So far, we have defined "regularity" of languages

- accepted by DFAs

We have also developed notions equivalent to regularity:

- accepted by NFAs
- denoted by regular expressions (last lecture).

We have learned tools to show regularity in earlier classes:

- closure properties:
  - set-theoretic operations: ∪, ∩, −, −
  - language operations: ∈, ∗, ∪

Have we done problem to answer a fundamental question: show that a language \( L \) is not regular? Yes, we have — Homework 2, Problem 3:

Show that there does not exist any DFA that accepts the language non-regularity \( L = \{ a^n b^n a^{2n} \mid n \geq 1 \} \).

— Via a contradiction argument with Pigeonhole Principle.
Now, we develop below a necessary condition for regularity
- **Pumping Lemma for Regular Languages**

And, we apply the lemma to disprove regularity
(i.e., show that a language is not regular).

We'll state the Pumping Lemma in First-Order form (our usual quantified boolean statement), the framework in setting up the contradiction argument (to prove non-regularity) and then few examples.

Rationale for the Pumping Lemma will be given later.

**Pumping Lemma** ( [Sip12] Theorem 1.70 )

∀ regular languages \( L \) over alphabet \( \Sigma \), ∃ a positive integer \( n \)
\[ ∀ z \in L \left( |z| ≥ n \implies \exists u, v, w \in \Sigma^* \left( z = vuv \wedge |uv| ≤ n \wedge |v| ≥ 1 \right) \right) \]

(Recall the usual structures of \( ∀ \) and \( ∃ \) statements)

\[ ∀ \cdots ( \exists e \implies \exists e ) \]

\[ ∃ \cdots ( \exists e \wedge \exists e ) \]
Repeat the above with annotations that help the semantics of the Pumping Lemma:

\[ \forall L : \text{regular} \Rightarrow \forall z \in L : |z| \geq 1 \Rightarrow \exists u,v,w \in \Sigma^* : z = uvw \land |uv| \leq n \land |v| \geq 1 \land (\exists i \geq 1 : uv^i w \in L) \]

The following part is a necessary condition for a regular \( L \):

- Imagine an adversary giving us a Pumping Lemma constant \( n \geq 1 \) (but we do not know its value).
- For any \( z \in L \) that is lengthy (compared to the Pumping Lemma constant \( n \)), there exist \( u,v,w \) such that:
  - \( z = uvw \) (decomposed)
  - \( |uv| \leq n \) (short relative to \( n \))
  - \( |v| \geq 1 \) (not empty)
  - \( v \neq \lambda \) (any unpumped string \( i = 0 \))
  - \( uv^i w \in L \) and pumped strings \( i \geq 1 \) are also in \( L \)

\[
\begin{align*}
  i = 0 & : uv^0 w = uvw \\
  \text{(unpumped)} \\
  i = 2 & : uv^2 w = uvvwv \in L \\
  \text{(pumped)}
\end{align*}
\]
Note that the Pumplin Lemma is stated as a necessary condition on a regular language. How is it applied to show the non-regularity of a language $L$?

1. For a given language $L = \{ x \in \Sigma^* | \ldots \}$, what should be our "educated guess" on the regularity of $L$? Regular or not regular? From extensive example languages and insights learned, (also, remember FAs are \textit{finite-state} machines) finite!

2. Assume that we have educated guess that $L$ is not regular.

   We prove its non-regularity by following a contradiction argument as follows.

   Suppose that $L = \{ x \in \Sigma^* | \ldots \}$ were regular.

   Now, we follow the Pumping Lemma on $L$ (as $L$ was supposed to be regular), we would enjoy the truth of the necessary outcome:
that would be:

\[ \exists n \geq 1 \ \forall z \in L \ (|z| > n \Rightarrow \exists u, v, w \ (z = u \cdot w \wedge |u| \leq n \wedge \forall i \geq 0 \ u \cdot i \cdot w \in L)) \]

So, as we would enjoy/follow the necessary outcome,
we let \( n \geq 1 \) be the Pumping Lemma constant
(remember, it is "existential" - given by adversary)
and we do NOT know its precise value.

Then, we want to show that:

\[ \forall z \in L \ (|z| > n \Rightarrow \exists u, v, w \ (z = u \cdot w \wedge |u| \leq n \wedge \forall i \geq 0 \ u \cdot i \cdot w \in L)) \]

is NOT TRUE (remember we are in the process of)
developing a contradiction.

That is, at this point, we want to argue/show that

\[ \neg (\forall z \in L \ (|z| > n \Rightarrow \exists u, v, w \ (z = u \cdot w \wedge |u| \leq n \wedge \forall i \geq 0 \ u \cdot i \cdot w \in L))) \]

is TRUE!

Now, can you rewrite statement 1?
Recall what we learned in Weeks 1/2 on
Common Logical Equivalences for Propositional Logic
and First-Order Logic:

\[-(\forall \ldots) \equiv \exists(-\ldots)\]

\[-(\exists \ldots) \equiv \forall(-\ldots)\]

\[p \rightarrow q \equiv \neg p \lor q\]

\[-(p \lor q) \equiv \neg p \land \neg q \equiv p \rightarrow \neg q\]

The above statement \((1)\) is logically equivalent to:

\[\exists z \in L \left( |z| \geq n \land \forall u, v, w \left( \begin{array}{c}
\exists z = uvw \\
\overset{2}{1} u, v, w \\
\overset{2}{1} u, v, w
\end{array} \right) \Rightarrow \exists i \geq 0 \quad u^i v^i w^i \notin L \right)\]

Want to argue/show that this is TRUE!

3. Now, continue to argue that statement \((2)\) is true.

So, we need to find (existential!) one \(z \in L\) that is long (i.e., \(|z| \geq n\))

Note: Do not commit the "ROOKIE" mistake
As the Pumping Lemma constant \(n\) is given by an adversary, its value is unknown - existential,
so, when we try to find \(z \in L\), any such candidate \(z \in L\) must be long,
like \(z = aabb - as (|z| = 2 \geq n)\)
So, we try to find \( z = \ldots \in L \)
with \( |z| = \mathfrak{m} \geq n \).

Then, we need to show that
\[ \forall u, v, w \left( \begin{array}{c}
z = uvw \\
uv \leq \eta \\
v \geq 1 \\
\implies \exists i \geq 0 \quad uv^i w \notin L \end{array} \right) \]

That is, we need to consider ALL possible \( u, v, w \)
such that \[ z = uvw \]
\( \wedge \) \( |uvw| \leq \eta \)
\( \wedge \) \( |v| \geq 1 \)

\( u, w \) is short
\( v \) is not empty

and for each such combination \( u, v, w \),
we find (existential) \( i \geq 0 \)
\( (i \) may be 0, may be 2, 3, \ldots \)
\( uv^i w \notin L \)
(\( u, w \) are \( L \), \( v \) is short, "\( i = 1 \)" will not work, since \( uv^1 w = z \in L \))

ALL \( u, v, w \), \( z = uvw \wedge (uvw \leq \eta \wedge |v| \geq 1 \)

\( u = a, v = b, w = \ldots \)
find \( i \geq 0 \)
\( (\) may be \( i = 0 \)\)
\( uv^i w = uvw \notin L \)

\( u = a, v = bb, w = \ldots \)
find \( i \geq 0 \)
\( (\) may be \( i = 3 \)\)
\( uv^3 w = uvvw \notin L \)
Really understand the "logical game" described above. Our textbook applies the same logics in setting up a contradiction argument. The details are given above to illustrate the important understanding of the logics behind.

Now, we consider a few examples.

Example 1. \( L = \{ a^i b^i \mid i \geq 0 \} \)

\[ \{ \varepsilon, ab, aabb, aababb, \ldots \} \]

Our educated guess is that \( L \) is not regular.

(any FA seems to require to remember the number of As — can not be achieved by "finite-state"

— this is not a proof, simply an educated guess.

We prove that \( L \) is not regular by Pumping Lemma (for regularity).

Notice the following "framework" for setting up a contradiction argument — detailed in pages 4, 5, 6, and 7 this notes.
Suppose that \( L = \{ a^i b^i | i \geq 0 \} \) were regular.

Let \( n \geq 1 \) be the Pumping Lemma constant.

We consider \( z = a^n b^n \in L \) with \( |z| = n + n \geq n \).

Exercise: You may want to try another candidate \( z = a^{\frac{n}{2}} b^{\frac{n}{2}} \in L \) with \( |z| = \frac{n}{2} + \frac{n}{2} \geq n \).

No! As we do not know the value of \( n \geq 1 \), \( \frac{n}{2} \) is not necessarily integral.

How about another candidate \( z = a^{\left\lfloor \frac{n}{2} \right\rfloor} b^{\left\lfloor \frac{n}{2} \right\rfloor} \in L \) with \( |z| = \left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{n}{2} \right\rfloor = 2 \left\lfloor \frac{n}{2} \right\rfloor \geq n \)?

Try this candidate \( z \) as exercise.

Now, back to our candidate \( z = a^n b^n \in L \) with \( |z| = n + n \geq n \).

Then, we need to consider:

\( \forall u, v, w \) such that \( |uv| \leq n \) and \( |v| \geq 1 \), and for each combination of \( u, v, w \), find \( i \geq 0 \) such that \( uv^i w \notin L \).

\( u^i v^i w \notin L \)
A last-resort method to consider  
for $u, v, w$ such that  
$z = u v w$  
and ($u v \leq n$)  
and ($v \geq 1$)  
consider all possible "sliding $u, v, w$" decomposing $w$  
such that ($u v \leq n$)  
and ($v \geq 1$)  

draw $u v w$  

$z = u v w$  

"slide $u, v, w$  
from left to right  
to generate all cases"

All these possibilities can be organized into following cases  
(by paying attention to the "boundary" between  
$a^n$ and $b^n$ that is, minding the symbols composing $u$ and $v$)

Case 1:

$u = a^{\nu}$  
$w = a^{n-\nu-1} b^n$

$u v w$:

$u = a^{\nu}$  
$v = a^{\nu v}$  
$w = a^{n-\nu v-1} b^n$

and ($v \geq 1$)

—for $\alpha, \beta \leq n$  
$u = a^\alpha$  
$v = a^\beta$  
$w = a^{n-\alpha-\beta} b^n$

Now, we try $i = 0$:  
$u v^i w = u w$  
(unpump $v$ to  
discard at least  
one $a$)

Since $n-\nu v \neq n$  
and ($v \geq 1$)
Case 2

\[ \begin{array}{c}
\text{slide } u, v, w \\
\text{from left to right to generate all cases}
\end{array} \]

So \( u \) is within \( a^n \) and \( v \) crosses the "mid-line".

\[ u = a^{n-1} u' \quad \text{and} \quad v = a^n b^\beta \]

As we are considering "\( v \) crosses ..."

so \( n-1u' \geq 1 \)

and \( \beta \geq 1 \)

This consideration is NOT possible since we must need \( |uv| \leq n \)

(here, \( |uv| = n + \beta > n \))

as \( \beta \geq 1 \)

and \( |v| \geq 1 \)

(satisfied)

Remaining Case 3

\[ \begin{array}{c}
\text{NOT possible} \\
|uv| = n
\end{array} \]

So, combining Cases 1, 2, or 3

(only Case 1 is possible)

for Case 1, \( \exists i (\neq 0) \quad u v^i w \notin L \)

Such contradiction shows that \( L \) is not regular.
How about we use the candidate string
\[ z = \alpha a b \in L \text{ with } 1 \leq |\alpha| + |\beta| \leq n. \]

Proceed as above, we consider

\[ u, v, w \text{ such that } z = uww \text{ and } |uv| \leq n \text{ and } |v| \geq 1 \]

and for each such combination \( u, v, w \), find \( i \geq 0 \)

with \( uv^i w \notin L \).

Consider the following cases for all such \( u, v, w \):

**Case 1:**

\[
\begin{array}{c}
\begin{array}{c}
\frac{\alpha}{a} \\
\frac{\beta}{b}
\end{array}
\end{array}
\]

\[ u = a^{1\frac{m}{2}} \]
\[ v = a^{1\frac{m}{2}} \]
\[ w = a^{b\frac{m}{2}} \]

Then, check that, for \( i = 0 \) (unpumped)

\[ u^0 w = u w = a^{1\frac{m}{2}} a^{b\frac{m}{2}} = a^{1\frac{m}{2} + b\frac{m}{2}} \notin L \]

as \( |v| \geq 1 \)

**Case 2:**

\[
\begin{array}{c}
\begin{array}{c}
\frac{\alpha}{a} \\
\frac{\beta}{b}
\end{array}
\end{array}
\]

\[ u = a^{1\frac{m}{2}} \]
\[ v = a^{b\frac{m}{2}} \]
\[ w = a^{b\frac{m}{2}} \]

with \( |uv| + \frac{n}{2} - |uv| + \beta \leq n \) (possible here)

\[ \frac{n}{2} - |uv| + \beta \geq 1 \]

\( v \) crosses midline boundary
Be careful for this case!

\[
\begin{array}{c|c|c}
\frac{a}{b} & \frac{a}{b} \\
\hline
u & v
\end{array}
\]

If we try \( i = 0 \) (unpumping to discard symbols of \( v \))

\[uv^0w = uw = a^b \] MAY still be in \( L \)

(\text{so, no contradiction!})

Think of the scenario that \( v \) may be

\[v = a^3b^3\]

so \( uv^0w = uw \)

but the resulting string is still in \( L \)!

But, we are "fortunate".

We try \( i = 2 \):

\[u v^2w = u v v w = a^1 a^b a^b b b\]

remember \( \beta > 1 \)

so we have \( b's \) preceding \( a's \)

and this string is \( \not\in L \)

as desired.

\[L \supseteq \{a^n b^n \mid n \geq 1\}\]
Case 3

\[
\begin{array}{c|c|c}
\frac{a}{u} & \frac{b}{v} \\
\hline
a & b
\end{array}
\]

\[
\begin{align*}
u &= a\frac{a^7}{b}^x \quad |v| + a + 1 \nu_1 &\leq n \quad \text{(possible)} \\
v &= b^1\nu_1 \\
w &= b^{a_7-a-1}\nu_1
\end{align*}
\]

What is \( i \) for this case?

Uring \( \text{pumping} \) to discard \( v \) (and its symbol = \( b^x \))

\[
i = 0: \quad u^i v^0 w = u^i w = a b^6 b^{a_7 - a - 1}\nu_1 = a b^6 b^{a_7 - a - 1}\nu_1 \notin L \quad \text{as } |v| \geq 1.
\]

Combining Cases 1, 2, and 3, for each case,

\[
\exists i > 0, \quad u v^i w \notin L.
\]

More examples next class.