Pumping Lemma
CSCI 338

Regular Language

A language is called a regular language if some DFA, NFA, or regular expression recognizes it.

How do you prove a language is regular?

Regular Language

A language is called a <u>regular language</u> if some DFA, NFA, or regular expression recognizes it.

How do you prove a language is regular?

Make a DFA, NFA, or regular expression that recognizes it.

DFA/NFA Limitations

What are some properties that regular languages must have?

DFA/NFA Limitations

What are some properties that regular languages must have?

If all regular languages have property $\boldsymbol{P}$, and some new language $\boldsymbol{L}$ does not have property $\boldsymbol{P}$, then…?

DFA/NFA Limitations

What are some properties that regular languages must have?

If all regular languages have property $\boldsymbol{P}$, and some new language $\boldsymbol{L}$ does not have property $\boldsymbol{P}$, then $\boldsymbol{L}$ cannot be a regular language.

DFA/NFA Limitations

What are some properties that regular languages must have?

If all regular languages have property $\boldsymbol{P}$, and some new language $\boldsymbol{L}$ does not have property $\boldsymbol{P}$, then $\boldsymbol{L}$ cannot be a regular language.

What are some properties that require a language to be regular?

Quest for Regular Language Properties

Claim: All languages that are finite in size are regular.

Proof: ?

Quest for Regular Language Properties

Claim: All 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.

Quest for Regular Language Properties

Claim: All 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.

What about strings that are infinitely long?

Quest for Regular Language Properties

Claim: All 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.

What about strings that are infinitely long?

DFAs/NFAs can process strings of arbitrary length, but not infinite length.

E.g. $L = \{\omega\colon \omega \text{ contains an even number of 0s}\}$

Quest for Regular Language Properties

Claim: All 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.

Connecting all start states to a new start state via ε-transitions gives an NFA that will recognize the (regular) language.

What about strings that are infinitely long?

DFAs/NFAs can process strings of arbitrary length, but not infinite length.

E.g. $L = \{\omega : \omega \text{ contains an even number of } 0\text{s}\}$

Quest for Regular Language Properties

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

Proof: ?

Quest for Regular Language Properties

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

Proof: Consider a language where each string has size $\leq n$.

Quest for Regular Language Properties

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

Proof: Consider a language where each string has size $\leq n$.

Since the alphabet is finite, there is a finite number of strings constructible with n characters.

Quest for Regular Language Properties

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

Proof: Consider a language where each string has size $\leq n$.

Since the alphabet is finite, there is a finite number of strings constructible with n characters.

Thus, the language is finite and regular.

Non-Regular Languages

What do we know about non-regular languages?

Non-Regular Languages

What do we know about non-regular languages?

- They must be infinite in length
- They must have arbitrarily long strings in them.

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:

?

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. The DFA/NFA representing that language has a finite 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. The DFA/NFA representing 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. The DFA/NFA representing 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 <u>any</u> 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. The DFA/NFA representing that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

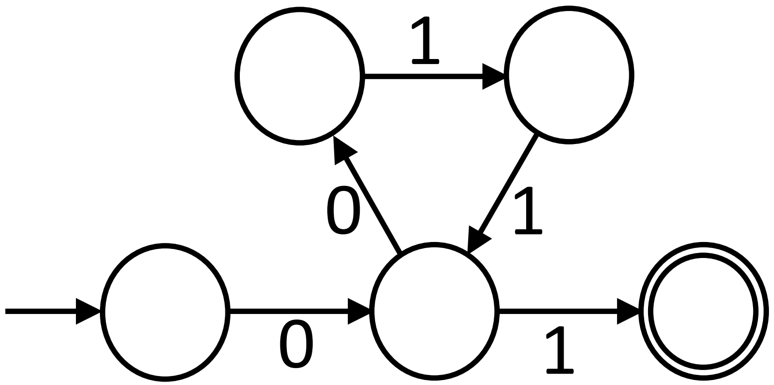
Let $p =$ number of states.
Let s be <u>any</u> string in L such that $|s| \geq p$.
Then, s must visit repeated states (i.e. loops).

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.
Then, s must visit repeated states (i.e. loops).



e.g. $s = 00111 \in L$

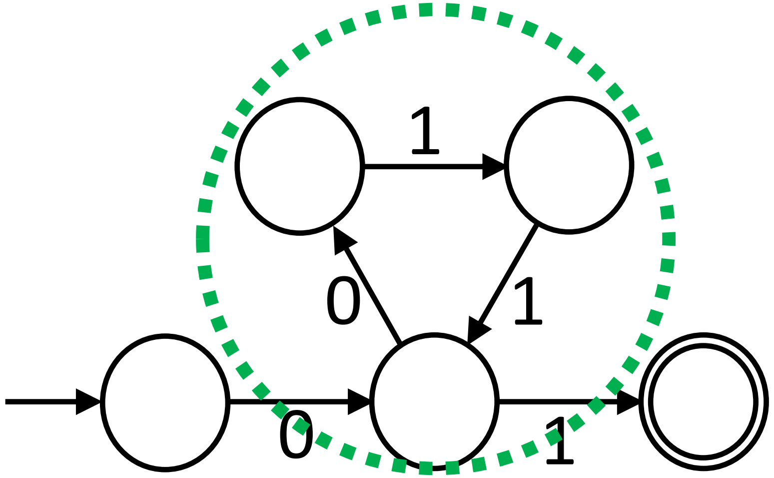
Quest for Regular Language Properties

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.
Then, s must visit repeated states (i.e. loops).

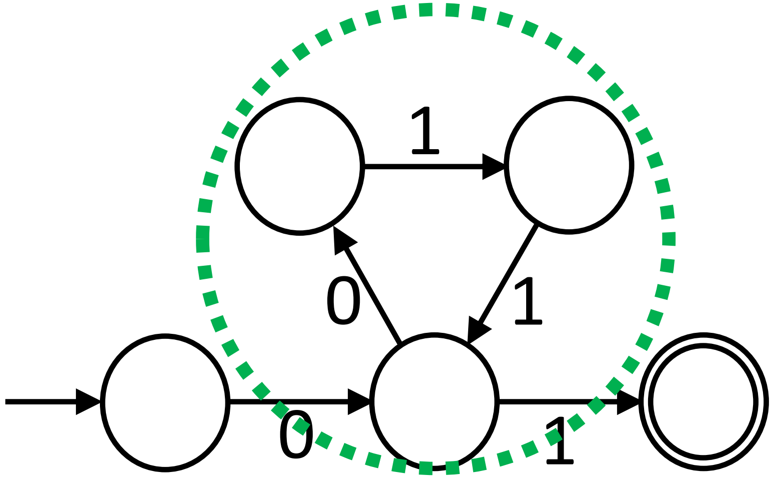
e.g. $s = 0|011|1 \in L$



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. The DFA/NFA representing that language has a finite number of states.
2. ∃ strings longer than the number of states.



Let $p =$ number of states.

Let s be <u>any</u> string in L such that $|s| \geq p$.

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

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

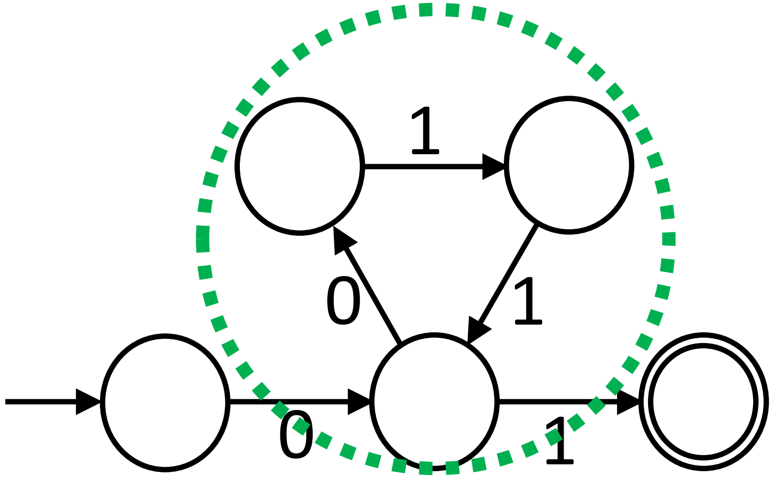
Is $s = 0|011|011|1 \in L$?

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.
Then, s must visit repeated states (i.e. loops).



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

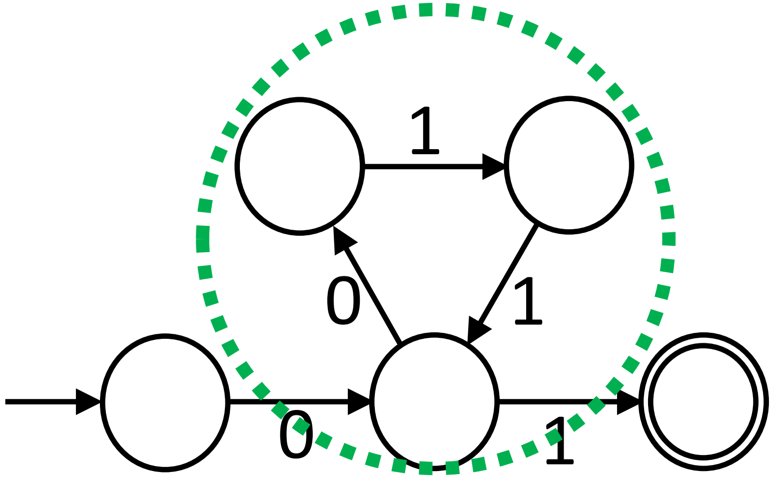
Is $s = 0|011|011|1 \in L$?

What about $s = 0|011|011|011|1 \in L$?

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. The DFA/NFA representing 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 <u>any</u> 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 $s = 0|011|011|1 \in L$?

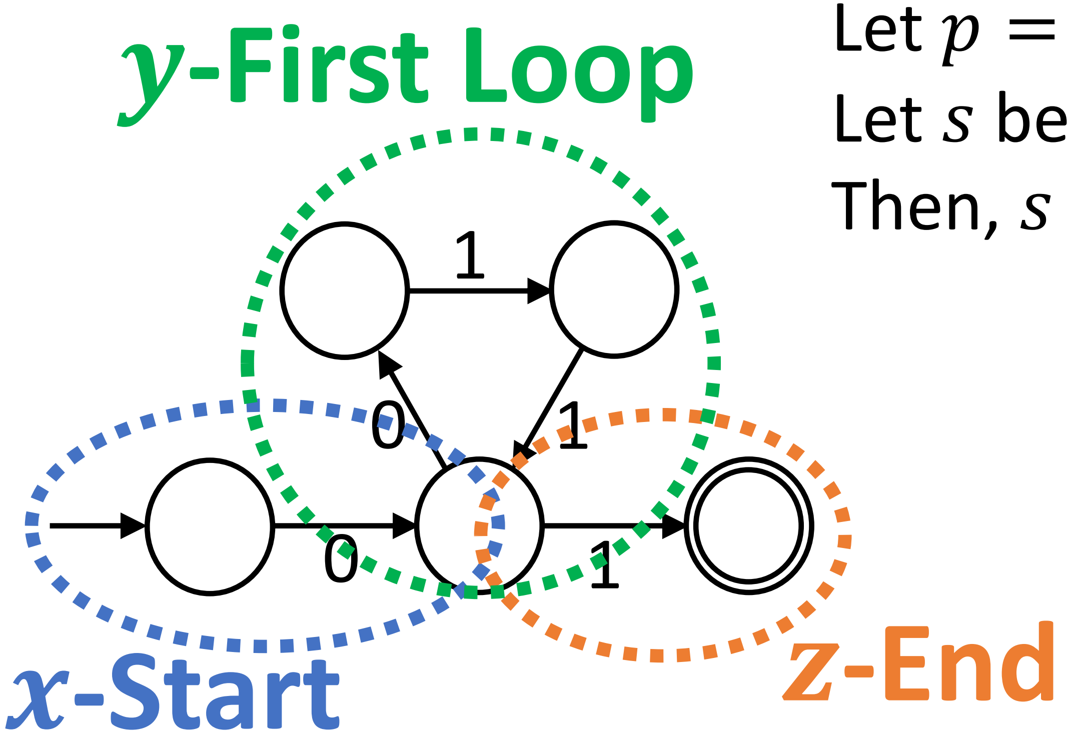
What about $s = 0|011|011|011|1 \in L$?

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.
Then, s must visit repeated states (i.e. loops).

$$\text{e.g. } s = \overset{x}{0}|\overset{y}{011}|\overset{z}{1} \in L$$

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

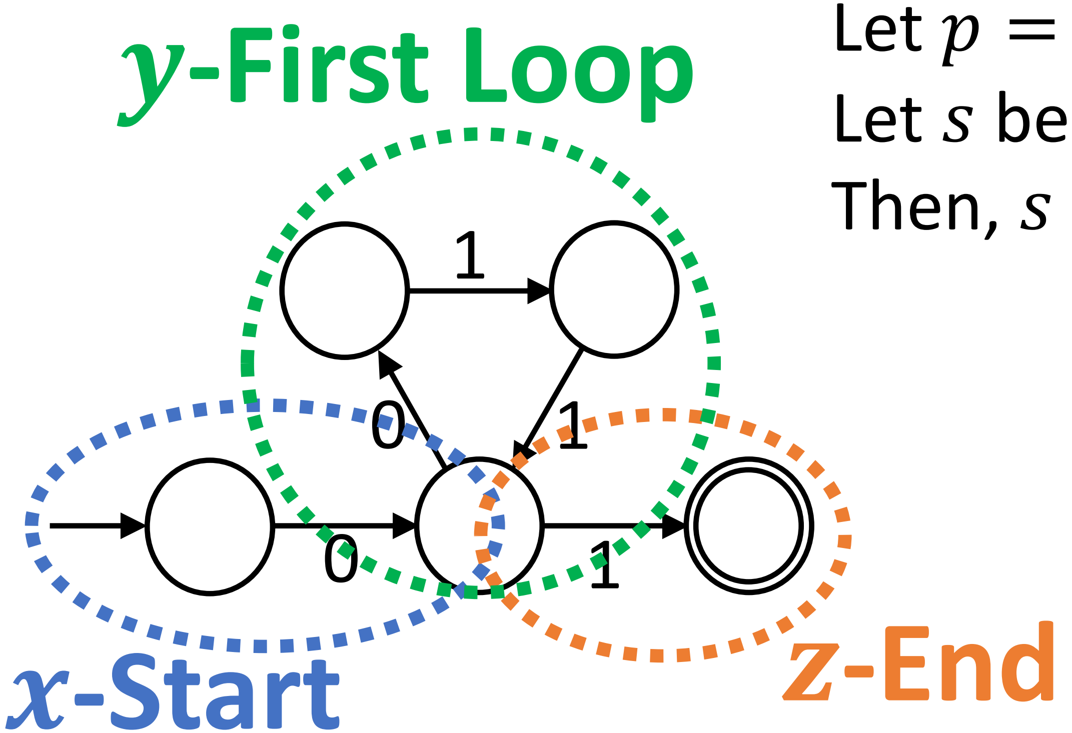
What about $s = 0|011|011|011|1 \in L$?

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.

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

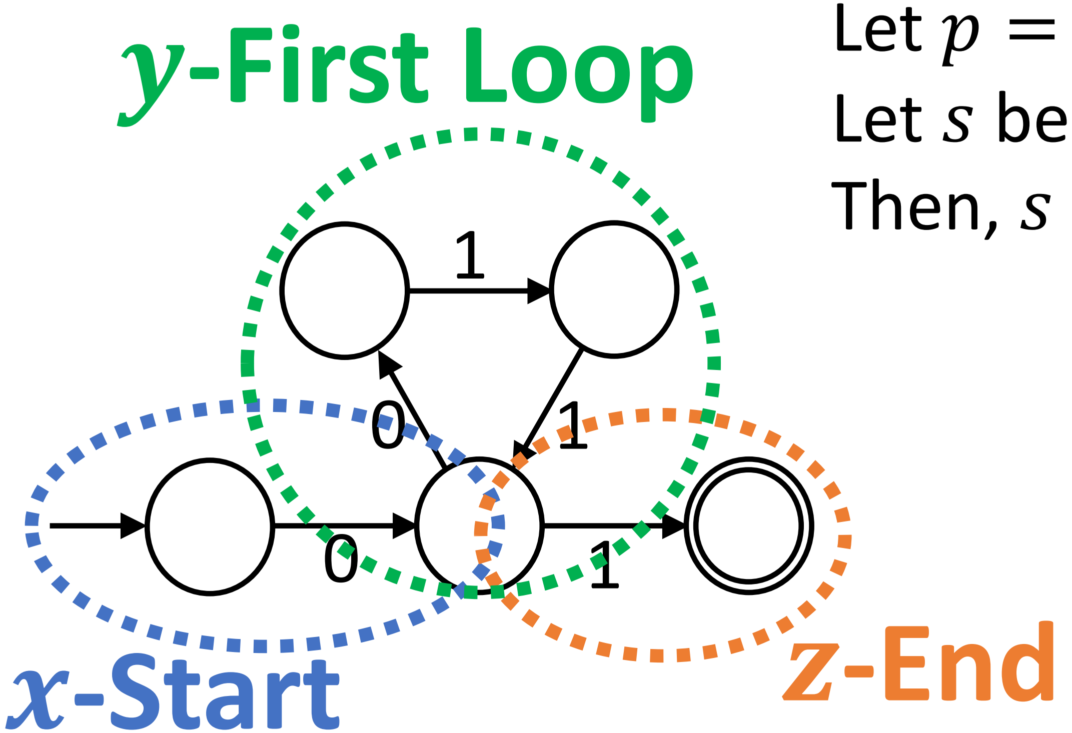
e.g. $s = \overset{x}{0}|\overset{y}{011}|\overset{z}{1} \in L$

What about $s = \overset{x}{0}|\overset{y}{011}|\overset{y}{011}|\overset{y}{011}|\overset{z}{1} \in L$?

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. The DFA/NFA representing 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 <u>any</u> string in L such that $|s| \geq p$.

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

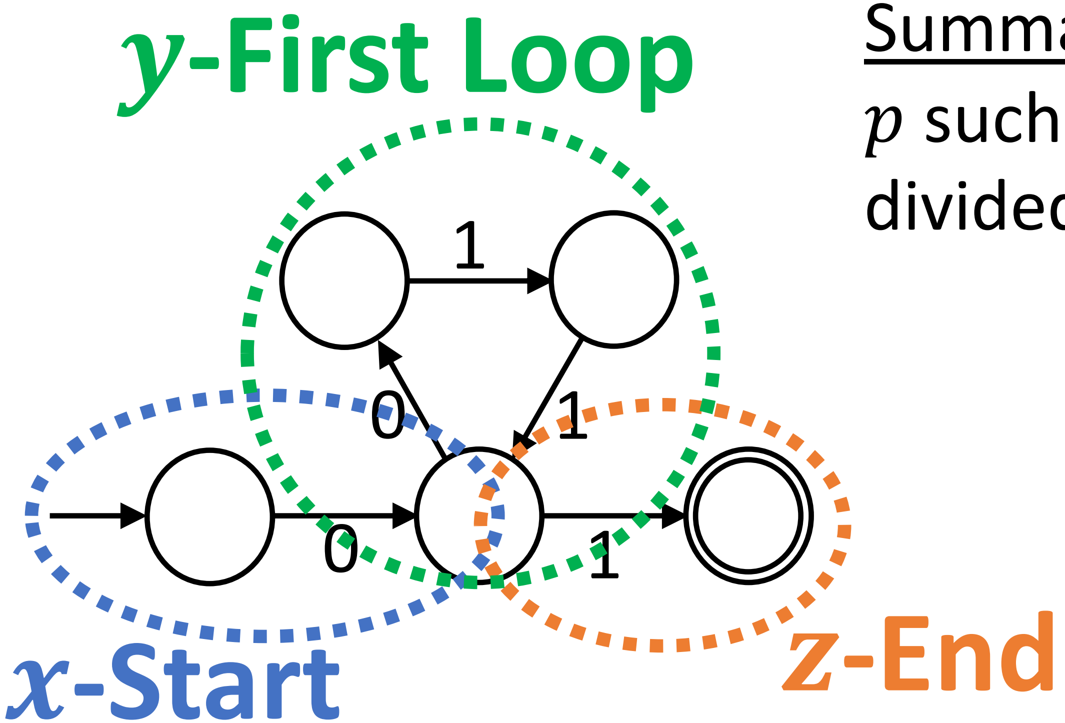
e.g. $s = \overset{x}{0}|\overset{y}{011}|\overset{z}{1} \in L$

What about $s = \overset{x}{0}|\overset{z}{1} \in L$?

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. The DFA/NFA representing that language has a finite number of states.
2. $\exists$ strings longer than the number of states.

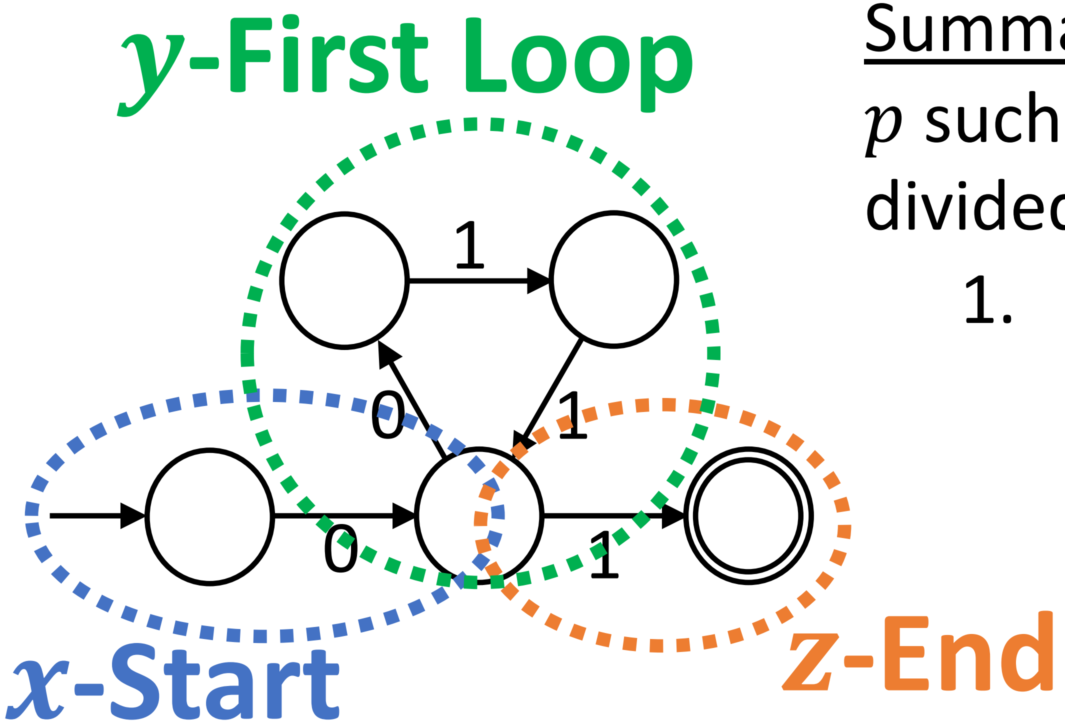


Summary: Given a regular language L, $\exists$ a number p such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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. The DFA/NFA representing 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:

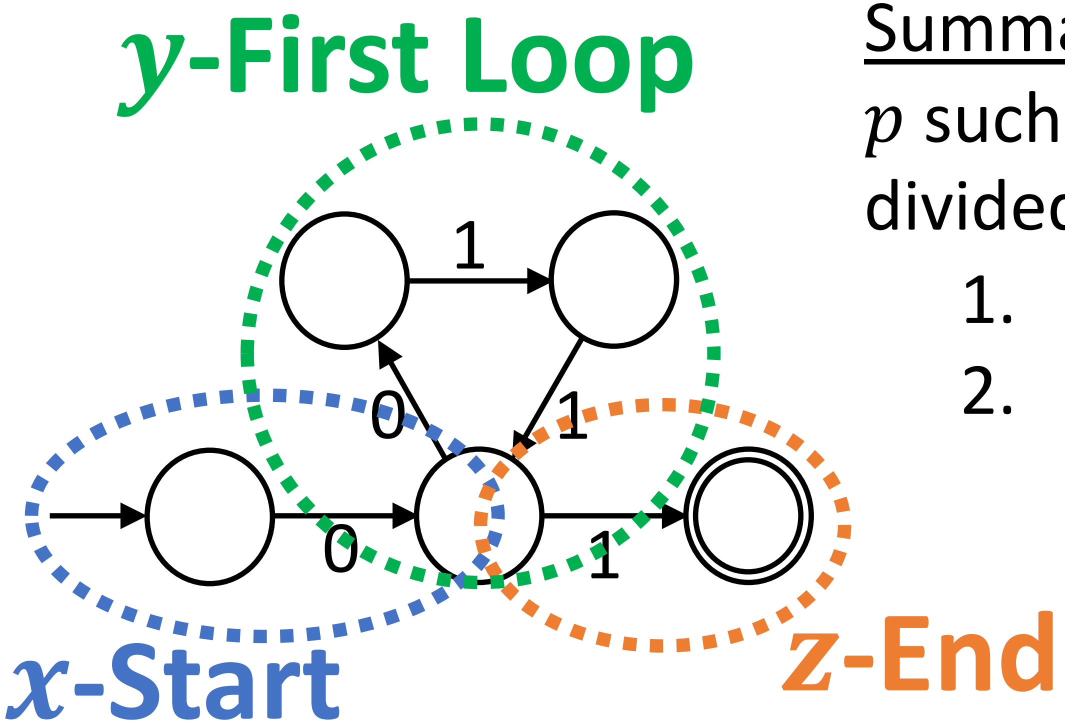
1. $xy^iz \in L, \forall i \geq 0.$

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. The DFA/NFA representing 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:

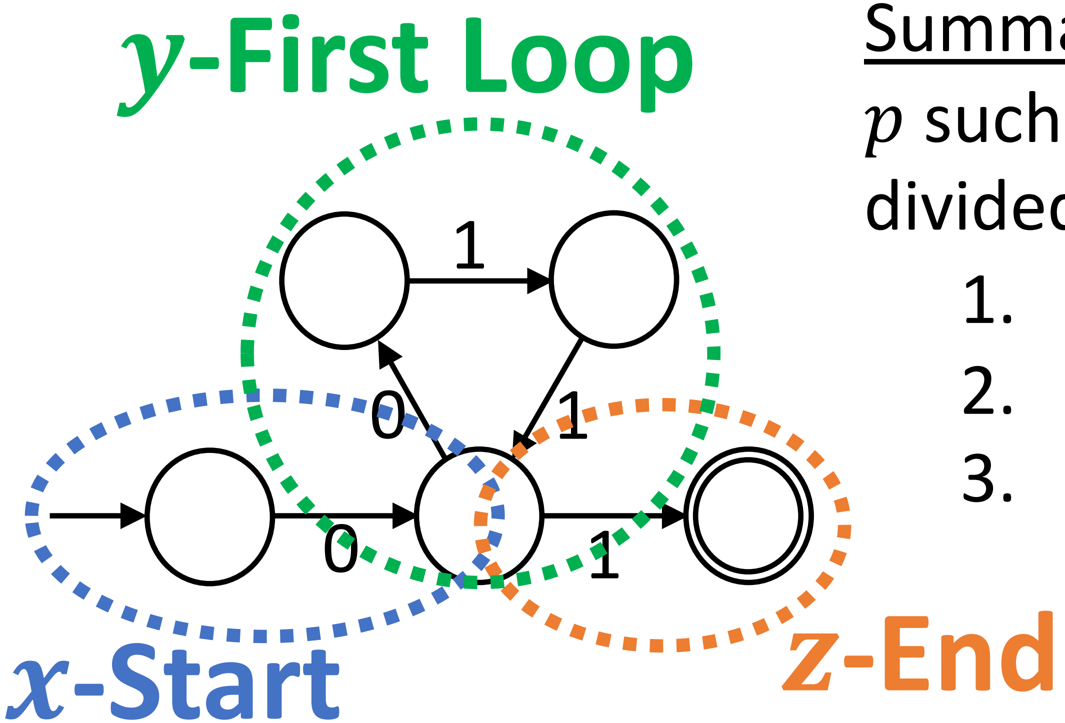
1. $xy^iz \in L, \forall i \geq 0.$
2. $|y| > 0.$

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. The DFA/NFA representing 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. $|xy| \leq p$.

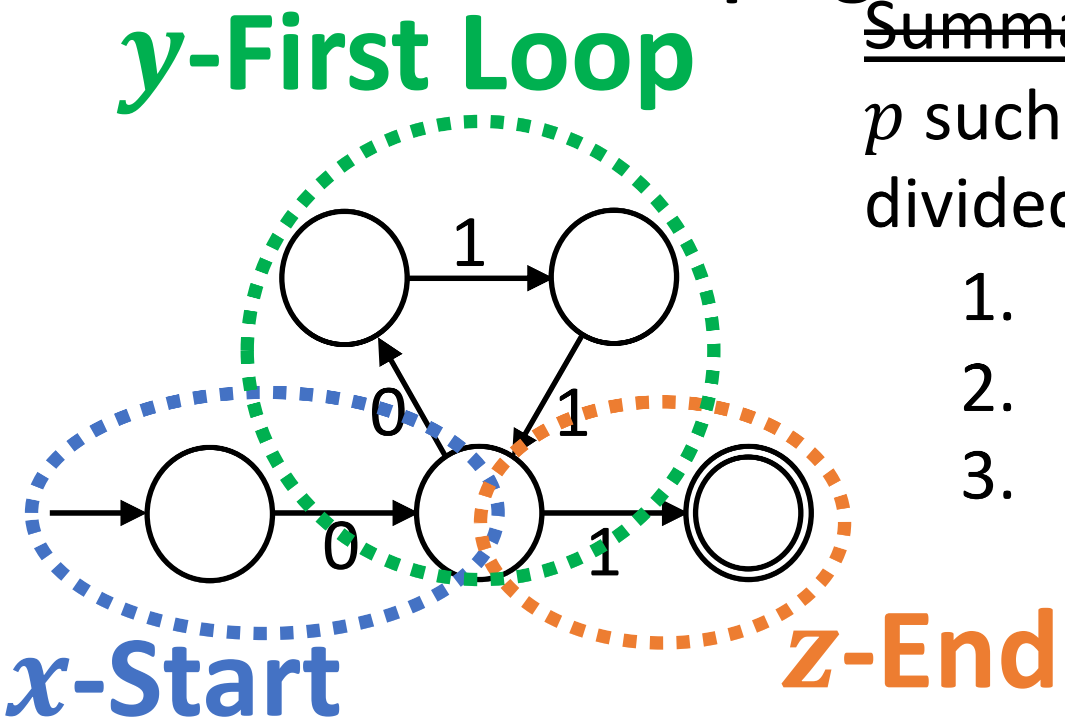
Since there have to be repeated states within the first p transitions.

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. The DFA/NFA representing 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. $xy^i z \in L, \forall i \geq 0.$
2. $|y| > 0.$
3. $|xy| \leq p.$

Quest for Regular Language Properties

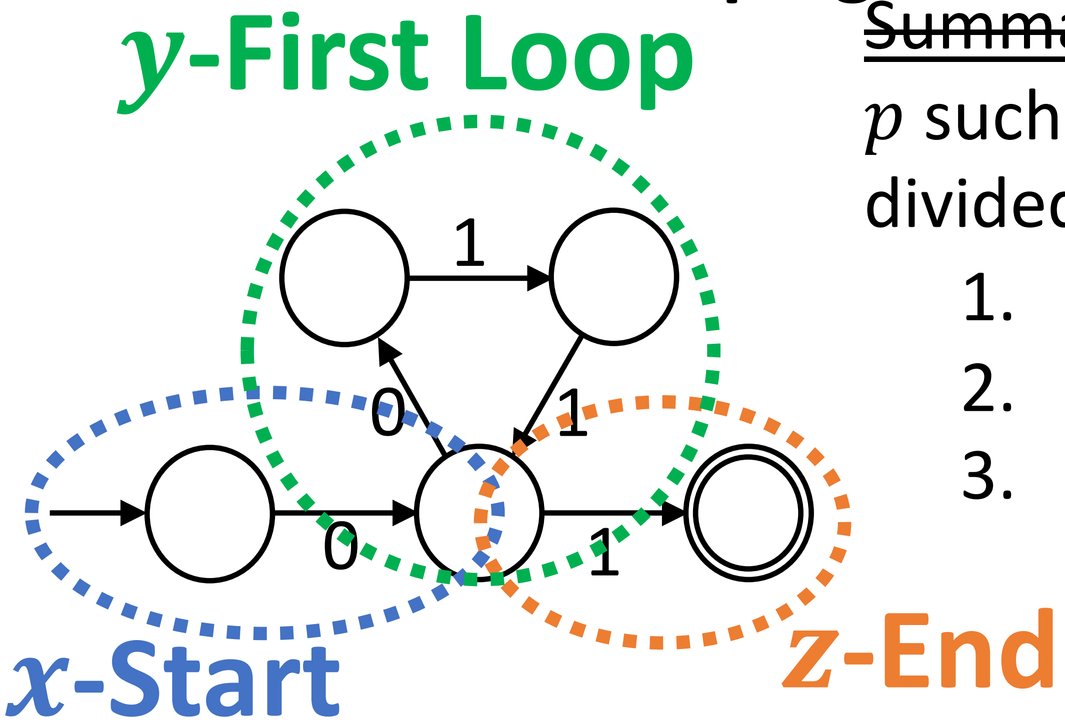
Suppos … gth

(i.e. $\forall n$ …

1. Th … states.
2. $\exists$ strings longer than the number of states.

The Pumping Lemma is our property that all regular languages must have.

Pumping Lemma



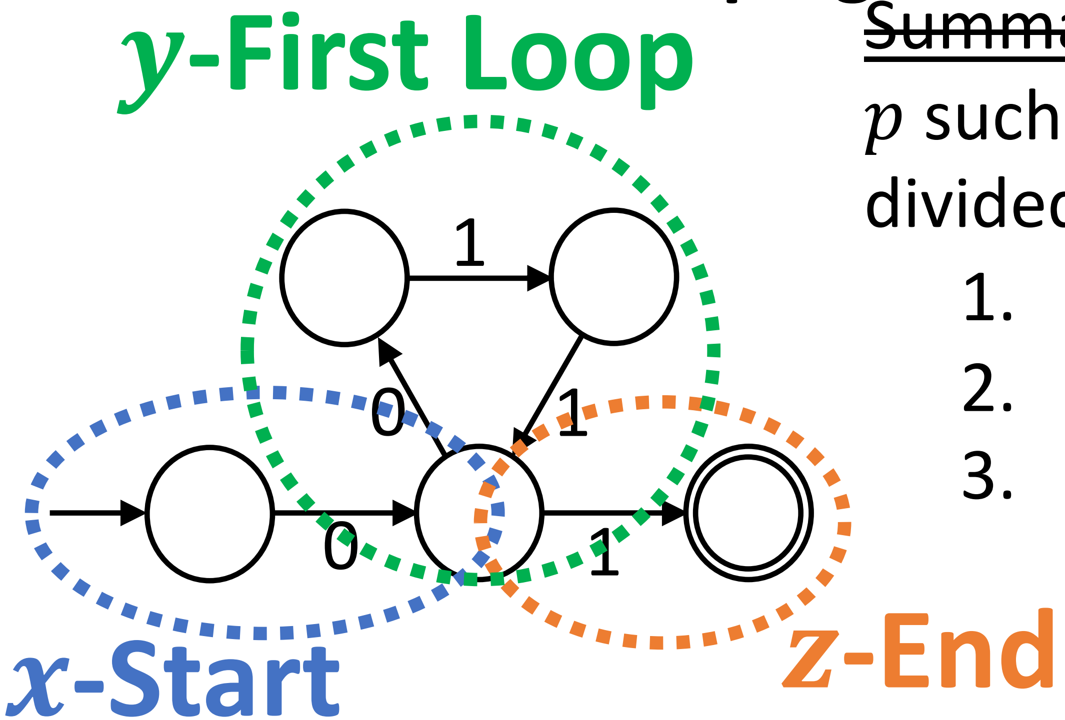
~~Summary~~: Given a regular language L, $\exists$ a number p such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

Quest for Regular Language Properties

Suppos **The Pumping Lemma is our property that all regular** gth
(i.e. $\forall n$ **languages must have. So, if some language does not**
1. Th **have that property, it cannot be a regular language.** states.
2. $\exists$ strings longer than the number of states.

Pumping Lemma



~~Summary~~: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $xy^i z \in L, \forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$.

Pumping Lemma

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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 User Manual:

1. The pumping lemma says all regular languages have property $\boldsymbol{P}$.
2. If we can show a language does not have property $\boldsymbol{P}$, then it cannot be regular.

Example with Regular Language

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

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

Example with Regular Language

Pumping Lemma: **Given a regular language $\boldsymbol{L}$**, $\exists$ 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. $|xy| \leq p.$

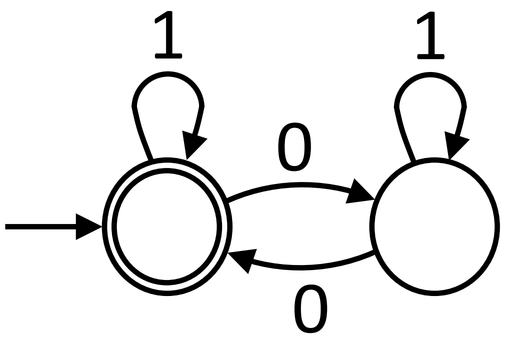
$\{\omega : \omega$ contains an even number of 0's$\}$

Example with Regular Language

<u>Pumping Lemma</u>: **Given a regular language L**, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p.$

$\{\omega : \omega$ contains an even number of 0's$\}$



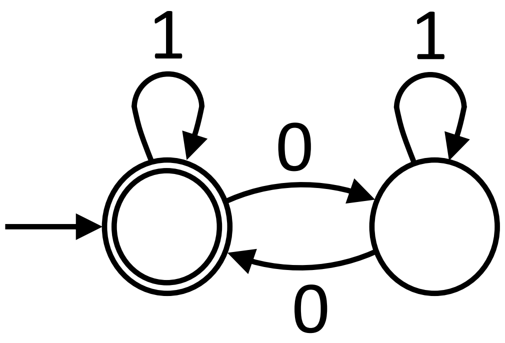
Example with Regular Language

Pumping Lemma: Given a regular language L, **∃ 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. $|xy| \leq p.$

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



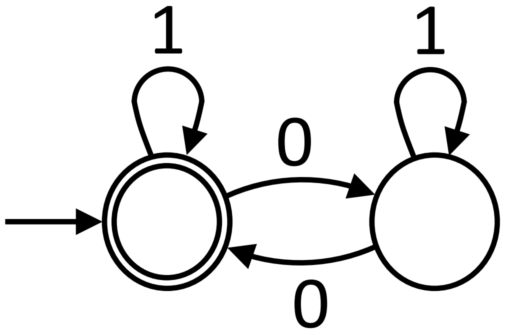
Example with Regular Language

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

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = 01110$$

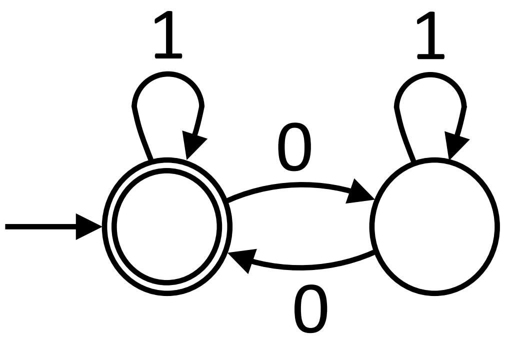
Example with Regular Language

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, $\boldsymbol{s = xyz}$** satisfying:

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \underbrace{0}_{x}|\underbrace{1}_{y}|\underbrace{110}_{z}$$

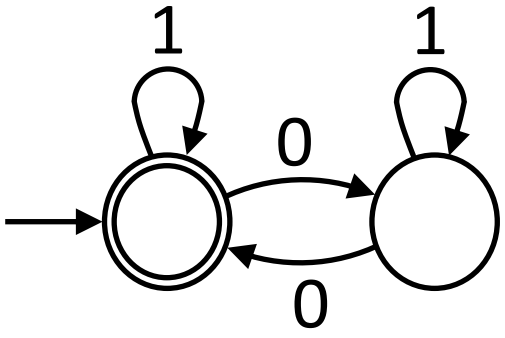
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p.$

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{1}|\overset{z}{110}$$

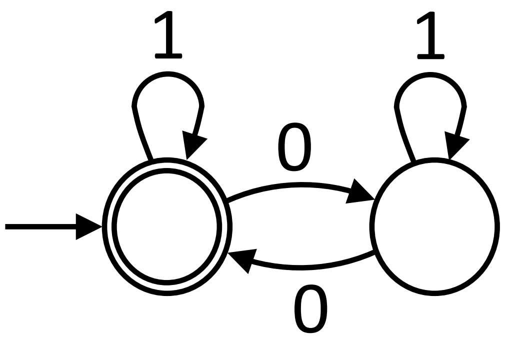
Example with Regular Language

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. **$|xy| \leq p$.**

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{1}|\overset{z}{110}$$

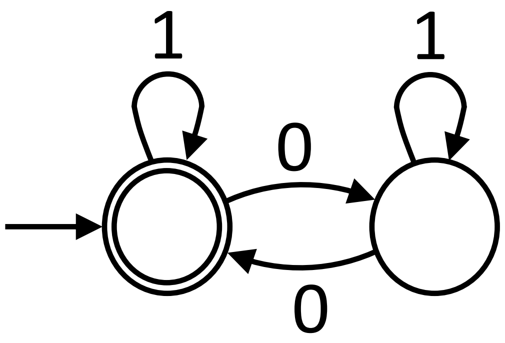
Example with Regular Language

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

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{1}|\overset{z}{110}$$

$$s' = \overset{x}{0}|\overset{y}{1}|\overset{y}{1}|\overset{z}{110}$$

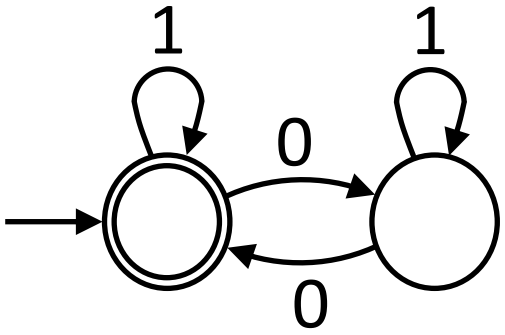
Example with Regular Language

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

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{1}|\overset{z}{110}$$

$$s' = \overset{x}{0}|\overset{y}{1}|\overset{y}{1}|\overset{z}{110}$$

$$s'' = \overset{x}{0}|\overset{y}{1}|\overset{y}{1}|\overset{y}{1}|\overset{y}{1}|\overset{z}{110}$$

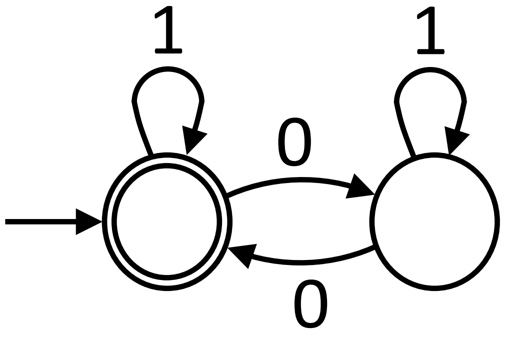
Example with Regular Language

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

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{1}|\overset{z}{110}$$

$$s' = \overset{x}{0}|\overset{y}{1}|\overset{y}{1}|\overset{z}{110}$$

$$s'' = \overset{x}{0}|\overset{y}{1}|\overset{y}{1}|\overset{y}{1}|\overset{y}{1}|\overset{z}{110}$$

$$s''' = \overset{x}{0}|\overset{z}{110}$$

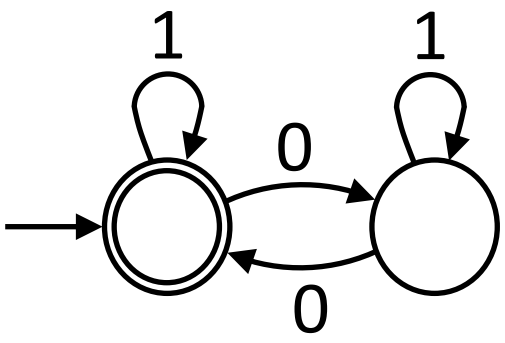
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p.$

$\{\omega : \omega \text{ contains an even number of 0's}\}$

$$p = 2$$



$$s = 0000$$

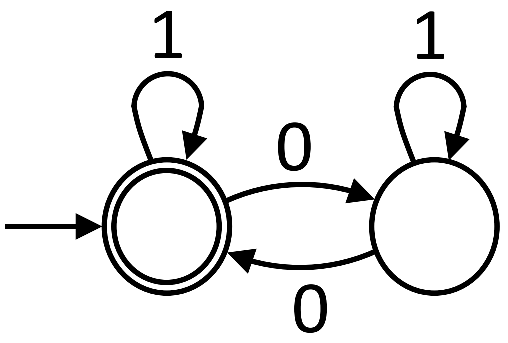
Example with Regular Language

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. $|xy| \leq p.$

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00}$$

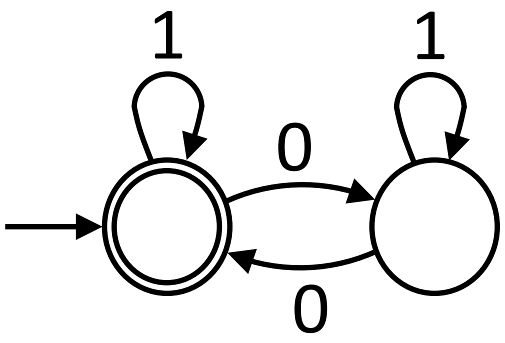
Example with Regular Language

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. $|\boldsymbol{y}| > \boldsymbol{0}.$
3. $|xy| \leq p.$

$\{\omega \colon \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{\boldsymbol{x}}{0}\,|\,\overset{\boldsymbol{y}}{0}\,|\,\overset{\boldsymbol{z}}{00}$$

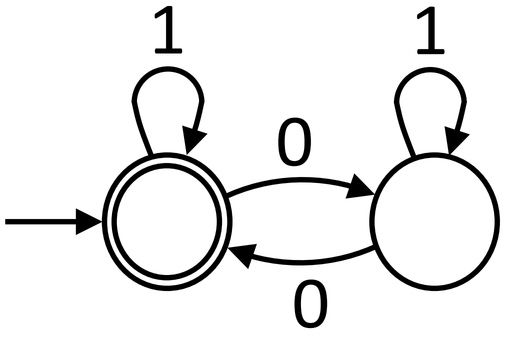
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $\boldsymbol{|xy| \leq p.}$

$\{\omega : \omega \text{ contains an even number of } 0\text{'s}\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{0}|\overset{z}{00}$$

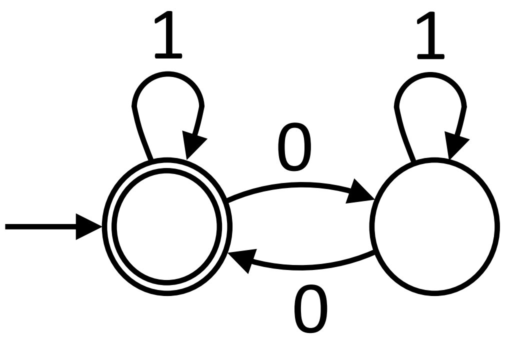
Example with Regular Language

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

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

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00}$$

$$s' = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00}$$

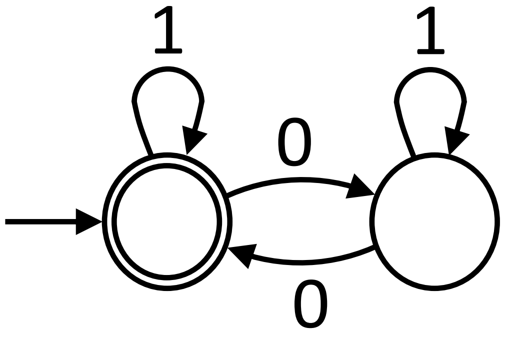
Example with Regular Language

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. $|xy| \leq p.$

$\{\omega : \omega$ contains an even number of 0's$\}$

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{0}|\overset{z}{0\,0}$$

$$s' = \overset{x}{0}|\overset{y}{0}|\overset{y}{0}|\overset{z}{0\,0} \notin L$$

Example with Regular Language

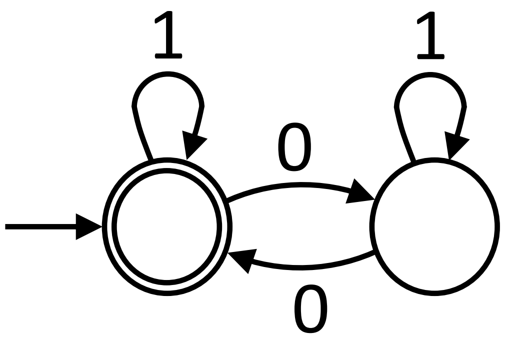
<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

s <u>**can**</u> be divided into three pieces such that…

$\{\omega: \omega$ contains an even

$$p = 2$$



$$s = \overset{x}{0}|\overset{y}{0}|\overset{z}{00}$$

$$s' = \overset{x}{0}|\overset{y}{0}|\overset{y}{0}|\overset{z}{00} \notin L$$

Example with Regular Language

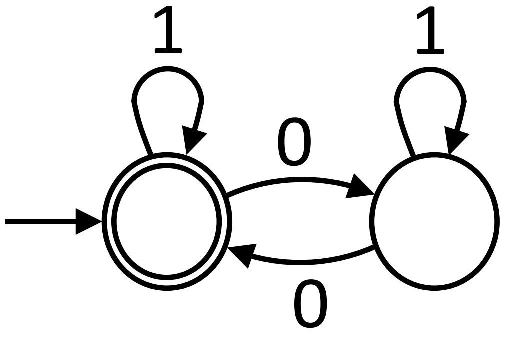
Pumping Lemma: Given a regular language L, $\exists$ a number p such that any string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

s **can** be divided into three pieces such that…

$\{\omega : \omega$ contains an even

$p = 2$



$$s = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00}$$

$$s = \overset{y}{00}\,|\,\overset{z}{00}$$

$$s' = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00} \notin \boldsymbol{L}$$

Example with Regular Language

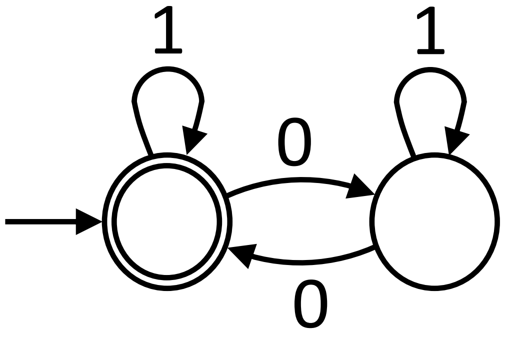
<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

$\{\omega : \omega$ contains an even

s **<u>can</u>** be divided into three pieces such that…

$p = 2$



$s = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00}$

$s = \overset{y}{00}\,|\,\overset{z}{00}$

$s' = \overset{x}{0}\,|\,\overset{y}{0}\,|\,\overset{y}{0}\,|\,\overset{z}{00} \notin L$

Example with Regular Language

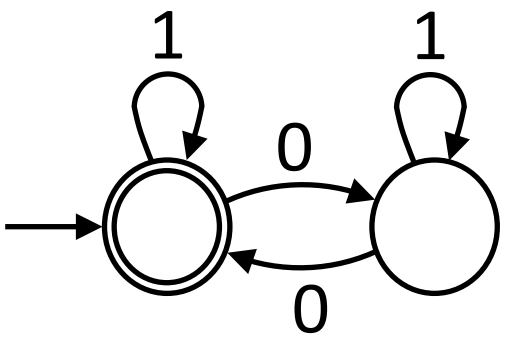
<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

s **<u>can</u>** be divided into three pieces such that…

$\{\omega : \omega$ contains an even

$p = 2$



$s = \overset{x}{0}|\overset{y}{0}|\overset{z}{00}$

$s = \overset{y}{00}|\overset{z}{00}$

$s' = \overset{x}{0}|\overset{y}{0}|\overset{y}{0}|\overset{z}{00} \notin L$

Example with Regular Language

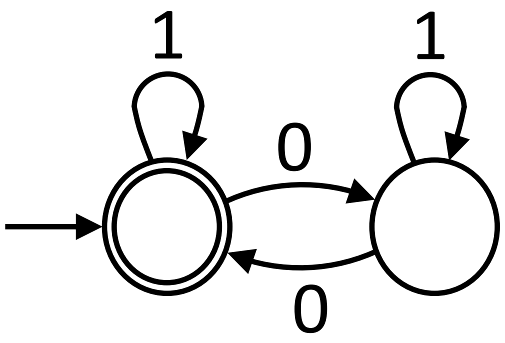
<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> string $s \in L$, with $|s| \geq p$, can be divided into three pieces, $s = xyz$ satisfying:

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

$\{\omega : \omega$ contains an even

s <u>**can**</u> be divided into three pieces such that…

$p = 2$



$s = \overset{x}{0}|\overset{y}{0}|\overset{z}{00}$

$s' = \overset{x}{0}|\overset{y}{0}|\overset{y}{0}|\overset{z}{00} \notin \boldsymbol{L}$

$s = \overset{y}{00}|\overset{z}{00}$

$s' = \overset{y}{00}|\overset{y}{00}|\overset{z}{00}$

Example with Regular Language

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:

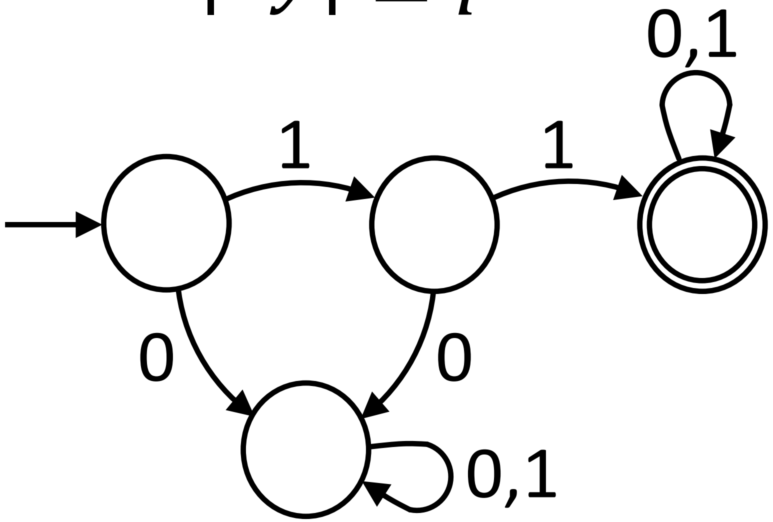
1. $xy^iz \in L, \forall i \geq 0$.
2. $|y| > 0$.
3. $|xy| \leq p$.

Example with Regular Language

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. $|xy| \leq p$.

$\{\omega : \omega \text{ starts with } 11\}$



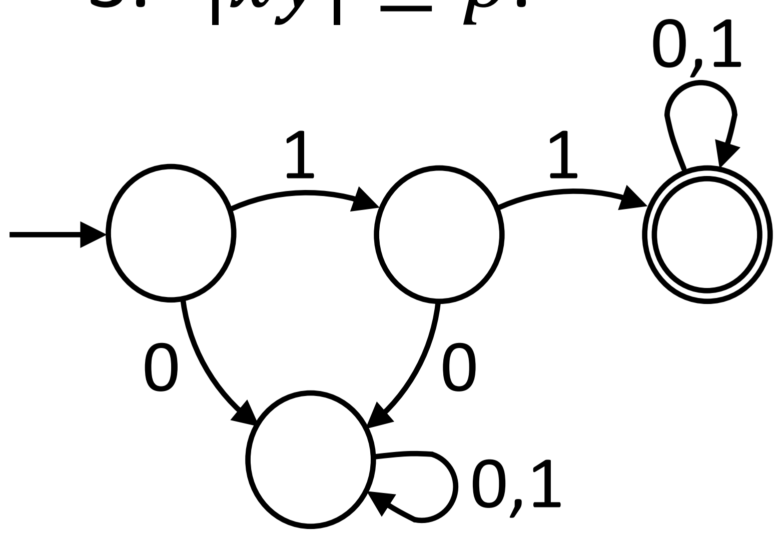
Example with Regular Language

Pumping Lemma: Given a regular language L, **∃ a number $\boldsymbol{p}$** 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. $|xy| \leq p.$

$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$



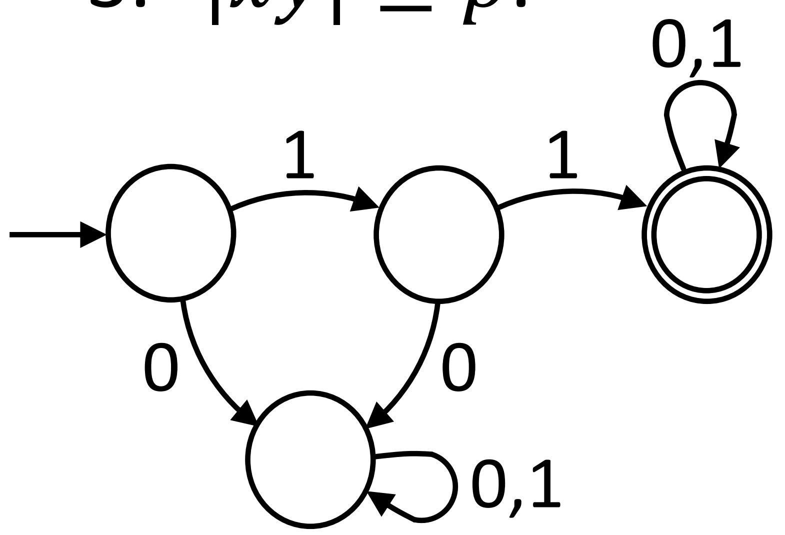
Example with Regular Language

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. $|xy| \leq p.$

$\{\omega\colon \omega \text{ starts with } 11\}$

$p = 4$

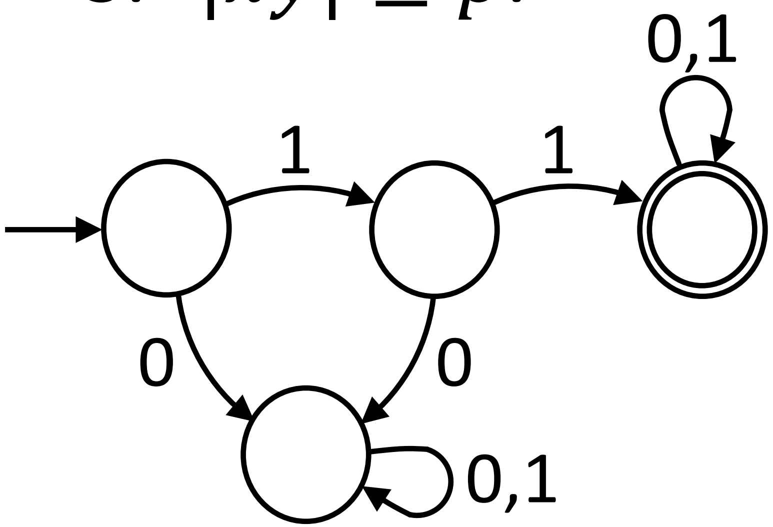


$s = 1110$

Example with Regular Language

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. $|xy| \leq p.$



$\{\omega: \omega \text{ starts with } 11\}$

$p = 4$

$$s = 1\overset{y}{1}10$$

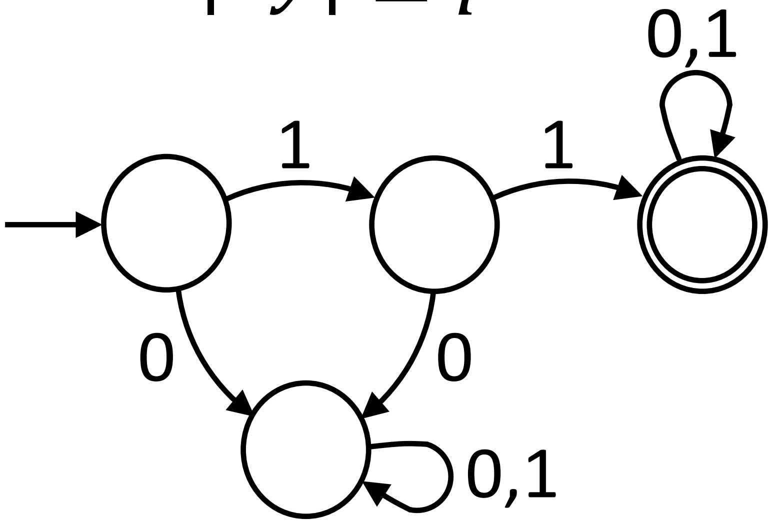
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p.$

$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$



$$s = 1\overset{y}{1}10$$

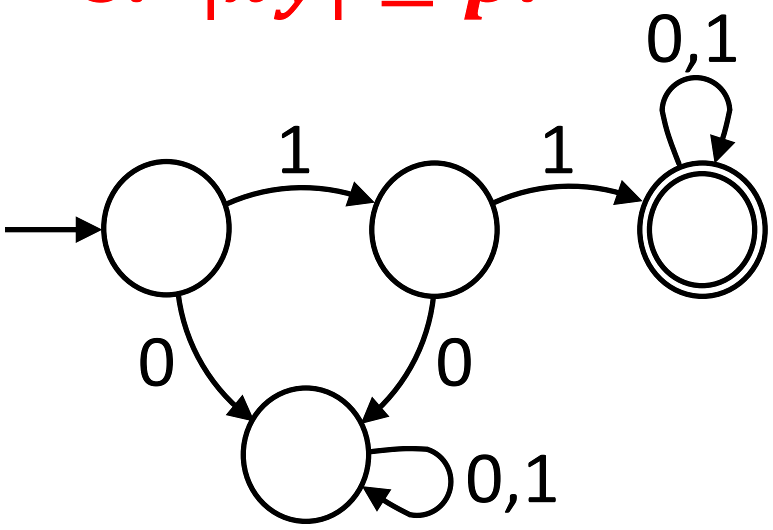
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. **$|xy| \leq p.$**

$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$



$\boldsymbol{y}$

$s = 1110$

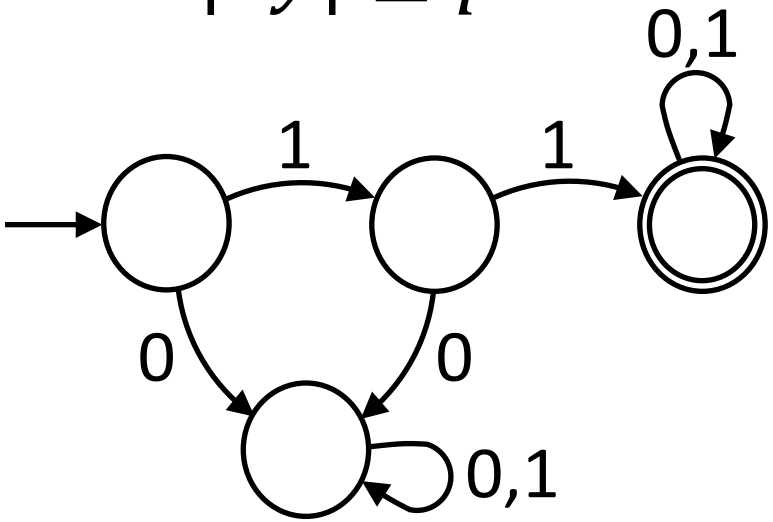
Example with Regular Language

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

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

$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$



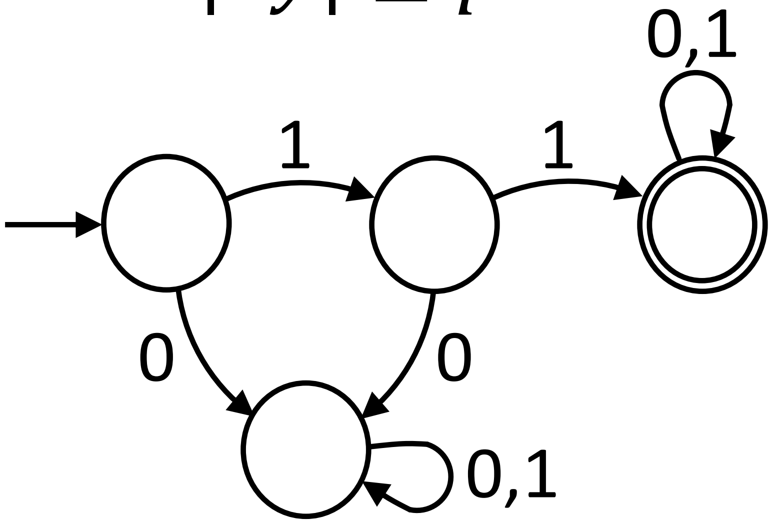
$\boldsymbol{y}$

$s = 1110$

Example with Regular Language

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

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



$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$

$$s = 11\overset{y}{1}0$$

What if $i = 0$? ✘

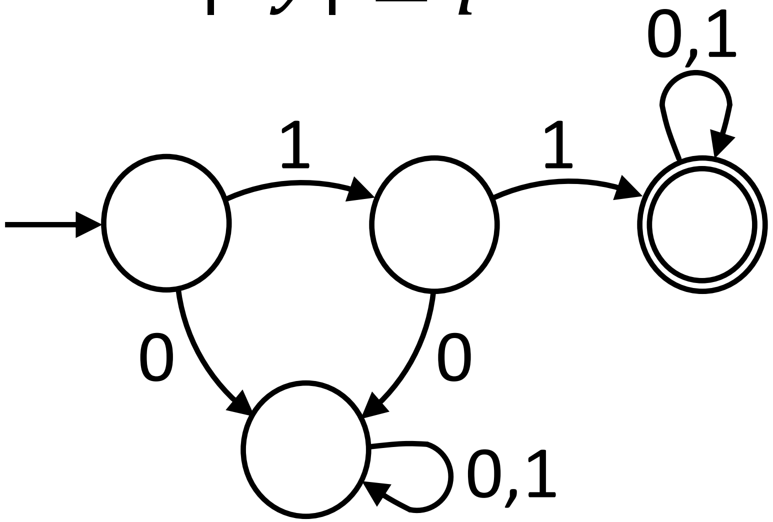
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p.$

$$\{\omega : \omega \text{ starts with } 11\}$$

$$p = 4$$



$$s = \overset{y}{1}110 \longleftarrow$$ What if $i = 0$? ✗

$$s = \overset{x}{11}|\overset{y}{10}$$

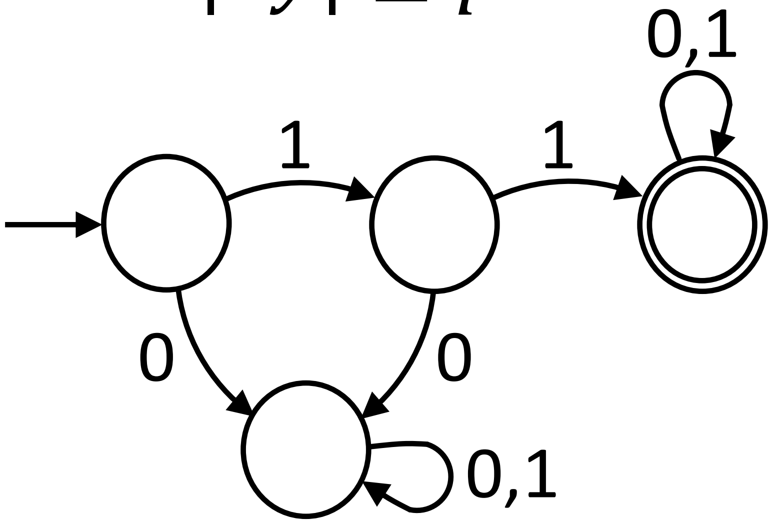
Example with Regular Language

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. $|xy| \leq p$.

$$\{\omega : \omega \text{ starts with } 11\}$$

$$p = 4$$



$$s = 11\overset{y}{1}0 \longleftarrow \text{What if } i = 0?$$

$$s = \overset{x}{11}|\overset{y}{10}$$

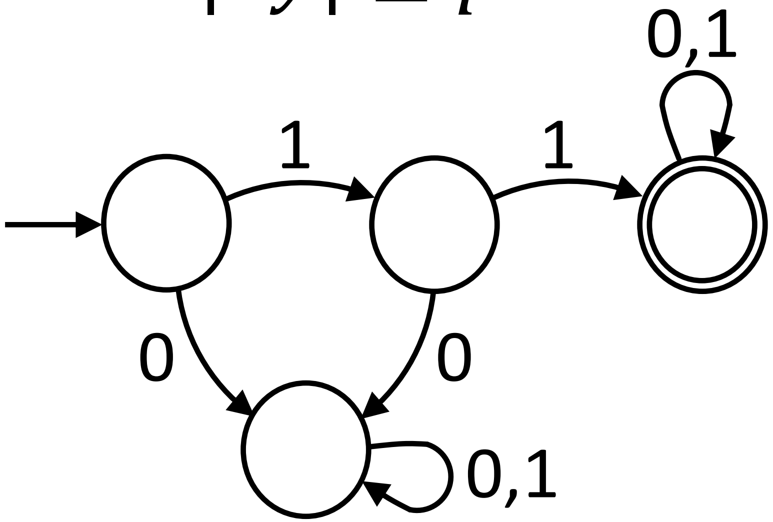
Example with Regular Language

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p such that <u>any</u> 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. $|xy| \leq p$.

$\{\omega : \omega \text{ starts with } 11\}$

$p = 4$



$s = 1\overset{y}{1}10$ ← What if $i = 0$? ✗

$s = \overset{x}{11}|\overset{y}{10}$ ✓

$s = \overset{x}{1}|\overset{y}{1}|\overset{z}{10}$

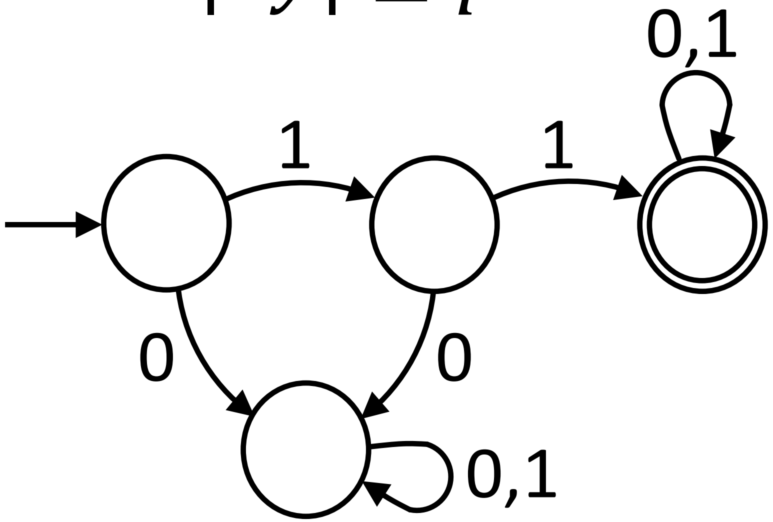
Example with Regular Language

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. $|xy| \leq p.$

$$\{\omega: \omega \text{ starts with } 11\}$$

$$p = 4$$



$$s = \overset{y}{1110}$$ ⟵ What if $i = 0$? ✗

$$s = \overset{x}{11}|\overset{y}{10}$$ ✓

$$s = \overset{x}{1}|\overset{y}{1}|\overset{z}{10}$$ ✓

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.

Non-Regularity Proofs

<u>Pumping Lemma</u>: Given a regular language L, **∃ a number $\boldsymbol{p}$** such that <u>any</u> 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.

Non-Regularity Proofs

<u>Pumping Lemma</u>: Given a regular language L, $\exists$ a number p **such that <u>any</u> string $\boldsymbol{s} \in \boldsymbol{L}$, with $|\boldsymbol{s}| \geq \boldsymbol{p}$**, 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, $\boldsymbol{s = xyz}$ satisfying:**

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

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. $xy^iz \in L, \forall i \geq 0$.
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.
6. Show new string is not in language.