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}$

- Let L be a regular language.
- Then, \exists DFA $D = (Q, \Sigma, \delta, q_0, F)$. s.t. $L(D) = L$. Let $n = |Q| > 0$.
- Take any $w = a_1 a_2 \cdots a_m \in L$ s.t. $|w| = m \geq n$.
- Let $p_i = \delta^*(q_0, a_1 \cdots a_i)$ for all $0 \leq i \leq m$. Then, $p_0 = q_0 \wedge p_m \in F$.

$$q_0 = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_m} p_m \in F$$

- Let L be a regular language.
- Then, \exists DFA $D = (Q, \Sigma, \delta, q_0, F)$. s.t. $L(D) = L$. Let $n = |Q| > 0$.
- Take any $w = a_1 a_2 \cdots a_m \in L$ s.t. $|w| = m \geq n$.
- Let $p_i = \delta^*(q_0, a_1 \cdots a_i)$ for all $0 \leq i \leq m$. Then, $p_0 = q_0 \wedge p_m \in F$.

$$q_0 = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_m} p_m \in F$$

$\underbrace{\hspace{10em}}_{n+1 \text{ states}}$

- By Pigeonhole Principle, there exists $i < j \leq n$ s.t. $p_i = p_j$.

$$q_0 = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_i} p_i \xrightarrow{a_{i+1}} \cdots \xrightarrow{a_j} p_j \xrightarrow{a_{j+1}} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_m} p_m \in F$$

$\overbrace{\hspace{10em}}^x$ $\overbrace{\hspace{10em}}^y$ $\overbrace{\hspace{10em}}^z$

- We can split $w = xyz$ as above. Then,

$$|y| = j - i > 0 \qquad |xy| = j \leq n$$

- Since y represents a **cycle** from p_i to p_i , $\forall i \geq 0$. $\delta^*(q_0, xy^i z) = p_m$.
- It means that $\forall i \geq 0$. $xy^i z \in L$.

- Let L be a regular language.
- Then, \exists DFA $D = (Q, \Sigma, \delta, q_0, F)$. s.t. $L(D) = L$. Let $n = |Q| > 0$.
- Take any $w = a_1 a_2 \cdots a_m \in L$ s.t. $|w| = m \geq n$.
- Let $p_i = \delta^*(q_0, a_1 \cdots a_i)$ for all $0 \leq i \leq m$. Then, $p_0 = q_0 \wedge p_m \in F$.

$$q_0 = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_m} p_m \in F$$

$\underbrace{\hspace{10em}}_{n+1 \text{ states}}$

- By Pigeonhole Principle, there exists $i < j \leq n$ s.t. $p_i = p_j$.

$$q_0 = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_i} p_i \xrightarrow{a_{i+1}} \cdots \xrightarrow{a_j} p_j \xrightarrow{a_{j+1}} \cdots \xrightarrow{a_n} p_n \xrightarrow{a_{n+1}} \cdots \xrightarrow{a_m} p_m \in F$$

$\overbrace{\hspace{10em}}^x \quad \overbrace{\hspace{10em}}^y \quad \overbrace{\hspace{10em}}^z$

- We can split $w = xyz$ as above. Then,

$$\textcircled{1} - |y| = j - i > 0 \qquad |xy| = j \leq n - \textcircled{2}$$

- Since y represents a **cycle** from p_i to p_i , $\forall i \geq 0$. $\delta^*(q_0, xy^i z) = p_m$.
- It means that $\forall i \geq 0$. $xy^i z \in L$. - $\textcircled{3}$

Lemma (Pumping Lemma for Regular Languages)

For a given regular language L , **there exists** a *positive integer* n such that **for all** $w \in L$, if $|w| \geq n$, **there exists** $w = xyz$ such that

- ① $|y| > 0$
- ② $|xy| \leq n$
- ③ $\forall i \geq 0. xy^iz \in L$

$A = L \text{ is regular}$



$B = \exists n > 0. \forall w \in L. |w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}$

Lemma (Pumping Lemma for Regular Languages)

$$A = L \text{ is regular}$$



$$B = \exists n > 0. \forall w \in L. |w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}$$

$$A \Rightarrow B \quad (O)$$

$$B \Rightarrow A \quad (X)$$

$$\neg B \Rightarrow \neg A \quad (O)$$

$$\begin{aligned} \neg B &= \neg(\exists n > 0. \forall w \in L. |w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}) \\ &= \forall n > 0. \neg(\forall w \in L. |w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}) \\ &= \forall n > 0. \exists w \in L. \neg(|w| \geq n \Rightarrow \exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}) \\ &= \forall n > 0. \exists w \in L. |w| \geq n \wedge \neg(\exists w = xyz. \textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}) \\ &= \forall n > 0. \exists w \in L. |w| \geq n \wedge \forall w = xyz. \neg(\textcircled{1} \wedge \textcircled{2} \wedge \textcircled{3}) \\ &= \forall n > 0. \exists w \in L. |w| \geq n \wedge \forall w = xyz. \neg(\textcircled{1} \wedge \textcircled{2}) \vee \neg\textcircled{3} \\ &= \forall n > 0. \exists w \in L. |w| \geq n \wedge \forall w = xyz. (\textcircled{1} \wedge \textcircled{2}) \Rightarrow \neg\textcircled{3} \end{aligned}$$

To prove a language L is **NOT** regular, we need to show that

$$\forall n > 0. \exists w \in L. |w| \geq n \wedge \forall w = xyz. (\textcircled{1} \wedge \textcircled{2}) \Rightarrow \neg \textcircled{3}$$

$$\textcircled{1} \quad |y| > 0$$

$$\textcircled{2} \quad |xy| \leq n$$

$$\textcircled{3} \quad \forall i \geq 0. xy^i z \in L$$

Note that $\neg \textcircled{3} = \exists i \geq 0. xy^i z \notin L$.

We can prove this by following the steps below:

① Assume **any** positive integer n is given.

② **Pick** a word $w \in L$.

③ Show that $|w| \geq n$.

④ Assume **any** split $w = xyz$ is given, and $\textcircled{1} \quad |y| > 0 \wedge \textcircled{2} \quad |xy| \leq n$.

⑤ $\neg \textcircled{3}$ **Pick** $i \geq 0$, and show that $xy^i z \notin L$ using $\textcircled{1}$ and $\textcircled{2}$.

1. Pumping Lemma for Regular Languages

Intuition

Pumping Lemma

Proof of Pumping Lemma

Application: Proving Languages are Not Regular

2. Examples

Example 1: $L = \{a^n b^n \mid n \geq 0\}$

Example 2: $L = \{ww^R \mid w \in \{a, b\}^*\}$

Example 3: $L = \{a^l b^m c^n \mid l + m \leq n\}$

Example 4: $L = \{a^{n^2} \mid n \geq 0\}$

Example 5: $L = \{a^n b^k c^{n+k} \mid n, k \geq 0\}$

Example 1

Let's prove that L is **NOT** regular using the Pumping Lemma:

$$L = \{a^n b^n \mid n \geq 0\}$$

- ① Assume any positive integer n is given.
- ② Let $w = a^n b^n \in L$.
- ③ $|w| = n + n = 2n \geq n$.
- ④ Assume any split $w = xyz$ is given, and ① $|y| > 0 \wedge$ ② $|xy| \leq n$.
- ⑤ Let $i = 0$. We need to show that \neg ③ $xy^0z \notin L$:
 - Since ② $|xy| \leq n$,

$$x = a^p \quad y = a^q \quad z = a^{n-p-q} b^n$$

for some $0 \leq p, q \leq n$.

- Since ① $|y| > 0$, we know $q > 0$.
- Finally, $xy^0z = xz = a^p a^{n-p-q} b^n = a^{n-q} b^n \notin L$ ($\because q > 0$).



Example 2

Let's prove that L is **NOT** regular using the Pumping Lemma:

$$L = \{ww^R \mid w \in \{a,b\}^*\}$$

- ① Assume any positive integer n is given.
- ② Let $w = a^n b^n b^n a^n \in L$.
- ③ $|w| = n + n + n + n = 4n \geq n$.
- ④ Assume any split $w = xyz$ is given, and ① $|y| > 0 \wedge$ ② $|xy| \leq n$.
- ⑤ Let $i = 0$. We need to show that \neg ③ $xy^0z \notin L$:
 - Since ② $|xy| \leq n$,

$$x = a^p \quad y = a^q \quad z = a^{n-p-q} b^n b^n a^n$$

for some $0 \leq p, q \leq n$.

- Since ① $|y| > 0$, we know $q > 0$.
- Finally, $xy^0z = xz = a^p a^{n-p-q} b^n b^n a^n = a^{n-q} b^n b^n a^n \notin L$
($\because q > 0$).



Example 3

Let's prove that L is **NOT** regular using the Pumping Lemma:

$$L = \{a^l b^m c^n \mid l + m \leq n\}$$

- ① Assume any positive integer n is given.
- ② Let $w = a^n b^n c^{2n} \in L$.
- ③ $|w| = n + n + 2n = 4n \geq n$.
- ④ Assume any split $w = xyz$ is given, and ① $|y| > 0 \wedge$ ② $|xy| \leq n$.
- ⑤ Let $i = 2$. We need to show that \neg ③ $xy^2z \notin L$:
 - Since ② $|xy| \leq n$,

$$x = a^p \quad y = a^q \quad z = a^{n-p-q} b^n c^{2n}$$

for some $0 \leq p, q \leq n$.

- Since ① $|y| > 0$, we know $q > 0$.
- Finally, $xy^2z = xyyz = a^{n+q} b^n c^{2n} \notin L$
($\because q > 0$. Thus, $(n + q) + n = 2n + q > 2n$).



Example 4

Let's prove that L is **NOT** regular using the Pumping Lemma:

$$L = \{a^{n^2} \mid n \geq 0\}$$

- ① Assume any positive integer n is given.
- ② Let $w = a^{n^2} \in L$.
- ③ $|w| = n^2 \geq n$.
- ④ Assume any split $w = xyz$ is given, and ① $|y| > 0 \wedge$ ② $|xy| \leq n$.
- ⑤ Let $i = 2$. We need to show that \neg ③ $xy^2z \notin L$:
 - Since ① $|y| > 0$ and ② $|xy| \leq n$,

$$y = a^k$$

where $0 < k \leq n$. Then,

$$n^2 < n^2 + k \quad (\because 0 < k) \qquad n^2 + k < (n+1)^2 \quad (\because k \leq n)$$

- Finally, $xy^2z = xyxz = a^{n^2+k} \notin L$



Example 5

Let's prove that L is **NOT** regular:

$$L = \{a^n b^k c^{n+k} \mid n, k \geq 0\}$$

- It is much easier to use **closure properties** under **homomorphisms**.
- Consider a homomorphism $h : \{a, b, c\} \rightarrow \{a, b\}^*$:

$$h(a) = a \quad h(b) = a \quad h(c) = b$$

- Then,

$$h(L) = \{a^{n+k} b^{n+k} \mid n, k \geq 0\} = \{a^n b^n \mid n \geq 0\}$$

- If L is regular, then $h(L)$ must be regular as well.
- However, we know $h(L)$ is **NOT** regular.
- Therefore, L is **NOT** regular. □

1. Pumping Lemma for Regular Languages

Intuition

Pumping Lemma

Proof of Pumping Lemma

Application: Proving Languages are Not Regular

2. Examples

Example 1: $L = \{a^n b^n \mid n \geq 0\}$

Example 2: $L = \{ww^R \mid w \in \{a, b\}^*\}$

Example 3: $L = \{a^l b^m c^n \mid l + m \leq n\}$

Example 4: $L = \{a^{n^2} \mid n \geq 0\}$

Example 5: $L = \{a^n b^k c^{n+k} \mid n, k \geq 0\}$

- Please see this document on GitHub:

<https://github.com/ku-plrg-classroom/docs/tree/main/cose215/equiv-re-fa>

- The due date is 23:59 on Apr. 17 (Wed.).
- Please implement the following functions in `Implementation.scala`.
 - `reToENFA` for the conversion from REs to ϵ -NFAs.
 - `dfaToRE` for the conversion from DFAs to REs.
 - `enfaToDFA` for the conversion from ϵ -NFAs to DFAs.
- Please only submit `Implementation.scala` file to [Blackboard](#).

- Equivalence and Minimization of Finite Automata

Jihyeok Park

jihyeok_park@korea.ac.kr

<https://plrg.korea.ac.kr>