

Lecture 9 — Properties of Regular Languages (2): Pumping Lemma

CSE205: Theory of Computation

Minseok Jeon

2026 Fall

Some Fundamental Questions

So far we have studied regular languages. But some fundamental questions remain:

- **Are all languages regular?**
 - No. For example, $L = \{ a^n b^n \mid n \geq 0 \}$ is *not* regular.
- **How can we prove that a language is non-regular?** Two methods:
 1. Direct proof using the *Pigeonhole principle*.
 2. Using the *Pumping Lemma*.

The intuition behind both: a DFA has *finitely many states* but must process *arbitrarily long* strings, so it cannot “count” without bound.

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

Direct proof (by contradiction, Pigeonhole principle).

Pigeonhole principle: if more than n pigeons are placed into n holes, some hole holds more than one pigeon.

- Assume L is regular. Then there is a DFA $M = (Q, \Sigma, \delta, q_0, F)$ recognizing L .
- Let **Pigeons** $= \{a^n \mid n \geq 0\} = \{a, aa, aaa, \dots\}$ and **Holes** $= Q$.
- Put pigeon a^n into hole $\hat{\delta}(q_0, a^n)$, the state reached on input a^n .
- There are $|Q|$ holes but infinitely many pigeons, so two distinct pigeons a^i and a^j ($i \neq j$) land in the same hole: they reach the same state.

Since M accepts $a^i b^i$, it must also accept $a^j b^i$ (the prefixes a^i and a^j end in the same state). But $a^j b^i \notin L$ when $i \neq j$ — a contradiction. Hence L is non-regular.

Example 2: $L = \{ww \mid w \in \{0, 1\}^*\}$ is non-regular

Direct proof (by contradiction, Pigeonhole principle).

- Assume L is regular, so there is a DFA $M = (Q, \Sigma, \delta, q_0, F)$ recognizing L .
- Let **Pigeons** $= \{0^i1 \mid i \geq 0\} = \{1, 01, 001, \dots\}$ and **Holes** $= Q$.
- Put pigeon 0^i1 into hole $\hat{\delta}(q_0, 0^i1)$.
- By the Pigeonhole principle, two pigeons share a hole: 0^i1 and 0^j1 with $i \neq j$ reach the same state.

M accepts $0^i10^j1 \in L$. Since 0^i1 and 0^j1 reach the same state, M also accepts 0^j10^i1 . But for $i \neq j$ this string is not of the form ww , so $0^j10^i1 \notin L$ — a contradiction. Hence L is non-regular.

The Pumping Lemma

Theorem (Pumping Lemma for Regular Languages)

For any regular language L there exists an integer $n \geq 1$ (the pumping length) such that for every string $x \in L$ with $|x| \geq n$, there exist $u, v, w \in \Sigma^*$ with

$$x = uvw$$

such that

1. $|uv| \leq n$,
2. $|v| \geq 1$ (so $v \neq \epsilon$),
3. for all $i \geq 0$, $uv^i w \in L$.

The substring v can be “pumped” (repeated, or deleted with $i = 0$) any number of times and the result stays in L .

Proof of the Pumping Lemma

- Let M be a DFA for L with $n = |Q|$ states.
- Take any $x \in L$ with $|x| \geq n$; write $x = a_1 a_2 \cdots a_m$ ($m = |x| \geq n$).
- Let $p_i = \hat{\delta}(q_0, a_1 a_2 \cdots a_i)$ be the state after reading i symbols; note $p_0 = q_0$ and $p_m \in F$.
- Consider the first $n + 1$ states $p_0, p_1, \dots, p_n$. There are only n states, so by the Pigeonhole principle two of them coincide: $p_i = p_j$ for some $0 \leq i < j \leq n$.

Split $x = uvw$ with

$$u = a_1 \cdots a_i, \quad v = a_{i+1} \cdots a_j, \quad w = a_{j+1} \cdots a_m.$$

Then $|uv| = j \leq n$ and $|v| = j - i \geq 1$. Since $\hat{\delta}(p_i, v) = p_j = p_i$, the loop on v can be traversed any number of times:

$$\hat{\delta}(q_0, uv^k w) = p_m \in F \quad \text{for all } k \geq 0.$$

Using the Pumping Lemma to Prove Non-regularity

The Pumping Lemma says: *if L is regular, then L satisfies the pumping property.*

Taking the contrapositive:

L does not satisfy the pumping property $\implies L$ is not regular.

Warning. The converse is false: a non-regular language may still satisfy the pumping property. So the lemma can be used *only* to prove that a language is **not** regular, never that it *is* regular.

The Pumping Lemma as an Adversary Game

To prove L is non-regular, view the proof as a game against an adversary:

1. **Adversary** picks the pumping length n (we do not know its value).
2. **We** choose a specific string $x \in L$ with $|x| \geq n$, cleverly, so that no split can survive pumping.
3. **Adversary** splits $x = uvw$ with $|uv| \leq n$ and $|v| \geq 1$.
4. **We** pick some $i \geq 0$ and show $uv^i w \notin L$.

If we can always win step 4 *no matter how* the adversary splits, then L violates the lemma and is therefore not regular. The art is choosing x in step 2; the constraint $|uv| \leq n$ then pins v to a known region of x .

Example 1: $L = \{0^i 1^i \mid i \geq 0\}$ is not regular

Proof. Suppose L is regular; let n be the pumping length. Choose

$$x = 0^n 1^n \in L, \quad |x| = 2n \geq n.$$

By the lemma, $x = uvw$ with $|uv| \leq n$ and $|v| \geq 1$.

Since $|uv| \leq n$, both u and v lie within the block of 0's, so

$$u = 0^s, \quad v = 0^t, \quad w = 0^p 1^n,$$

with $s + t \leq n$, $t \geq 1$, $p \geq 0$, and $s + t + p = n$.

Pump down with $i = 0$:

$$uv^0w = uw = 0^s 0^p 1^n = 0^{s+p} 1^n \notin L,$$

because $s + p = n - t < n$, so the number of 0's no longer equals the number of 1's.
Contradiction; hence L is not regular. □

Example 2: $L = \{ww^R \mid w \in \{a, b\}^*\}$ is not regular

Proof. Suppose L is regular with pumping length n . Choose

$$x = a^n b^n b^n a^n \in L \quad (\text{here } w = a^n b^n), \quad |x| = 4n \geq n.$$

By the lemma, $x = uvw$ with $|uv| \leq n$ and $|v| \geq 1$. Since $|uv| \leq n$, both lie in the leading block of a 's:

$$u = a^s, \quad v = a^t, \quad w = a^p b^n b^n a^n,$$

with $s + t \leq n$, $t \geq 1$, $s + t + p = n$.

Pump down with $i = 0$:

$$uv^0w = a^s a^p b^n b^n a^n = a^{s+p} b^{2n} a^n \notin L,$$

since $s + p < n$, breaking the palindrome structure. Contradiction; hence L is not regular. □

Example 3: $L = \{w \in \{a, b\}^* \mid n_a(w) < n_b(w)\}$

Here $n_a(w)$ and $n_b(w)$ count the a 's and b 's in w .

Proof. Suppose L is regular with pumping length n . Choose

$$x = a^n b^{n+1} \in L \quad (n < n+1), \quad |x| = 2n+1 \geq n.$$

By the lemma, $x = uvw$ with $|uv| \leq n$, $|v| \geq 1$, so v lies in the a 's:

$$u = a^s, \quad v = a^t, \quad w = a^p b^{n+1}, \quad s+t \leq n, \quad t \geq 1, \quad s+t+p = n.$$

Pump up with $i = 2$:

$$uv^2w = a^s a^{2t} a^p b^{n+1} = a^{s+2t+p} b^{n+1}.$$

Since $s+2t+p = n+t \geq n+1$, we have $n_a \geq n+1 \geq n_b$, so $uv^2w \notin L$.

Contradiction; hence L is not regular. □

Example 4: $L = \{a^n \mid n \text{ is a perfect square}\}$

Proof. Suppose L is regular with pumping length n . Choose

$$x = a^{n^2} \in L, \quad |x| = n^2 \geq n.$$

By the lemma, $x = uvw$ with $|uv| \leq n$ and $|v| \geq 1$, so $v = a^k$ with $1 \leq k \leq n$.

Pump down with $i = 0$:

$$uv^0w = a^{n^2-k}.$$

But $n^2 - k$ is *not* a perfect square, because it lies strictly between two consecutive squares:

$$(n-1)^2 = n^2 - 2n + 1 < n^2 - k < n^2,$$

using $1 \leq k \leq n < 2n - 1$. Hence $uv^0w \notin L$. Contradiction; L is not regular. □

Example 5: Using Closure under Homomorphism

Claim. $L = \{ a^n b^k c^{n+k} \mid n \geq 0, k \geq 0 \}$ is not regular.

We could apply the Pumping Lemma directly, but it is even easier to use closure under *homomorphism*. Define h by

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

Then

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

which we already know is *not* regular.

Since regular languages are closed under homomorphism, if L were regular then $h(L)$ would be regular. By contraposition, L is not regular. □

The Converse of the Pumping Lemma is False

The pumping property is *necessary* but *not sufficient* for regularity. Consider

$$L = \{ a^i b^j c^k \mid i, j, k \geq 0, \text{ if } i = 1 \text{ then } j = k \}.$$

L satisfies the pumping lemma (pumping length $n = 2$): for $x = a^i b^j c^k \in L$ with $|x| \geq 2$,

- if $i \neq 1$: choose v (length 1 or 2) among the leading a 's, or, if $i = 0$, anywhere in $b^j c^k$, so that pumping never produces exactly one a ; the result keeps $i \neq 1$ (or stays at $i = 0$), so it lies in L regardless of j, k ;
- if $i = 1$: the only valid split is $v = a$ itself; pumping it changes the a -count away from 1 for every exponent except the original, so the result is again automatically in L .

The Converse of the Pumping Lemma is False (cont.)

Yet L is **not regular**: intersecting with the regular language $L(ab^*c^*)$ gives

$$L \cap L(ab^*c^*) = \{ab^ic^i \mid i \geq 0\},$$

which is not regular (the homomorphism $a \mapsto \epsilon$, $b \mapsto a$, $c \mapsto b$ maps it to $\{a^ib^i\}$, already known non-regular). If L were regular, closure under intersection would make this language regular too — contradiction.

Takeaway. Satisfying the pumping lemma does not guarantee regularity. For an exact (necessary and sufficient) characterization of regular languages, see the *Myhill–Nerode theorem*.

Summary

- Not all languages are regular; e.g. $\{a^n b^n\}$, $\{ww\}$, $\{a^{n^2}\}$.
- **Pigeonhole principle** gives direct non-regularity proofs: a finite-state machine cannot distinguish infinitely many “pigeons”.
- The **Pumping Lemma** packages this idea: every long enough string in a regular language contains a pumpable loop v with $|uv| \leq n$, $|v| \geq 1$, and $uv^i w \in L$ for all i .
- Use it (as an adversary game) to prove non-regularity; closure properties such as homomorphism can simplify the argument.
- The converse fails — for an exact characterization use the Myhill–Nerode theorem.