Pumping Lemma
CSCI 338
If all regular languages have property $P$, and some new language $L$ does not have property $P$, then...?
If all regular languages have property \( P \), and some new language \( L \) does not have property \( P \), then \( L \) cannot be a regular language.
Properties That Imply Language is Regular?

**Claim**: Languages that are finite in size are regular.

**Claim**: Languages where all strings have bounded size (each string has size $\leq$ some $n$) are regular.
Properties That Imply Language is Regular?

Claim: Languages that are finite in size are regular.

Proof: Since there are a finite number of strings, build a DFA for each individual string in the language. Then connect all start states to a new start state via $\varepsilon$-transitions.

Claim: Languages where all strings have bounded size (each string has size $\leq$ some $n$) are regular.
Properties That Imply Language is Regular?

Claim: Languages that are finite in size are regular.

Proof: Since there are a finite number of strings, build a DFA for each individual string in the language. Then connect all start states to a new start state via $\varepsilon$-transitions.

Claim: Languages where all strings have bounded size (each string has size $\leq$ some $n$) are regular.

Proof: Since the alphabet is finite, there is a finite number of strings constructible with $n$ characters. Thus, the language is finite and regular.
What do we know about non-regular languages?

- They must be infinite in size.
- They must have arbitrarily long strings in them.
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

?
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p =$ number of states.
Let $s$ be any string in $L$ such that $|s| \geq p$.
Then...
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p = \text{number of states}$.
Let $s$ be any string in $L$ such that $|s| \geq p$.
Then, $s$ must visit repeated states (i.e. loops).
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p = $ number of states.
Let $s$ be any string in $L$ such that $|s| \geq p$.
Then, $s$ must visit repeated states (i.e. loops).

\[ e.g. \ s = 00111 \in L \]
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p =$ number of states.

Let $s$ be any string in $L$ such that $|s| \geq p$.

Then, $s$ must visit repeated states (i.e. loops).

e.g. $s = 0|011|1 \in L$
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p =$ number of states.
Let $s$ be any string in $L$ such that $|s| \geq p$.
Then, $s$ must visit repeated states (i.e. loops).

e.g. $s = 0|011|1 \in L$
Is $s = 0|011|011|1 \in L$?
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p = \text{number of states}$.
Let $s$ be any string in $L$ such that $|s| \geq p$.
Then, $s$ must visit repeated states (i.e. loops).

E.g. $s = 0|011|1 \in L$  
Is $s = 0|011|011|1 \in L$?  
What about $s = 0|011|011|011|1 \in L$?
Suppose some regular language \( L \) contains strings of arbitrary length (i.e. \( \forall n \geq 0, \exists s \in L \text{ such that } |s| \geq n \)). Then:

1. We know the DFA for that language has a finite number of states.
2. \( \exists \) strings longer than the number of states.

Let \( p = \text{number of states} \).

Let \( s \) be any string in \( L \) such that \( |s| \geq p \).

Then, \( s \) must visit repeated states (i.e. loops).

\[ e.g. \ s = 0|011|1 \in L \]
\[ \text{Is } s = 0|011|011|011|1 \in L? \]

\[ \text{What about } s = 0|11|011|011|1 \in L? \]
\[ \text{What about } s = 0|1 \in L? \]
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p = \text{number of states.}$

Let $s$ be any string in $L$ such that $|s| \geq p$. Then, $s$ must visit repeated states (i.e. loops).

Let $s = 0|011|1 \in L$

Is $s = 0|011|011|011|1 \in L$?

What about $s = 0|1|1 \in L$?
Quest for Regular Language Properties

Suppose some regular language \( L \) contains strings of arbitrary length (i.e. \( \forall n \geq 0, \exists s \in L \) such that \( |s| \geq n \)). Then:

1. We know the DFA for that language has a finite number of states.
2. \( \exists \) strings longer than the number of states.

Let \( p = \text{number of states} \).

Let \( s \) be any string in \( L \) such that \( |s| \geq p \).

Then, \( s \) must visit repeated states (i.e. loops).

\[ x \quad y \quad z \]

e.g. \( s = 0|011|1 \in L \)

What about \( s = 0|011|011|011|1 \in L \)?
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Let $p =$ number of states. Let $s$ be any string in $L$ such that $|s| \geq p$. Then, $s$ must visit repeated states (i.e. loops).

\[
\text{e.g. } s = 0|011|1 \in L
\]

What about $s = 0|1 \in L$?
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Summary: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

**Summary:** Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xy^iz \in L, \forall i \geq 0$.

From our previous argument.
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

**Summary:** Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xy^iz \in L, \forall i \geq 0$.
2. $|y| > 0$.

Since $|s| \geq p$, we must have repeated states.
Quest for Regular Language Properties

Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Summary: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

Since there have to be repeated states within the first $p$ transitions.
Suppose some regular language $L$ contains strings of arbitrary length (i.e. $\forall n \geq 0, \exists s \in L$ such that $|s| \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

**Pumping Lemma**

*Summary:* Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xy^iz \in L, \forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$. 
The Pumping Lemma is our property that all regular languages must have.

Summary: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

Pumping Lemma
Suppose some regular language $\mathcal{L}$ contains strings of arbitrary length (i.e. $\forall n \geq 0$, $\exists s \in \mathcal{L}$ such that $s \geq n$). Then:

1. We know the DFA for that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

Summary: Given a regular language $\mathcal{L}$, $\exists$ a number $p$ such that any string $s \in \mathcal{L}$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xy^iz \in \mathcal{L}, \forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$. 

The Pumping Lemma is our property that all regular languages must have. So, if some language does not have that property, it cannot be a regular language.
Pumping Lemma

**Pumping Lemma**: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xy^iz \in L$, $\forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$. 
Pumping Lemma

Pumping Lemma: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xyz^i z \in L, \forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$.

Pumping Lemma User Manual:

1. The pumping lemma says all regular languages have property $P$.
2. If we can show a language does not have property $P$, then it cannot be regular.
Non-Regularity Proofs

Pumping Lemma: Given a regular language $L$, there exists a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

1. Suppose language is regular.
Pumping Lemma: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $x y^i z \in L$, $\forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$.

Non-Regularity Proofs

1. Suppose language is regular.
2. Select $p$ from pumping lemma.
Non-Regularity Proofs

**Pumping Lemma**: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

1. Suppose language is regular.
2. Select $p$ from pumping lemma.
3. Carefully select $s \in L$ and $|s| \geq p$. 
Non-Regularity Proofs

**Pumping Lemma**: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

1. Suppose language is regular.
2. Select $p$ from pumping lemma.
3. Carefully select $s \in L$ and $|s| \geq p$.
4. Determine what $y$ must consist of.
Non-Regularity Proofs

Pumping Lemma: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

1. Suppose language is regular.
2. Select $p$ from pumping lemma.
3. Carefully select $s \in L$ and $|s| \geq p$.
4. Determine what $y$ must consist of.
5. Make new string by selecting $i$. 
Non-Regularity Proofs

Pumping Lemma: Given a regular language $L$, $\exists$ a number $p$ such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

1. $xyz^i z \in L$, $\forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$.

1. Suppose language is regular.
2. Select $p$ from pumping lemma.
3. Carefully select $s \in L$ and $|s| \geq p$.
4. Determine what $y$ must consist of.
5. Make new string by selecting $i$.
6. Show new string is not in language.