Proving That a Language Is Not Regular
Three ways to represent regular languages (so far)
Three ways to represent regular languages (so far)

▶ DFA
Three ways to represent regular languages (so far)

▶ DFA
▶ NFA
Three ways to represent regular languages (so far)

- DFA
- NFA
- Regular expressions
Three ways to represent regular languages (so far)

- DFA
- NFA
- Regular expressions

To prove that a language is not regular it is easiest to use DFA’s.
Three ways to represent regular languages (so far)

- DFA
- NFA
- Regular expressions

To prove that a language is not regular it is easiest to use DFA’s.

Why?
Two Methods of Proof

▶ Method 1: Run the DFA on many small words. By the pigeon hole principle two of the words must finish in the same state. Then do some magic.

▶ Method 2 (Pumping Lemma): Run the DFA on one long word. By the pigeon hole principle the word must visit the same state twice. Then do some magic.
Two Methods of Proof

- **Method 1**: Run the DFA on many small words. By the **pigeon hole principle** two of the words must finish in the same state. Then do some **magic**.
Two Methods of Proof

- **Method 1:** Run the DFA on many small words. By the **pigeon hole principle** two of the words must finish in the same state. Then do some **magic**.

- **Method 2 (Pumping Lemma):** Run the DFA on one long word. By the **pigeon hole principle** the word must visit the same state twice. Then do some **magic**.
Method 1
Have already used Method 1. When? To prove lower bounds for number of states for DFA's.

\[
\left\{ a \cup b \right\}^* a \left\{ a \cup b \right\}^n : 2^n + 1.
\]

\[
a^n : n + 1.
\]
Have already used Method 1.
Have already used Method 1.

When?
Have already used Method 1.

**When?**

To prove lower bounds for **number of states** for DFA’s.
Have already used Method 1.

When?

To prove lower bounds for **number of states** for DFA’s.

\[ \{a \cup b\}^*a\{a \cup b\}^n: \]
Have already used Method 1.

**When?**

To prove lower bounds for **number of states** for DFA’s.

\[ (a \cup b)^* a (a \cup b)^n : 2^{n+1}. \]
Have already used Method 1.

When?

To prove lower bounds for **number of states** for DFA’s.

- \( \{a \cup b\}^* a \{a \cup b\}^n \): \(2^{n+1}\).

- \( a^n \):
Have already used Method 1.

When?

To prove lower bounds for **number of states** for DFA’s.

- \( \{a \cup b\}^* a\{a \cup b\}^n : 2^{n+1} \).
- \( a^n : n \).
$L_1 = a^n b^n : n \geq 0$ is Not Regular
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Intuition**

DFA's only have finite memory. A DFA has to "remember" the length of an arbitrarily long sequence of $a$'s when processing the $b$'s.

Intuition is not proof.
Intuition

- DFA’s only have finite memory.
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Intuition**

- DFA’s only have finite memory.
- A DFA has to “remember” the length of an arbitrarily long sequence of $a$’s when processing the $b$’s.
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Intuition**

- DFA’s only have finite memory.
- A DFA has to “remember” the length of an arbitrarily long sequence of $a$’s when processing the $b$’s.

**Intuition is not proof.**
\( L_1 = a^n b^n : n \geq 0 \) is Not Regular
$L_1 = a^n b^n : n \geq 0$ is Not Regular

Proof
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states.
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. 
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. By the pigeon hole principle two of the inputs, say $a^i$ and $a^j$ ($i \neq j$), must be in the same state. Append $b^i$ to both inputs. Finish running $M$ on both inputs. They will end up in the same state, but $a^i b^i$ should be accepted and $a^j b^i$ should not. Contradiction.
\[ L_1 = a^n b^n : n \geq 0 \text{ is Not Regular} \]

**Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^0, a^1, a^2, \ldots, a^m \). By the **pigeon hole principle** two of the inputs, say \( a^i \) and \( a^j \) \((i \neq j)\), must be in the same state. Append \( b^i \) to both inputs.

Contradiction.

Intuition
A DFA with \( m \) states can only "remember" \( m \) pieces of information. This idea is formalized in the Myhill-Nerode theorem. We do not care.
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. By the **pigeon hole principle** two of the inputs, say $a^i$ and $a^j$ ($i \neq j$), must be in the same state. Append $b^i$ to both inputs. Finish running $M$ on both inputs. Contradiction.

Intuition: A DFA with $m$ states can only "remember" $m$ pieces of information. This idea is formalized in the Myhill-Nerode theorem. We do not care.
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. By the **pigeon hole principle** two of the inputs, say $a^i$ and $a^j$ ($i \neq j$), must be in the same state. Append $b^i$ to both inputs. Finish running $M$ on both inputs. They will end up in the same state, but $a^i b^i$ should be accepted and $a^j b^i$ should not.
$L_1 = a^n b^n : n \geq 0$ is Not Regular

**Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. By the **pigeon hole principle** two of the inputs, say $a^i$ and $a^j$ ($i \neq j$), must be in the same state. Append $b^i$ to both inputs. Finish running $M$ on both inputs. They will end up in the same state, but $a^i b^i$ should be accepted and $a^j b^i$ should not. Contradiction.
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^0, a^1, a^2, \ldots, a^m$. By the pigeon hole principle two of the inputs, say $a^i$ and $a^j (i \neq j)$, must be in the same state. Append $b^i$ to both inputs. Finish running $M$ on both inputs. They will end up in the same state, but $a^i b^i$ should be accepted and $a^j b^i$ should not. Contradiction.

Intuition A DFA with $m$ states can only “remember” $m$ pieces of information.
\[ L_1 = a^n b^n : n \geq 0 \text{ is Not Regular} \]

**Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^0, a^1, a^2, \ldots, a^m \). By the **pigeon hole principle** two of the inputs, say \( a^i \) and \( a^j \) (\( i \neq j \)), must be in the same state. Append \( b^i \) to both inputs. Finish running \( M \) on both inputs. They will end up in the same state, but \( a^i b^i \) should be accepted and \( a^j b^i \) should not. Contradiction.

**Intuition** A DFA with \( m \) states can only “remember” \( m \) pieces of information.

This idea is formalized in the Myhill-Nerode theorem.
\[ L_1 = a^n b^n : n \geq 0 \] is Not Regular

**Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^0, a^1, a^2, \ldots, a^m \). By the **pigeon hole principle** two of the inputs, say \( a^i \) and \( a^j \) (\( i \neq j \)), must be in the same state. Append \( b^i \) to both inputs. Finish running \( M \) on both inputs. They will end up in the same state, but \( a^i b^i \) should be accepted and \( a^j b^i \) should not. Contradiction.

**Intuition** A DFA with \( m \) states can only “remember” \( m \) pieces of information.

This idea is formalized in the Myhill-Nerode theorem.

We do not care.
Method 2: Pumping Lemma
$L_1 = a^n b^n : n \geq 0$ is Not Regular: Alternative Proof
Proof

$L_1 = a^n b^n : n \geq 0$ is Not Regular: Alternative Proof
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states.
**Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$.
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$S_0, S_1, S_2, \ldots, S_m$
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

\[ s_0, s_1, s_2, \ldots, s_m \]

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$’s.
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$s_0, s_1, s_2, \ldots, s_m$

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$’s. So there is a loop at that state where $k \geq 1$ $a$’s are processed.
\( L_1 = a^n b^n : n \geq 0 \) is Not Regular: Alternative Proof

**Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^m b^m \). States encountered processing \( a^m \):

\[ s_0, s_1, s_2, \ldots, s_m \]

By the **pigeon hole principle** the same state must have been encountered twice while still processing the \( a \)'s. So there is a loop at that state where \( k \geq 1 \) \( a \)'s are processed.

\[ q_0 \xrightarrow{a} q_1 \xrightarrow{a} \cdots \xrightarrow{a} q_i \xrightarrow{a} \cdots \xrightarrow{b} \cdots \xrightarrow{b} q_{m-1} \]
Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$$s_0, s_1, s_2, \ldots, s_m$$

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$’s. So there is a loop at that state where $k \geq 1$ $a$’s are processed.

That means $a^{n+k} b^n$ is also accepted by following the loop again. Contradiction.
\[ L_2 = \{ w : \#_a(w) = \#_b(w) \} \] is not regular.
Exactly the same Proof
\( L_2 = \{ w : \#_a(w) = \#_b(w) \} \) is not regular.

Exactly the same **Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states.
\[ L_2 = \{ w : \#_a(w) = \#_b(w) \} \text{ is not regular.} \]

Exactly the same **Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^m b^m \).
$L_2 = \{ w : \#_a(w) = \#_b(w) \}$ is not regular.

Exactly the same **Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$s_0, s_1, s_2, \ldots, s_m$
$L_2 = \{ w : \#_a(w) = \#_b(w) \}$ is not regular.

Exactly the same **Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. **Run** $M$ on $a^m b^m$. States encountered processing $a^m$:

$s_0, s_1, s_2, \ldots, s_m$

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$'s.
Let $L_2 = \{ w : \#a(w) = \#b(w) \}$ be not regular.

Exactly the same Proof Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$s_0, s_1, s_2, \ldots, s_m$

By the pigeon hole principle the same state must have been encountered twice while still processing the $a$'s. So there is a loop at that state where $k \geq 1$ $a$'s are processed.

Contradiction. This idea can be formalized into the pumping lemma ... and we will do so.
$L_2 = \{w : \#_a(w) = \#_b(w)\}$ is not regular.

Exactly the same **Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$s_0, s_1, s_2, \ldots, s_m$

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$’s. So there is a loop at that state where $k \geq 1$ $a$’s are processed. That means $a^{n+k} b^n$ is also accepted by following the loop again.
**Proof** Assume $L_1$ is regular via DFA $M$ with $m$ states. Run $M$ on $a^m b^m$. States encountered processing $a^m$:

$$s_0, s_1, s_2, \ldots, s_m$$

By the **pigeon hole principle** the same state must have been encountered twice while still processing the $a$’s. So there is a loop at that state where $k \geq 1$ $a$’s are processed. That means $a^{n+k} b^n$ is also accepted by following the loop again. Contradiction.
\[ L_2 = \{ w : \#_a(w) = \#_b(w) \} \text{ is not regular.} \]

Exactly the same Proof Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^m b^m \). States encountered processing \( a^m \):

\[ s_0, s_1, s_2, \ldots, s_m \]

By the pigeon hole principle the same state must have been encountered twice while still processing the \( a \)'s. So there is a loop at that state where \( k \geq 1 \) \( a \)'s are processed. That means \( a^{n+k} b^n \) is also accepted by following the loop again.

Contradiction.

This idea can be formalized into the pumping lemma ...
\[ L_2 = \{ w : \#_a(w) = \#_b(w) \} \text{ is not regular.} \]

Exactly the same **Proof** Assume \( L_1 \) is regular via DFA \( M \) with \( m \) states. Run \( M \) on \( a^m b^m \). States encountered processing \( a^m \):

\[ s_0, s_1, s_2, \ldots, s_m \]

By the **pigeon hole principle** the same state must have been encountered twice while still processing the \( a \)'s. So there is a loop at that state where \( k \geq 1 \) \( a \)'s are processed. That means \( a^{n+k} b^n \) is also accepted by following the loop again.

Contradiction.

**This idea can be formalized into the pumping lemma ...**

**... and we will do so.**
General Technique

Pumping Lemma

If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:

For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy| \leq n_1$.
3. For all $i \geq 0$, $xy^iz \in L$.

Proof by picture

$q_0 \cdots q_i \cdots q_{m-1} \sigma x y z \sigma \cdots \sigma$
General Technique

**Pumping Lemma** If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:

1. $w = xyz$ and $y \neq \varepsilon$.
2. $|xy| \leq n_1$.
3. For all $i \geq 0$, $xy^iz \in L$.
**General Technique**

**Pumping Lemma** If \( L \) is regular then there exist \( n_0 \) and \( n_1 \) such that the following holds:
For all \( w \in L, \ |w| \geq n_0 \) there exist \( x, y, z \) such that:

1. \( w = xyz \) and \( y \neq e \).
2. \( |xy| \leq n_1 \).
3. For all \( i \geq 0 \), \( xy^iz \in L \).

Proof by picture
General Technique

Pumping Lemma If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:
For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:
1. $w = xyz$ and $y \neq e$. 

Proof by picture

$q_0 \cdots q_i \cdots q_{m-1} \sigma x y z \sigma \cdots \sigma$
General Technique

**Pumping Lemma** If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:
For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:
1. $w = xyz$ and $y \neq e$.
2. $|xy| \leq n_1$. 
General Technique

**Pumping Lemma** If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:
For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy| \leq n_1$.
3. For all $i \geq 0$, $xy^iz \in L$. 

Proof by picture
General Technique

**Pumping Lemma** If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:
For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy| \leq n_1$.
3. For all $i \geq 0$, $xy^iz \in L$.

Proof by picture
General Technique

**Pumping Lemma** If $L$ is regular then there exist $n_0$ and $n_1$ such that the following holds:
For all $w \in L$, $|w| \geq n_0$ there exist $x, y, z$ such that:
1. $w = xyz$ and $y \neq e$.
2. $|xy| \leq n_1$.
3. For all $i \geq 0$, $xy^i z \in L$.

**Proof by picture**
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.

Pumping Lemma
If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy|$ is short.
3. for all $i$, $xy^iz \in L$.

We then find some $i$ such that $xy^iz \notin L$ for the contradiction.
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.

**Pumping Lemma** If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy|$ is short.
3. for all $i$, $xy^iz \in L$.

We then find some $i$ such that $xy^iz \notin L$ for the contradiction.
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.

**Pumping Lemma** If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$. 

We restate it in the way that we use it.

**Pumping Lemma** If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy|$ is short.
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.

**Pumping Lemma** If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy|$ is short.
3. for all $i$, $xy^iz \in L$. 

We then find some $i$ such that $xy^iz / \in L$ for the contradiction.
How We Use the Pumping Lemma (PL)

We restate it in the way that we use it.

**Pumping Lemma** If $L$ is reg then for large enough strings $w$ in $L$ there exist $x, y, z$ such that:

1. $w = xyz$ and $y \neq e$.
2. $|xy|$ is short.
3. for all $i$, $xy^iz \in L$.

We then find some $i$ such that $xy^iz \notin L$ for the contradiction.
REDO:  \( L_1 = \{a^n b^n : n \in \mathbb{N}\} \) is Not Regular
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular.
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$'s. $x = a^j, y = a^k, z = a^{n-j-k}b^n$. Note $k \geq 1$.

By the PL, all of the words $a^j a^k a^{n-j-k}b^n = a^{n+k(i-1)}b^n$ are in $L_1$.

Take $i = 2$ to get $a^n b^n \in L_1$.

Contradiction since $k \geq 1$. 
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$. 


REDO: $L_1 = \{ a^n b^n : n \in \mathbb{N} \}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.  

REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$’s.
**REDO:** $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$'s. $x = a^j$, $y = a^k$, $z = a^{n-j-k} b^n$. Note $k \geq 1$. 
**REDO:** \( L_1 = \{a^n b^n : n \in \mathbb{N}\} \) is Not Regular

Assume \( L_1 \) is regular. By the PL, for long enough string \( a^n b^n \in L_1 \), there exist \( x, y, z \) such that:

1. \( y \neq e \).
2. \( |xy| \) is short.
3. For all \( i \geq 0 \), \( xy^i z \in L_1 \).

Take \( w \) long enough so that the \( xy \) part only has \( a \)'s. \( x = a^j, y = a^k, z = a^{n-j-k} b^n \). Note \( k \geq 1 \).

By the PL, all of the words
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$’s. $x = a^j, y = a^k, z = a^{n-j-k} b^n$. Note $k \geq 1$.

By the PL, all of the words

$$a^j (a^k)^i a^{n-j-k} b^n$$
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.  
2. $|xy|$ is short.  
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$’s. $x = a^i$, $y = a^k$, $z = a^{n-j-k} b^n$. Note $k \geq 1$.

By the PL, all of the words

$$a^j (a^k)^i a^{n-j-k} b^n = a^{n+k(i-1)} b^n$$
**REDO:** \( L_1 = \{a^n b^n : n \in \mathbb{N} \} \) is Not Regular

Assume \( L_1 \) is regular. By the PL, for long enough string \( a^n b^n \in L_1 \), there exist \( x, y, z \) such that:

1. \( y \neq e \).
2. \( |xy| \) is short.
3. For all \( i \geq 0 \), \( xy^i z \in L_1 \).

Take \( w \) long enough so that the \( xy \) part only has \( a \)'s. \( x = a^j, y = a^k, z = a^{n-j-k} b^n \). Note \( k \geq 1 \).

By the PL, all of the words

\[
a^j (a^k)^i a^{n-j-k} b^n = a^{n+k(i-1)} b^n
\]

are in \( L_1 \).
REDO: $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$’s. $x = a^j$, $y = a^k$, $z = a^{n-j-k}b^n$. Note $k \geq 1$.

By the PL, all of the words

$$a^j \left(a^k\right)^i a^{n-j-k} b^n = a^{n+k(i-1)} b^n$$

are in $L_1$.

Take $i = 2$ to get
**REDO:** $L_1 = \{a^n b^n : n \in \mathbb{N}\}$ is Not Regular

Assume $L_1$ is regular. By the PL, for long enough string $a^n b^n \in L_1$, there exist $x, y, z$ such that:

1. $y \neq e$.
2. $|xy|$ is short.
3. For all $i \geq 0$, $xy^i z \in L_1$.

Take $w$ long enough so that the $xy$ part only has $a$’s. $x = a^j$, $y = a^k$, $z = a^{n-j-k} b^n$. Note $k \geq 1$.

By the PL, all of the words

$$a^j (a^k)^i a^{n-j-k} b^n = a^{n+k(i-1)} b^n$$

are in $L_1$.

Take $i = 2$ to get

$$a^{n+k} b^n \in L_1$$
REDO: \( L_1 = \{a^n b^n : n \in \mathbb{N}\} \) is Not Regular

Assume \( L_1 \) is regular. By the PL, for long enough string \( a^n b^n \in L_1 \), there exist \( x, y, z \) such that:

1. \( y \neq e \).
2. \(|xy| \) is short.
3. For all \( i \geq 0 \), \( xy^i z \in L_1 \).

Take \( w \) long enough so that the \( xy \) part only has \( a \)'s. \( x = a^j \), \( y = a^k \), \( z = a^{n-j-k} b^n \). Note \( k \geq 1 \).

By the PL, all of the words

\[
a^j (a^k)^i a^{n-j-k} b^n = a^{n+k(i-1)} b^n
\]

are in \( L_1 \).
Take \( i = 2 \) to get

\[
a^{n+k} b^n \in L_1
\]

Contradiction since \( k \geq 1 \).
$L_2 = \{ w : \#_a(w) = \#_b(w) \}$ is Not Regular

**Proof:** Same Proof as $L_1$ not Reg: Still look at $a^m b^m$.

**Key** Pumping Lemma says for ALL long enough $w \in L$. 
$L_3 = \{ w : \#_a(w) \neq \#_b(w) \}$ is Not Regular
$L_3 = \{ w : \#_a(w) \neq \#_b(w) \}$ is Not Regular

Think about.
$L_3 = \{w : \#_a(w) \neq \#_b(w)\}$ is Not Regular

Think about.

**Pumping Lemma Does Not Help.** When you increase the number of $y$'s there is no way to control it so carefully to make the number of $a$'s EQUAL the number of $b$'s.
$L_3 = \{ w : \#_a(w) \neq \#_b(w) \} \text{ is Not Regular}$

Think about.

**Pumping Lemma Does Not Help.** When you increase the number of $y$’s there is no way to control it so carefully to make the number of $a$’s EQUAL the number of $b$’s. So what do to?
L₃ = \{w : \#a(w) \neq \#b(w)\} is Not Regular

Think about.

**Pumping Lemma Does Not Help.** When you increase the number of y’s there is no way to control it so carefully to make the number of a’s EQUAL the number of b’s.

So what do to?

If L₃ is regular then L₂ = \overline{L₃} is regular. But we know that L₂ is not regular. DONE!
$L_4 = \{a^{n^2} : n \in \mathbb{N}\}$ is Not Regular

Intuition
Perfect squares keep getting further apart.
Pumping lemma says you can always add some constant $k$ to produce a word in the language.

Proof
By Pumping Lemma for long enough $a^{n^2} \in L_4$ there exist $x = a^j, y = a^k, z = a^\ell$ such that $a^j(a^k)^i a^\ell \in L_4$.

$(\forall i \geq 0)[j + ik + \ell \text{ is a square}]$.
In particular, $n^2 + k$ is a square.
Consider two consecutive squares: $n^2$ and $(n + 1)^2 = n^2 + 2n + 1$.
But if $2n + 1 > k$ then $n^2 + k$ is not.

Pick $n > (k - 1)/2$. 
\[ L_4 = \{a^{n^2} : n \in \mathbb{N}\} \text{ is Not Regular} \]

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant \( k \) to produce a word in the language.
$L_4 = \{a^{n^2} : n \in \mathbb{N}\}$ is Not Regular

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant $k$ to produce a word in the language.

**Proof**
$L_4 = \{ a^{n^2} : n \in \mathbb{N} \}$ is Not Regular

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant $k$ to produce a word in the language.

**Proof**
By Pumping Lemma for long enough $a^{n^2} \in L_4$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that $a^j(a^k)^ia^\ell \in L_4$. 

In particular, $n^2 + k$ is not a square.

Consider two consecutive squares: $n^2$ and $(n+1)^2 = n^2 + 2n + 1$.

But if $2n + 1 > k$ then $n^2 + k$ is not.
\[ L_4 = \{a^{n^2} : n \in \mathbb{N}\} \text{ is Not Regular} \]

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant \( k \) to produce a word in the language.

**Proof**

By Pumping Lemma for long enough \( a^{n^2} \in L_4 \) there exist \( x = a^j, y = a^k, z = a^\ell \) such that \( a^j(a^k)^i a^\ell \in L_4 \).

\[
(\forall i \geq 0)[j + ik + \ell \text{ is a square}].
\]
$L_4 = \{a^{n^2} : n \in \mathbb{N}\}$ is Not Regular

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant $k$ to produce a word in the language.

**Proof**
By Pumping Lemma for long enough $a^{n^2} \in L_4$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that $a^j(a^k)^i a^\ell \in L_4$.

$$\forall i \geq 0)[j + ik + \ell \text{ is a square}].$$

In particular, $n^2 + k$ is a square.
\( L_4 = \{a^{n^2} : n \in \mathbb{N}\} \) is Not Regular

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant \( k \) to produce a word in the language.

**Proof**
By Pumping Lemma for long enough \( a^{n^2} \in L_4 \) there exist \( x = a^j, y = a^k, z = a^\ell \) such that \( a^j(a^k)^i a^\ell \in L_4 \).

\[(\forall i \geq 0)[j + ik + \ell \text{ is a square}].\]

In particular, \( n^2 + k \) is a square.
Consider two consecutive squares:
\( n^2 \) and \( (n + 1)^2 = n^2 + 2n + 1 \).
$$L_4 = \{a^{n^2} : n \in \mathbb{N}\} \text{ is Not Regular}$$

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant $k$ to produce a word in the language.

**Proof**
By Pumping Lemma for long enough $a^{n^2} \in L_4$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that $a^j(a^k)^i a^\ell \in L_4$.

$$(\forall i \geq 0)[j + ik + \ell \text{ is a square}].$$

In particular, $n^2 + k$ is a square.

Consider two consecutive squares: $n^2$ and $(n + 1)^2 = n^2 + 2n + 1$.

But if $2n + 1 > k$ then $n^2 + k$ is not.
$L_4 = \{ a^{n^2} : n \in \mathbb{N} \}$ is Not Regular

**Intuition** Perfect squares keep getting further apart. Pumping lemma says you can always add some constant $k$ to produce a word in the language.

**Proof**
By Pumping Lemma for long enough $a^{n^2} \in L_4$ there exist $x = a^j, y = a^k, z = a^\ell$ such that $a^j(a^k)^i a^\ell \in L_4$.

$$(\forall i \geq 0)[j + ik + \ell \text{ is a square}].$$

In particular, $n^2 + k$ is a square.

Consider two consecutive squares: $n^2$ and $(n + 1)^2 = n^2 + 2n + 1$.

But if $2n + 1 > k$ then $n^2 + k$ is not.

Pick $n > (k - 1)/2$.  

\( L_5 = \{ a^p : p \text{ is prime} \} \) is Not Regular
$L_5 = \{ a^p : p \text{ is prime} \}$ is Not Regular

**Intuition** Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language.
$L_5 = \{a^p : p \text{ is prime}\}$ is Not Regular

**Intuition** Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language. **Too hard.** Easier proof.
Intuition Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language. **Too hard.** Easier proof.

Think about.
$L_5 = \{ a^p : p \text{ is prime} \}$ is Not Regular

**Intuition** Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language. **Too hard.** Easier proof.

**Think about.**

By Pumping Lemma, for large enough $p$, $a^p \in L_5$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that

$$a^j(a^k)^i a^\ell \in L_5$$

$$(\forall i \geq 0)[j + ik + \ell \text{ is prime}].$$
$L_5 = \{a^p : p \text{ is prime}\}$ is Not Regular

**Intuition** Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language. **Too hard.** Easier proof.

**Think about.**

By Pumping Lemma, for large enough $p$, $a^p \in L_5$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that

$$a^j(a^k)^i a^\ell \in L_5$$

$$(\forall i \geq 0)[j + ik + \ell \text{ is prime}].$$

So,

$$p, p + k, p + 2k, \ldots, p + pk$$

are all prime.
$L_5 = \{ a^p : p \text{ is prime} \}$ is Not Regular

**Intuition** Primes keep getting further apart on average. Pumping lemma says you always add some constant $k$ to produce a word in the language. Too hard. Easier proof.

**Think about.**

By Pumping Lemma, for large enough $p$, $a^p \in L_5$ there exist $x = a^j$, $y = a^k$, $z = a^\ell$ such that

$$a^j(a^k)^i a^\ell \in L_5$$

$$(\forall i \geq 0)[j + ik + \ell \text{ is prime}].$$

So,

$$p, p + k, p + 2k, \ldots, p + pk$$

are all prime.

But $p + pk = p(k + 1)$. 
We will be brief here.
$L_6 = \{ \#_a(w) > \#_b(w) \}$ is Not Regular

We will be brief here.
Take $w = b^n a^{n+1}$, long enough so the $y$-part is in the $b$'s.
We will be brief here.
Take \( w = b^n a^{n+1} \), long enough so the \( y \)-part is in the \( b \)'s.
Pump the \( y \) to get more \( b \)'s than \( a \)'s.
$L_7 = \{a^n b^m : n > m\}$ is Not Regular

Think about. Problematic. Can take $w$ long and pump $a$'s, but that won't get out of the language. So what to do? Revise Pumping Lemma. Pumping Lemma had a bound on $|xy|$. Can also bound $|yz|$ by same proof. Do that and then you can get $y$ to be all $b$'s, pump $b$'s, and get out of the language.
$L_7 = \{a^n b^m : n > m \}$ is Not Regular

Think about.
$L_7 = \{ a^n b^m : n > m \}$ is Not Regular

Think about.

Problematic: Can take $w$ long and pump $a$’s, but that won’t get out of the language.
$L_7 = \{ a^n b^m : n > m \}$ is Not Regular

Think about.

**Problematic** Can take $w$ long and pump $a$’s, but that won’t get out of the language.

**So what to do?** Revise Pumping Lemma
$L_7 = \{a^n b^m : n > m\}$ is Not Regular

Think about.

**Problematic** Can take $w$ long and pump $a$’s, but that won’t get out of the language.

**So what to do?** Revise Pumping Lemma

Pumping Lemma had a bound on $|xy|$. 
$L_7 = \{a^n b^m : n > m\}$ is Not Regular

Think about.

Problematic Can take $w$ long and pump $a$’s, but that won’t get out of the language.
So what to do? Revise Pumping Lemma
Pumping Lemma had a bound on $|xy|$.
Can also bound $|yz|$ by same proof.
$L_7 = \{a^n b^m : n > m\}$ is Not Regular

Think about.

**Problematic** Can take $w$ long and pump $a$’s, but that won’t get out of the language.

**So what to do?** Revise Pumping Lemma

Pumping Lemma had a bound on $|xy|$.

Can also bound $|yz|$ by same proof.

Do that and then you can get $y$ to be all $b$’s, pump $b$’s, and get out of the language.
$L_8 = \left\{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \right\}$ is Not Regular
\[ L_8 = \{a^{n_1}b^mc^{n_2} : n_1, n_2 > m\} \text{ is Not Regular} \]

Think about.
L_8 = \{a^{n_1}b^mc^{n_2} : n_1, n_2 > m\} is Not Regular

Think about.

Problematic Neither pumping on the left or on the right works.
$L_8 = \{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \}$ is Not Regular

Think about.

**Problematic** Neither pumping on the left or on the right works.

**So what to do?** Let’s go back to the pumping lemma with a carefully chosen string.
\( L_8 = \{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \} \) is Not Regular

Think about.

**Problematic** Neither pumping on the left or on the right works.

**So what to do?** Let’s go back to the pumping lemma with a carefully chosen string.

\( w = a^n b^{n-1} c^n \).
$L_8 = \{a^{n_1}b^mc^{n_2} : n_1, n_2 > m\}$ is Not Regular

Think about.

**Problematic** Neither pumping on the left or on the right works.

**So what to do?** Let’s go back to the pumping lemma with a carefully chosen string.

$w = a^n b^{n-1} c^n$.

$x = a^j$, $y = a^k$, $z = a^{n-j-k} b^{n-1} c^n$. 
Let's think about it.

**Problematic** Neither pumping on the left or on the right works.

**So what to do?** Let's go back to the pumping lemma with a carefully chosen string.

Let $w = a^n b^{n-1} c^n$.

Let $x = a^j$, $y = a^k$, $z = a^{n-j-k} b^{n-1} c^n$.

For all $i \geq 0$, $xy^i z \in L_8$. 

$L_8 = \{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \}$ is Not Regular
$L_8 = \{a^{n_1}b^mc^{n_2} : n_1, n_2 > m\}$ is Not Regular

Think about.

**Problematic** Neither pumping on the left or on the right works.

**So what to do?** Let's go back to the pumping lemma with a carefully chosen string.

$w = a^n b^{n-1} c^n$.

$x = a^j$, $y = a^k$, $z = a^{n-j-k} b^{n-1} c^n$.

For all $i \geq 0$, $xy^iz \in L_8$.

$$xy^iz = a^{j+ik+(n-j-k)} b^{n-1} c^n$$
$L_8 = \{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \}$ is Not Regular (Cont)

$xy^iz = a^{j+ik+(n-j-k)} b^{n-1} c^n$
$L_8 = \{a^{n_1}b^m c^{n_2} : n_1, n_2 > m\}$ is Not Regular (Cont)

$$xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n$$

For all $i$, $xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n \in L_8$. 
$L_8 = \{ a^{n_1} b^m c^{n_2} : n_1, n_2 > m \}$ is Not Regular (Cont)

\[ xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n \]

For all $i$ \[ xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n \in L_8. \]

**Key** We are used to thinking of $i$ large. But we can also take $i = 0$, cut out that part of the word. We take $i = 0$ to get

\[ xy^0 z = a^{n-k} b^{n-1} c^n \]
\( L_8 = \{ a^{n_1}b^mc^{n_2} : n_1, n_2 > m \} \) is Not Regular (Cont)

\[
xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n
\]

For all \( i \) \( xy^i z = a^{j+ik+(n-j-k)} b^{n-1} c^n \in L_8 \).

**Key** We are used to thinking of \( i \) large. But we can also take \( i = 0 \), cut out that part of the word. We take \( i = 0 \) to get

\[
xy^0 z = a^{n-k} b^{n-1} c^n
\]

Since \( k \geq 1 \), we have that \( \#_a(xy^0z) < n \leq n-1 = \#_b(xy^0z) \). Hence \( xy^0z \notin L_8 \).
$i = 0$ Case as a Picture