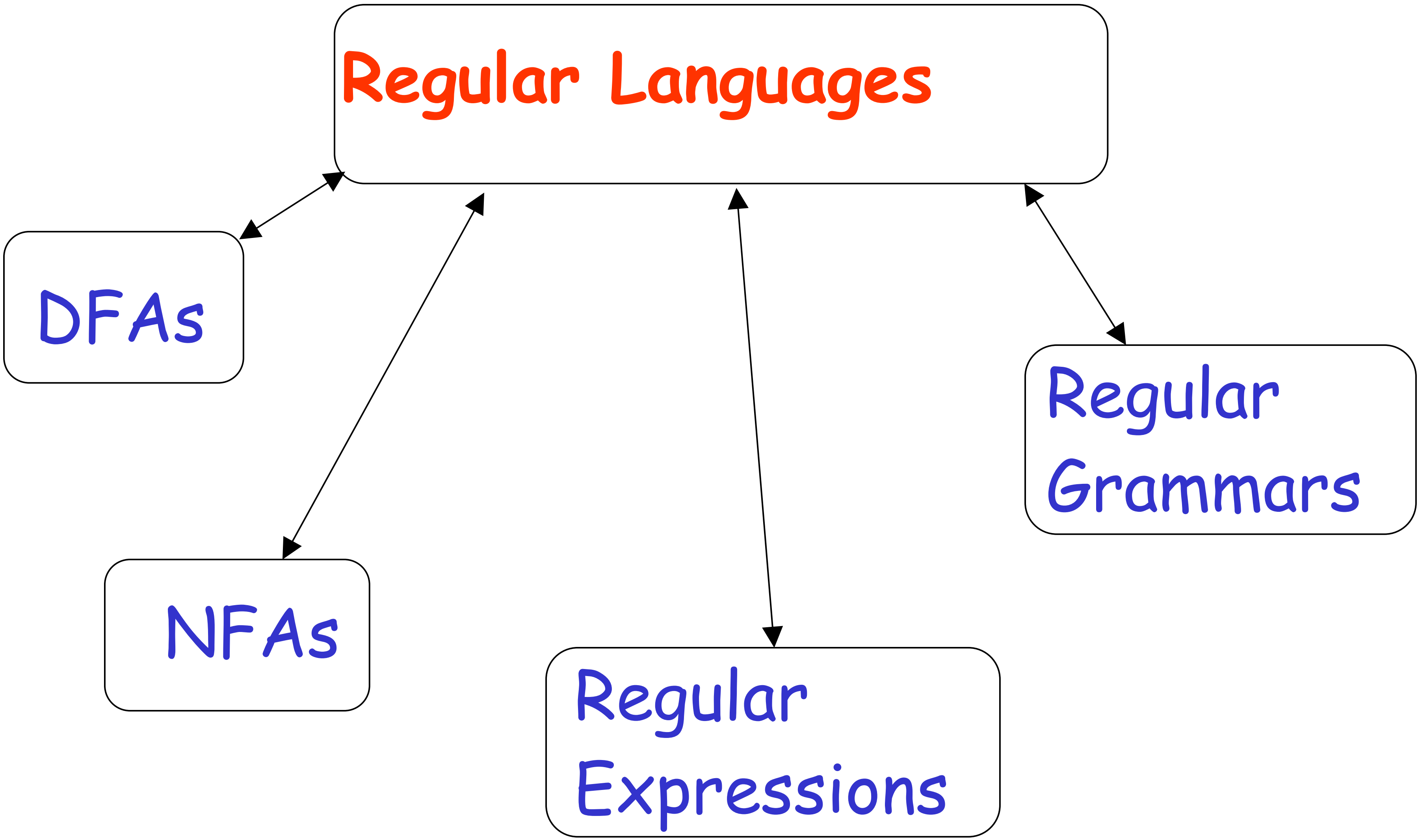
Standard Representations of Regular Languages



When we say: We are given a Regular Language L

We mean: Language L is in a standard representation

Elementary Questions

about

Regular Languages

Membership Question

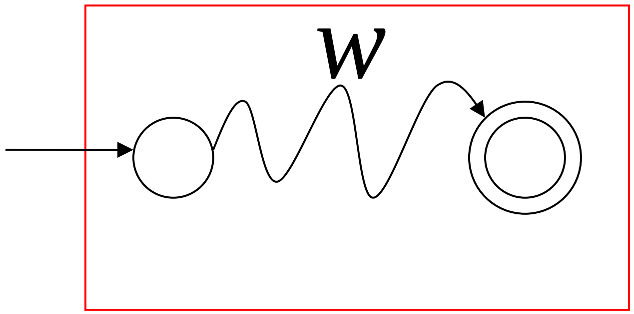
Question: Given regular language L
and string w
how can we check if $w \in L$?

Membership Question

Question: Given regular language L
and string w
how can we check if $w \in L$?

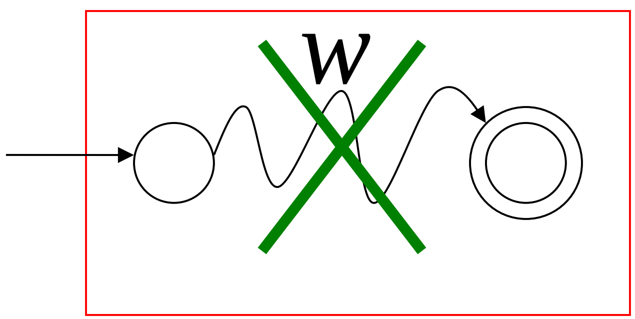
Answer: Take the DFA that accepts L
and check if w is accepted

DFA



$$w \in L$$

DFA



$$w \notin L$$

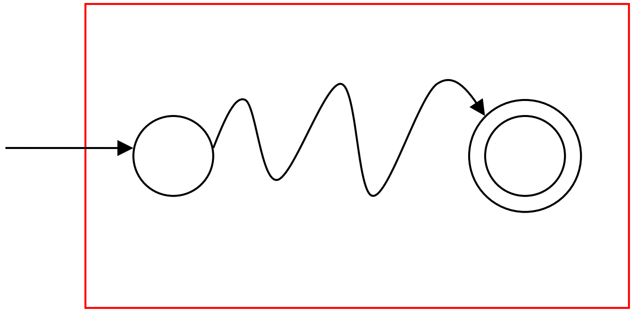
Question: Given regular language L
how can we check
if L is empty: $(L = \emptyset)$?

Question: Given regular language L how can we check if L is empty: $(L = \varnothing)$?

Answer: Take the DFA that accepts L

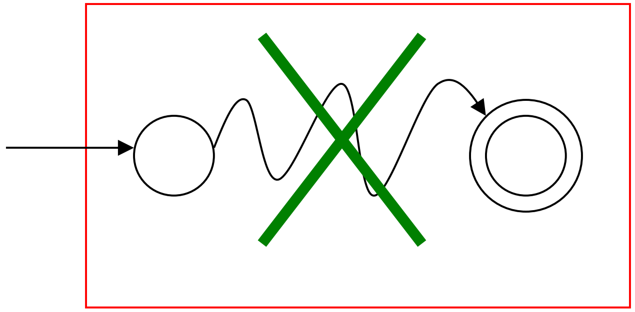
Check if there is any path from the initial state to a final state

DFA



$$L \neq \varnothing$$

DFA



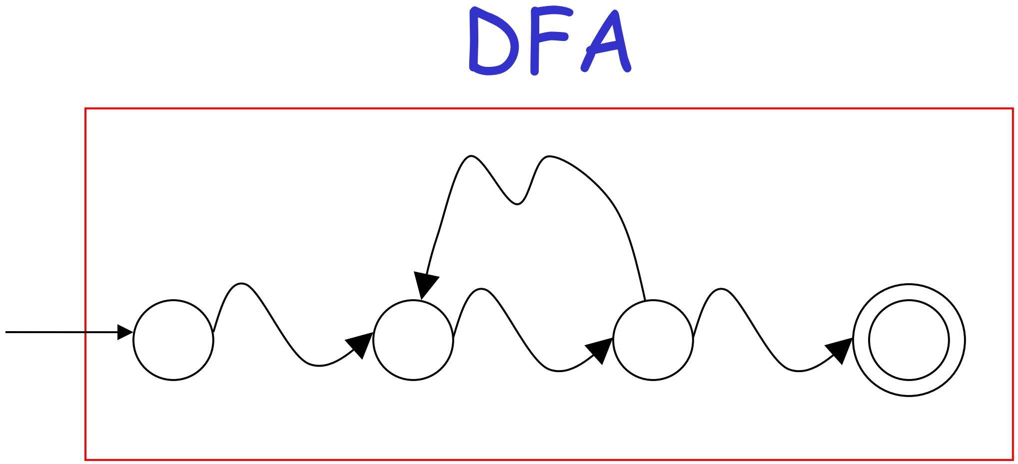
$$L = \varnothing$$

Question: Given regular language L
how can we check
if L is finite?

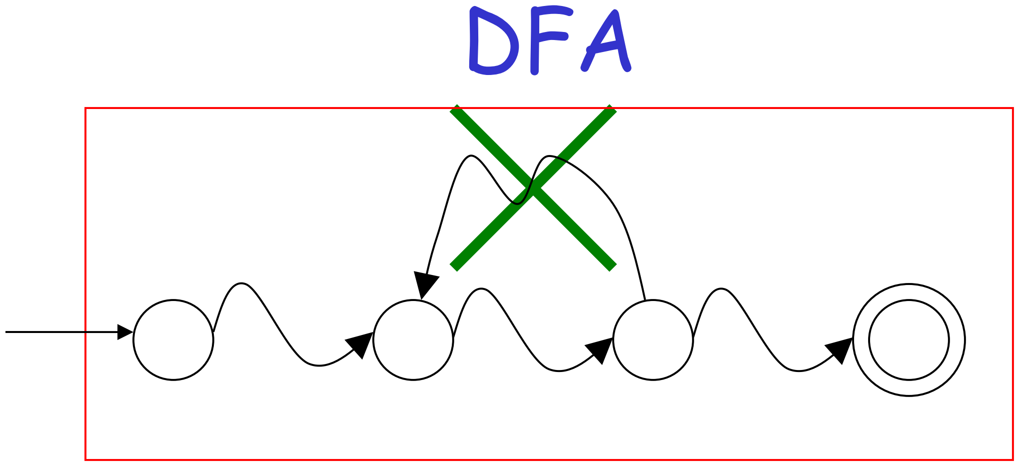
Question: Given regular language L
how can we check
if L is finite?

Answer: Take the DFA that accepts L

Check if there is a walk with cycle
from the initial state to a final state



L is infinite



L is finite

Question:

Given regular languages L_1 and L_2

how can we check if $L_1 = L_2$?

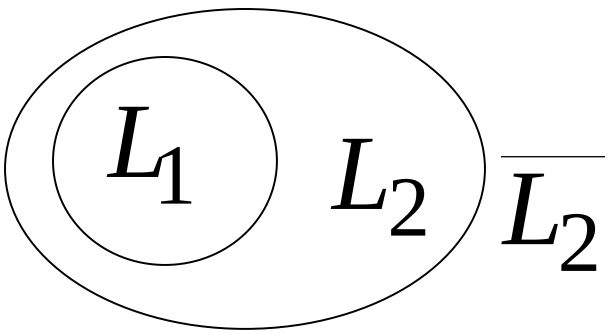
Question:

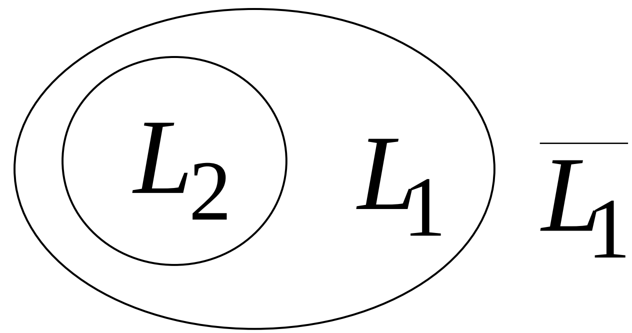
Given regular languages L_1 and L_2
how can we check if $L_1 = L_2$?

Answer: Find if $(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2) = \emptyset$

$$(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2) = \varnothing$$

$$L_1 \cap \overline{L_2} = \varnothing \quad \text{and} \quad \overline{L_1} \cap L_2 = \varnothing$$





$$L_1 \subseteq L_2 \qquad L_2 \subseteq L_1$$

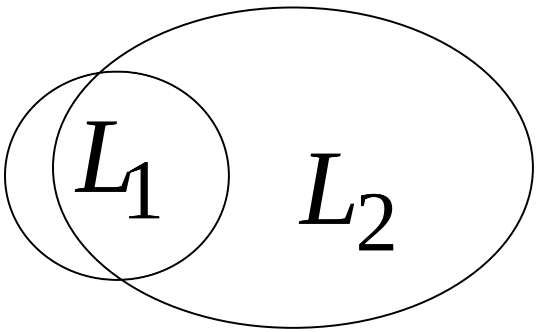
$$L_1 = L_2$$

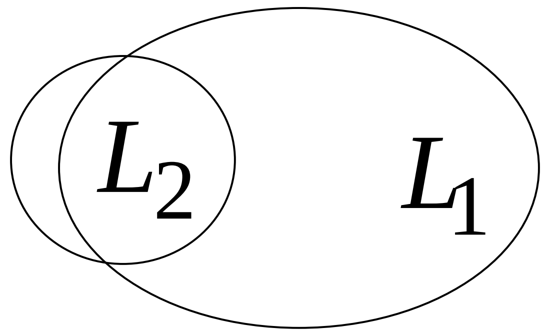
$$(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2) \neq \emptyset$$

$$L_1 \cap \overline{L_2} \neq \emptyset$$

or

$$\overline{L_1} \cap L_2 \neq \emptyset$$





$$L_1 \not\subset L_2 \qquad L_2 \not\subset L_1$$

$$L_1 \neq L_2$$

Closure Properties of Regular languages

Closure Properties of Regular languages

If L1 and L2 are regular languages, then so
$L1 \cup L2$; $L1 \cap L2$; $L1L2$, $L1^c$ and $L1^*$.

We say that the family of regular languages is closed under union, intersection, concatenation, complementation, and Star-closure.

Closure Properties of Regular languages

If L1 and L2 are regular languages, there exist regular expressions r1 and r2 such that L1 = L(r1) and L2 = L(r2).

By def^n, r1+r2, r1r2 and r1* are regular expressions denoting the languages L1 U L2; L1L2 and L1* respectively.

For complement of L1, we can design complemented automata (as discussed in previous classes).

For L1 ∩ L2, we can design product automata for given L1 and L2 (as discussed in previous classes).

Closure Properties of Regular languages

Claim 1: If L1 and L2 are regular, then L1 - L2 (set difference) is necessarily regular also ???

Claim 2: The family of regular languages is closed under reversal (L^R) ???

Claim 3: If L is a regular language, prove that the language $\{uv : u \in L,\ v \in L^R\}$ is also regular ???

Claim 4: Show that the family of regular languages is closed under symmetric difference ???

Claim 5: Show that the family of languages is closed under NOR operation ???

Non-regular languages

$$\{a^n b^n : \; n \geq 0\}$$

Non-regular languages

$$\{vv^R : \; v \in \{a,b\}^*\}$$

Regular languages

$$a^*b$$

$$b^*c+a$$

$$b+c(a+b)^*$$

$$etc...$$

How can we prove that a language L is not regular?

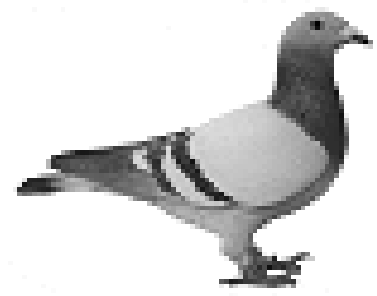
Prove that there is no DFA that accepts L

How can we prove that a language L is not regular?

Prove that there is no DFA that accepts L

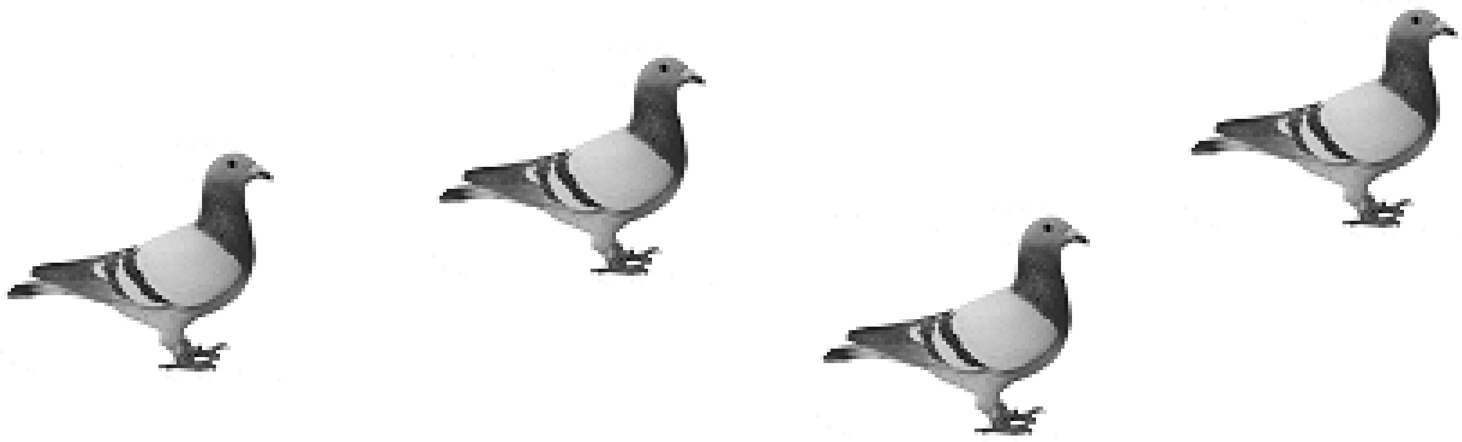
Problem: this is not easy to prove

Solution: the Pumping Lemma !!!

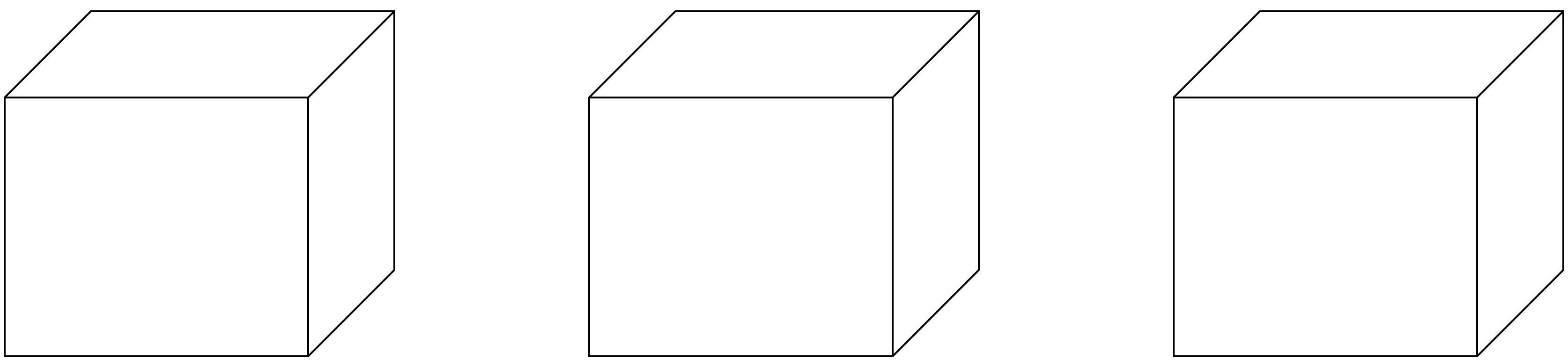


The Pigeonhole Principle

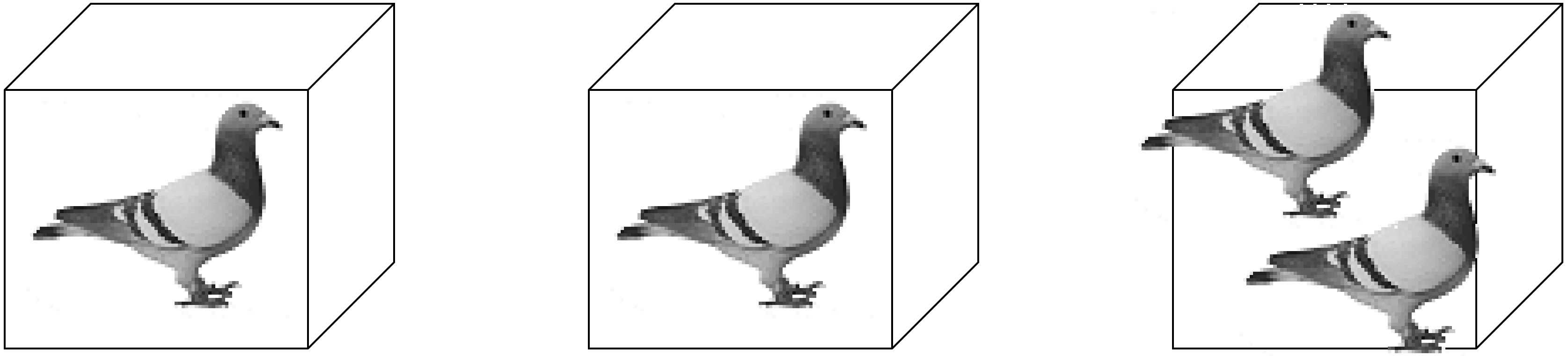
4 pigeons



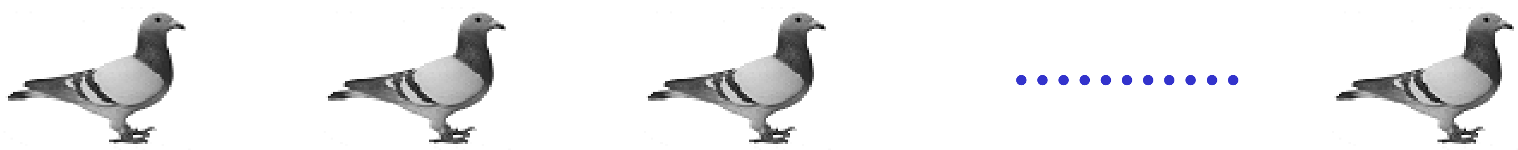
3 pigeonholes



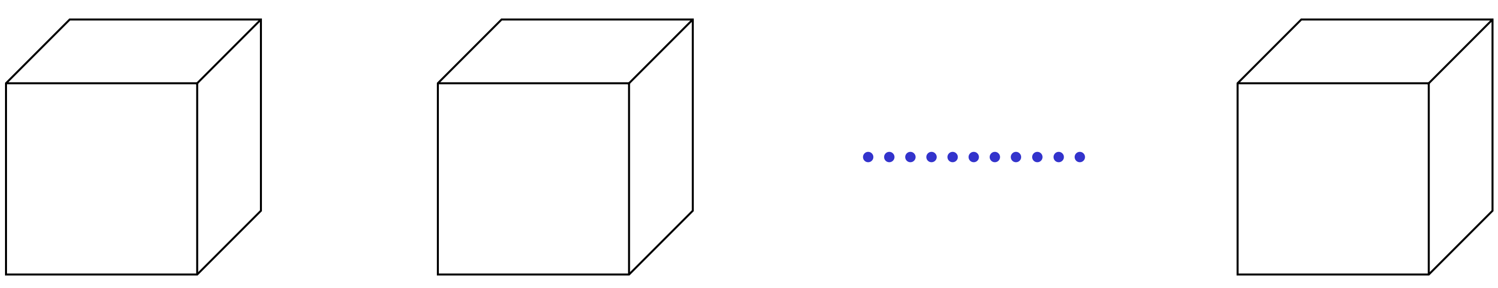
A pigeonhole must contain at least two pigeons



n pigeons



m pigeonholes $\qquad n > m$



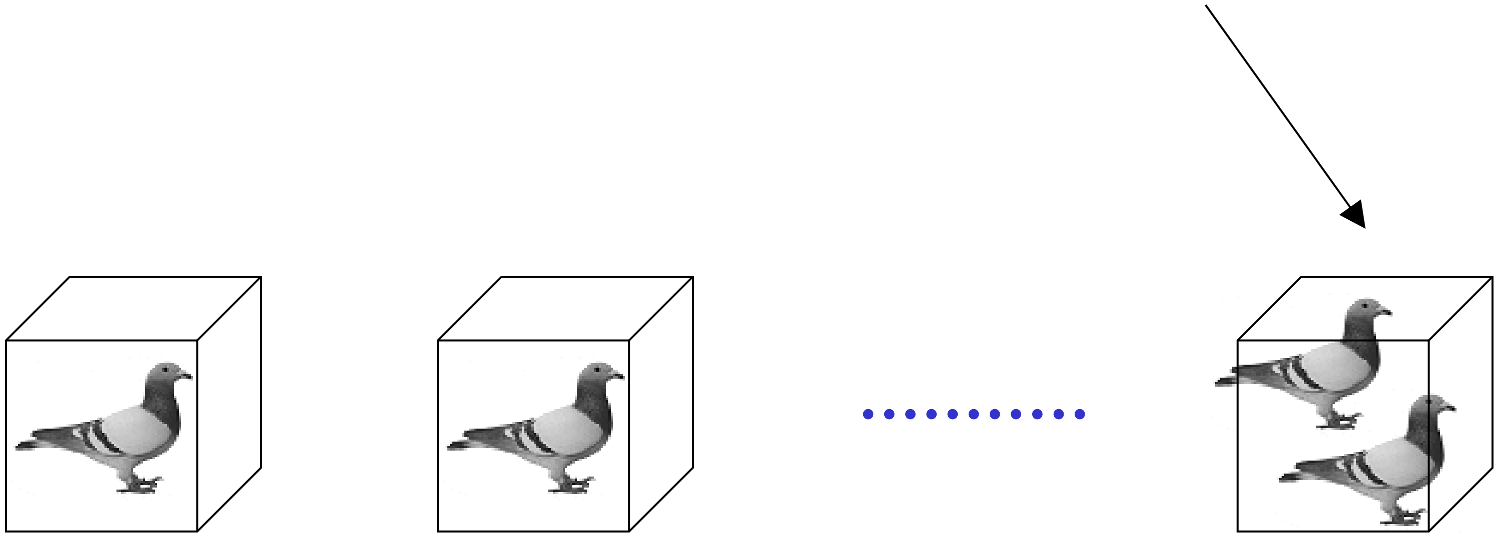
The Pigeonhole Principle

n pigeons

m pigeonholes

$$n > m$$

There is a pigeonhole with at least 2 pigeons

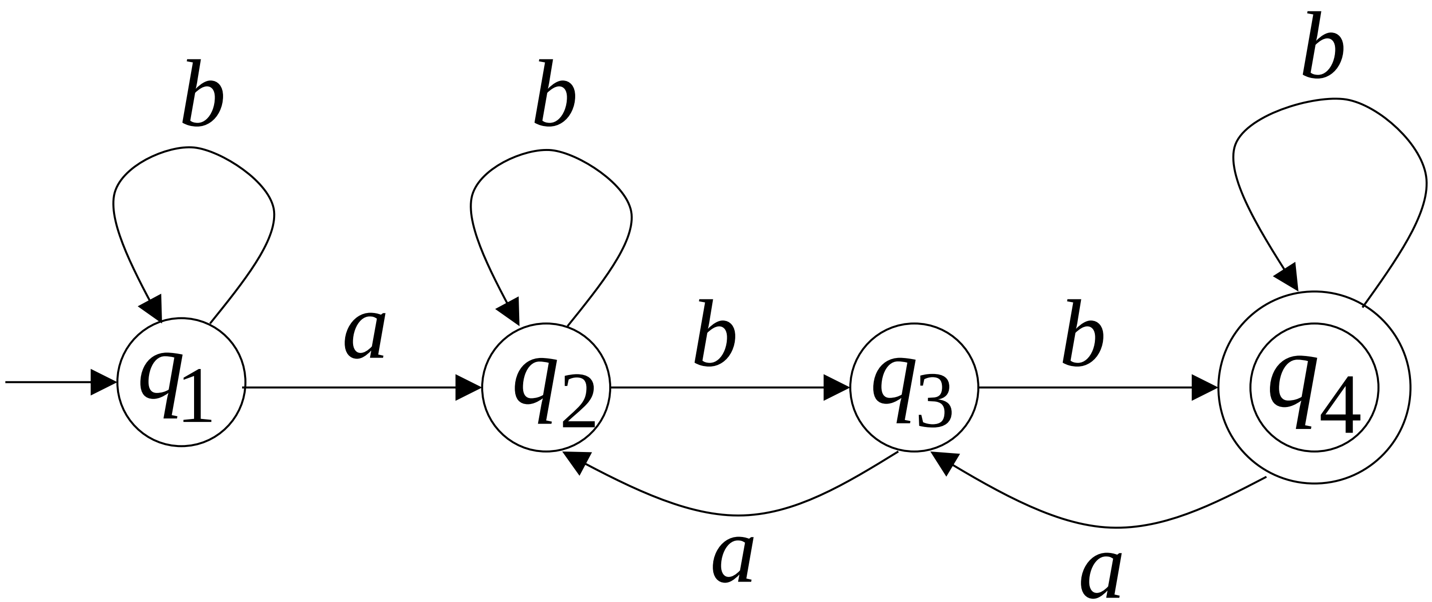


The Pigeonhole Principle

and

DFAs

DFA with 4 states

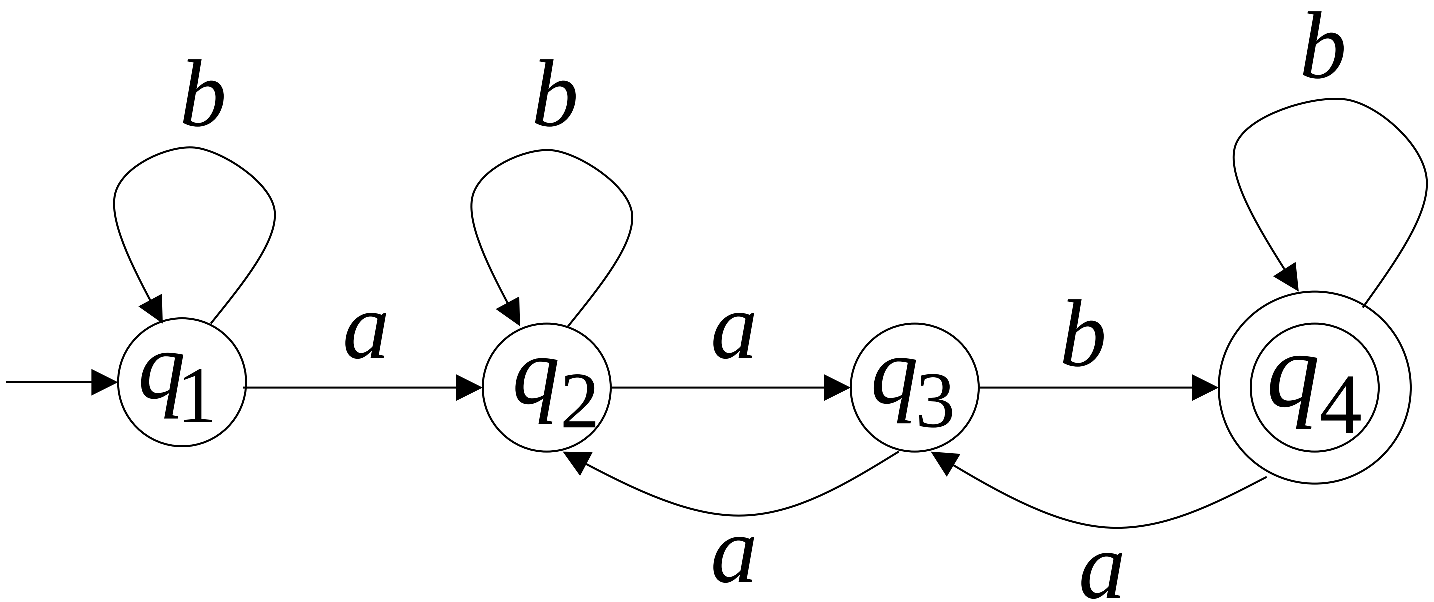


In walks of strings: a

aa

aab

no state
is repeated



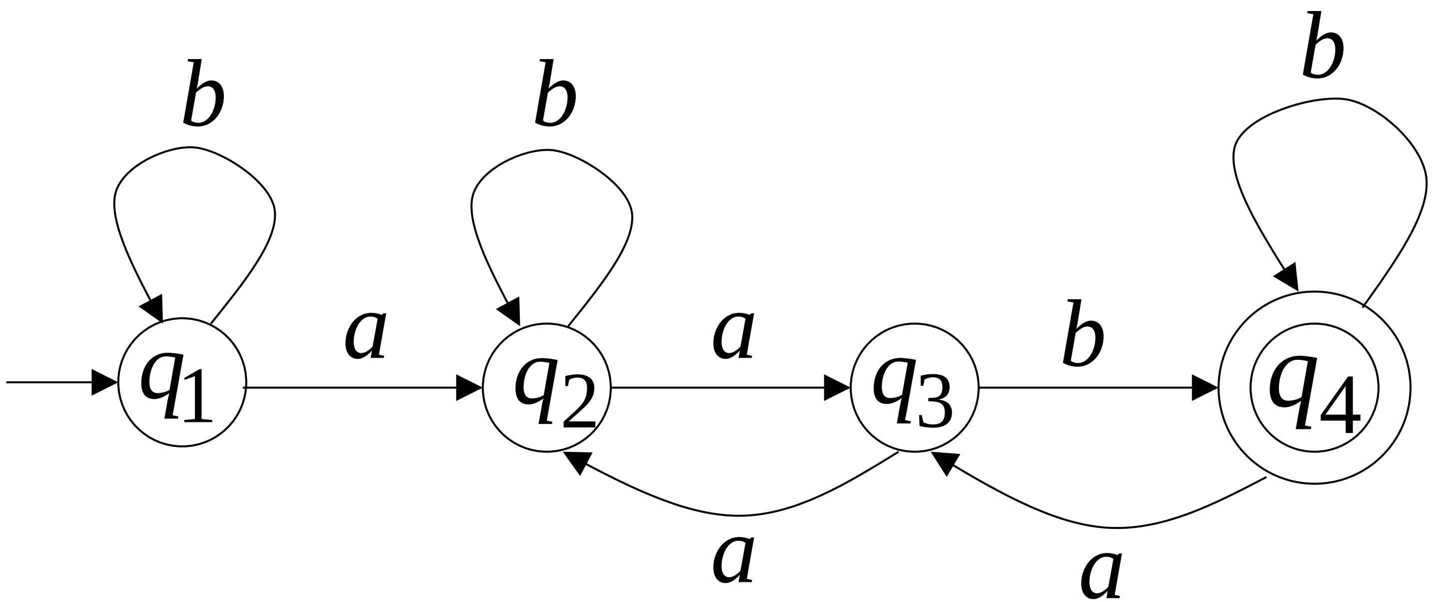
In walks of strings: $aabb$

$bbaa$

$abbabb$

$abbbabbabb...$

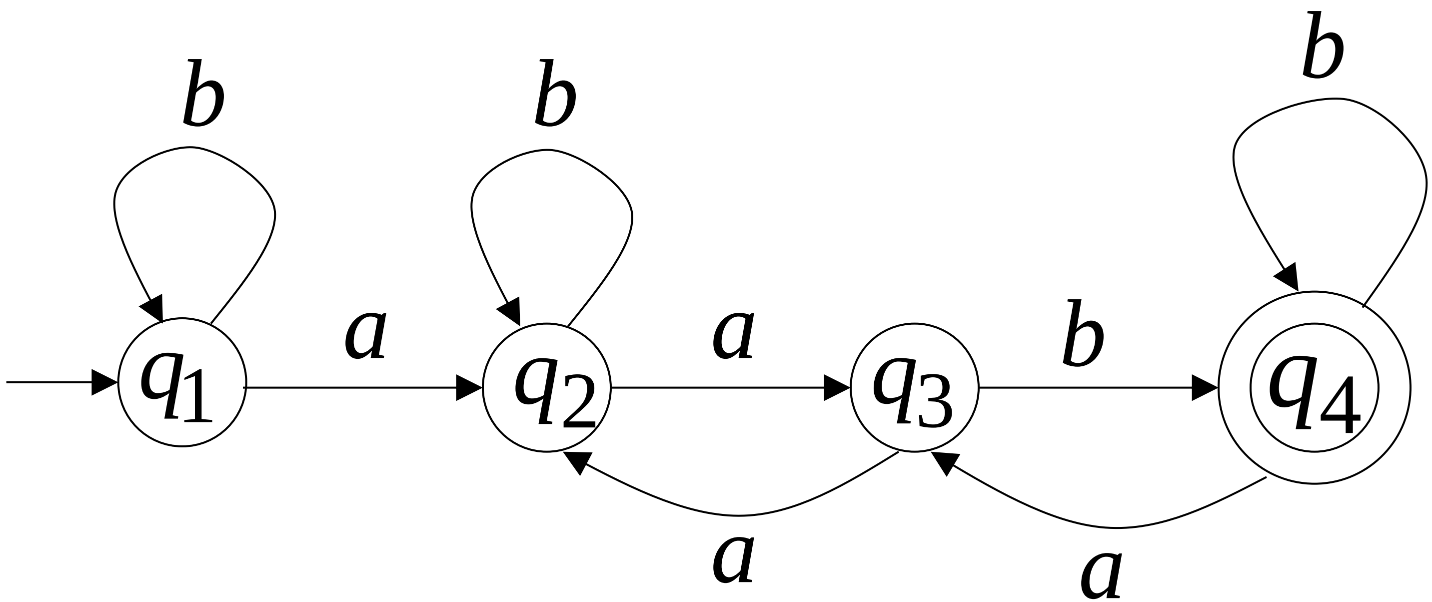
a state is repeated



If string w has length $|w| \geq 4$:

Then the transitions of string w are more than the states of the DFA

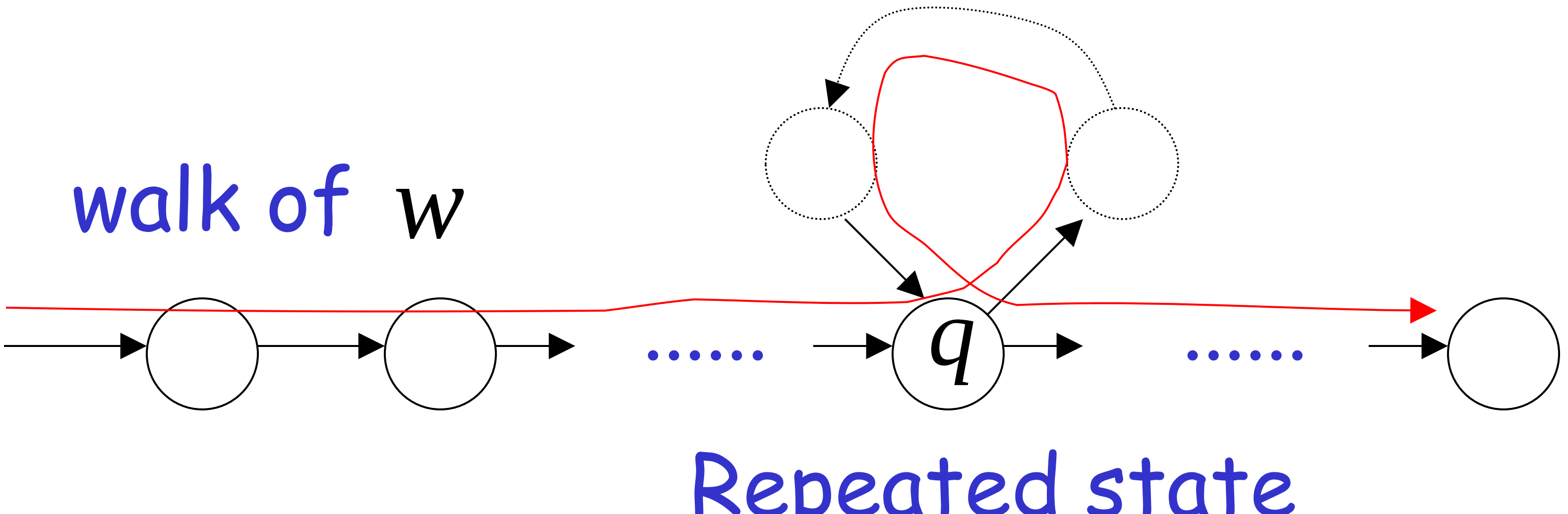
Thus, a state must be repeated



In general, for any DFA:

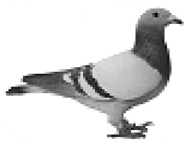
String w has length $\geq$ number of states

A state q must be repeated in the walk of w

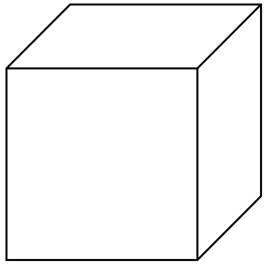


In other words for a string w:

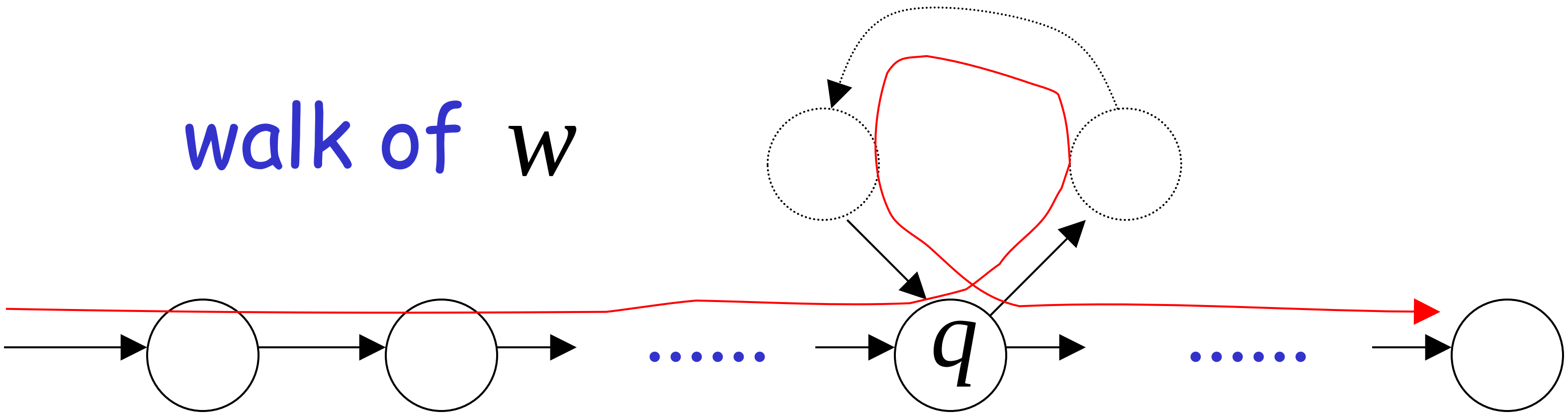
$\xrightarrow{a}$ transitions are pigeons



q states are pigeonholes



walk of w

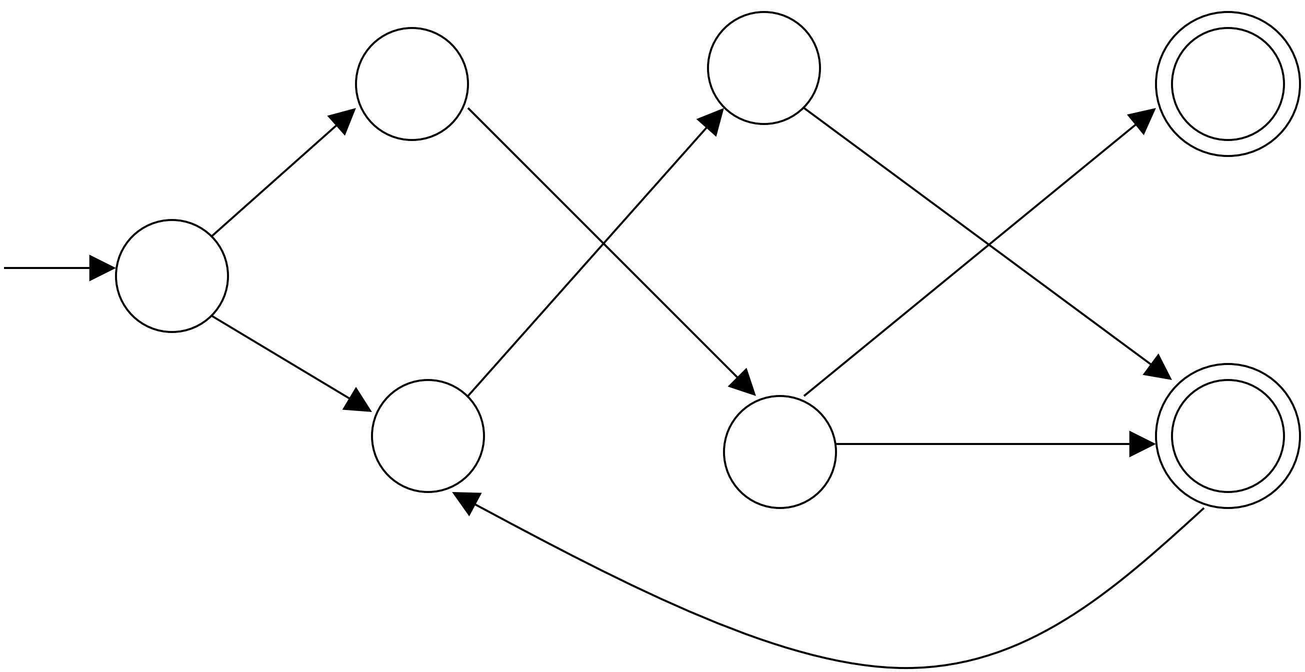


Repeated state

The Pumping Lemma

Take an infinite regular language L

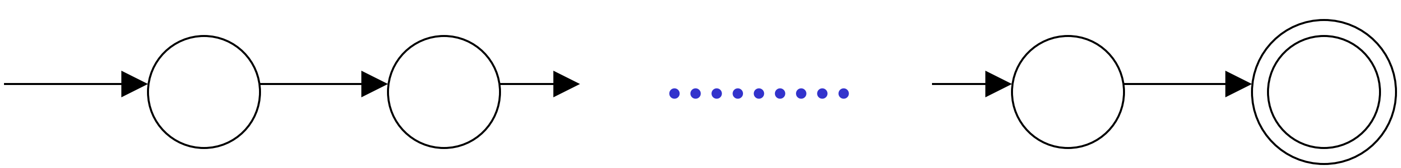
There exists a DFA that accepts L



m states

Take string w with $w \in L$

There is a walk with label w:

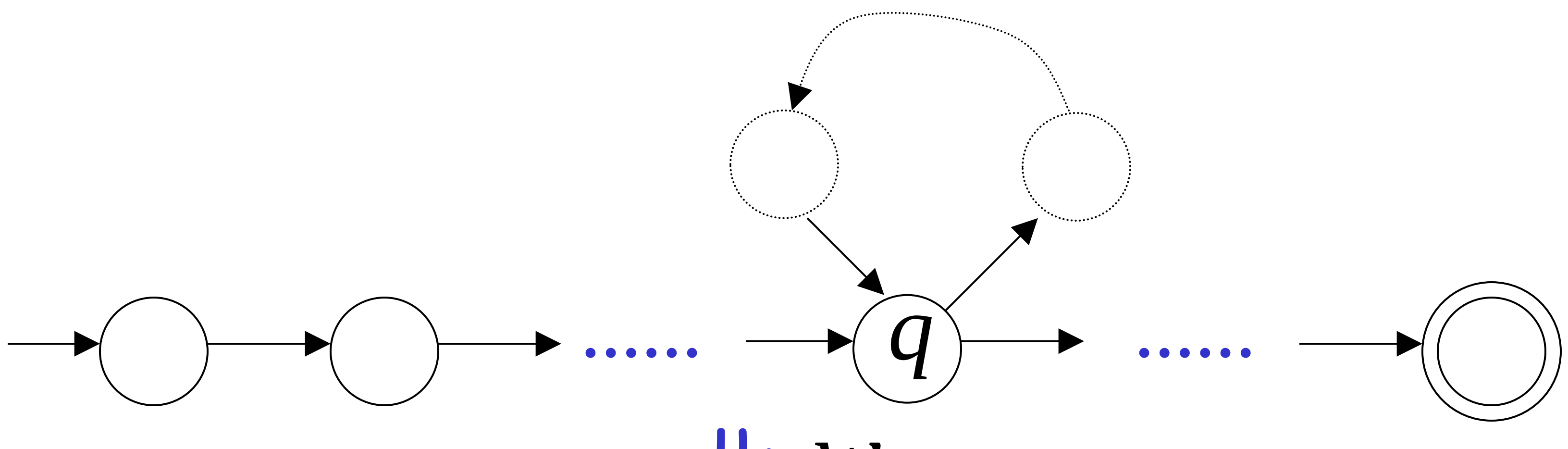


walk w

If string w has length $|w| \geq m$ (number of states of DFA)

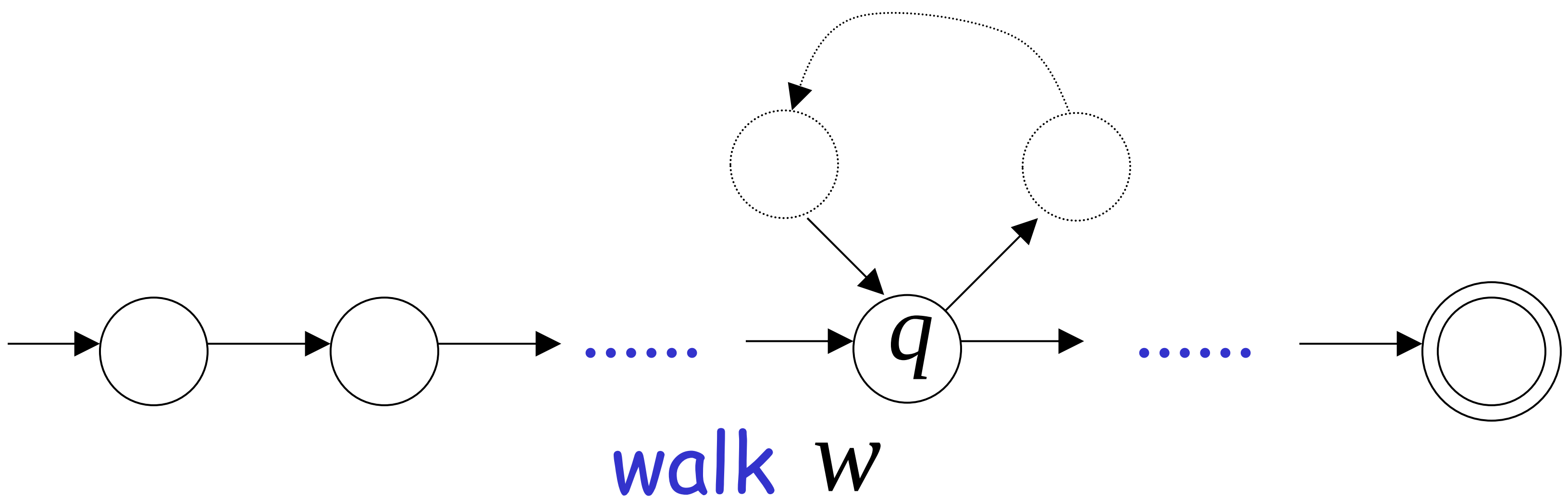
then, from the pigeonhole principle:

a state is repeated in the walk w

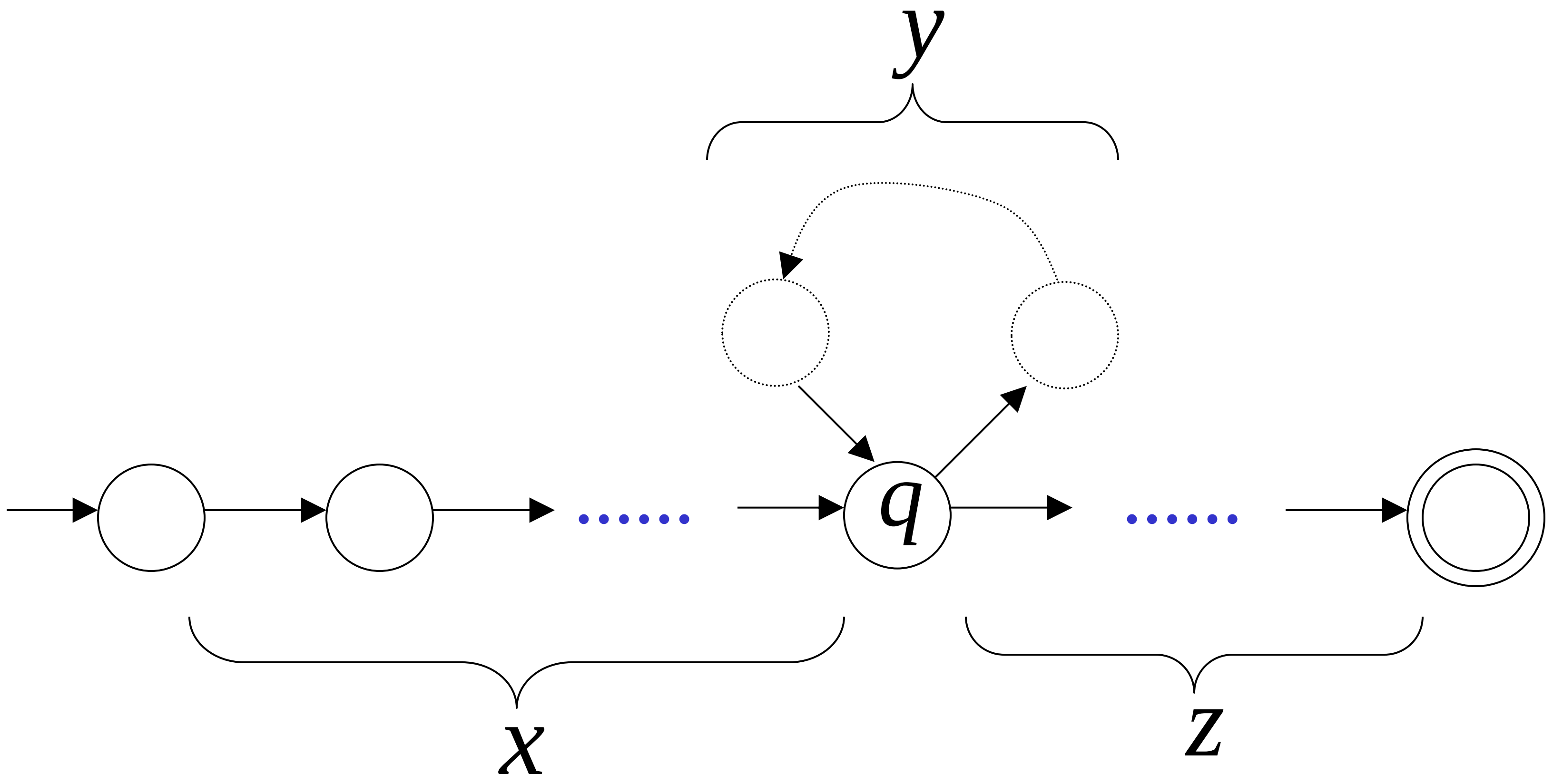


walk w

Let q be the first state repeated in the walk of w



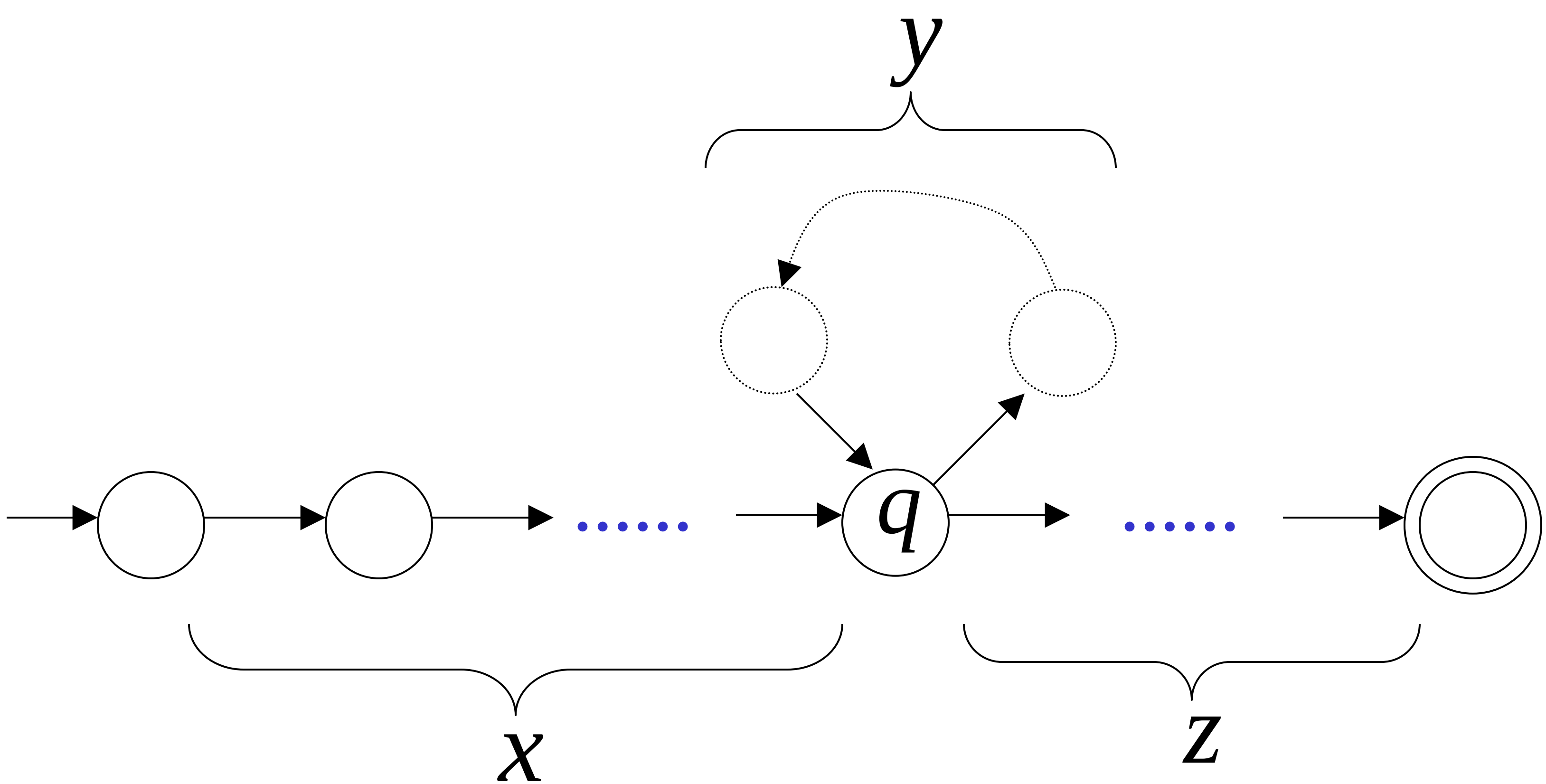
Write $w = x\ y\ z$



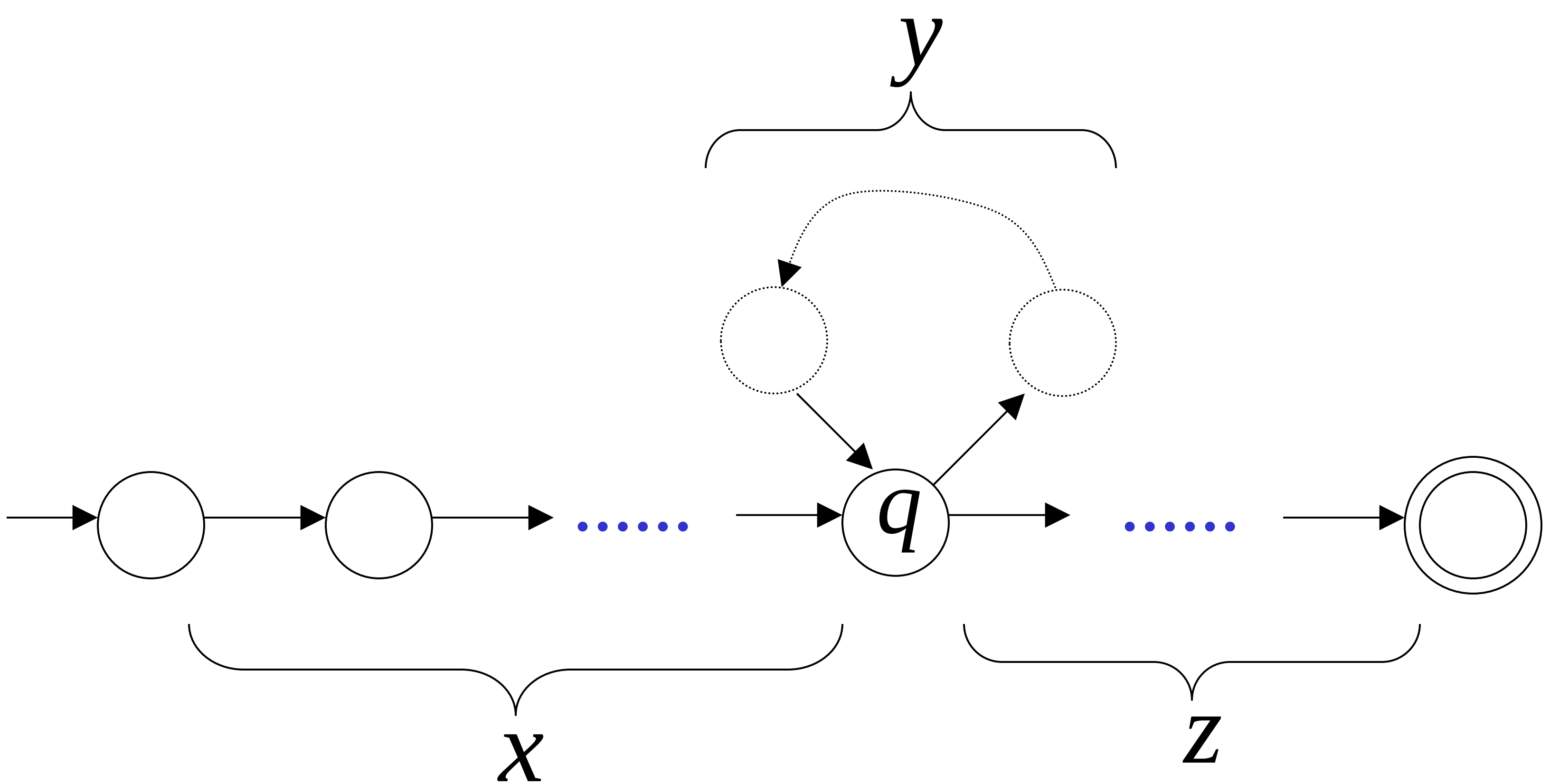
Observations:

length $|x\,y| \leq m$ number of states of DFA

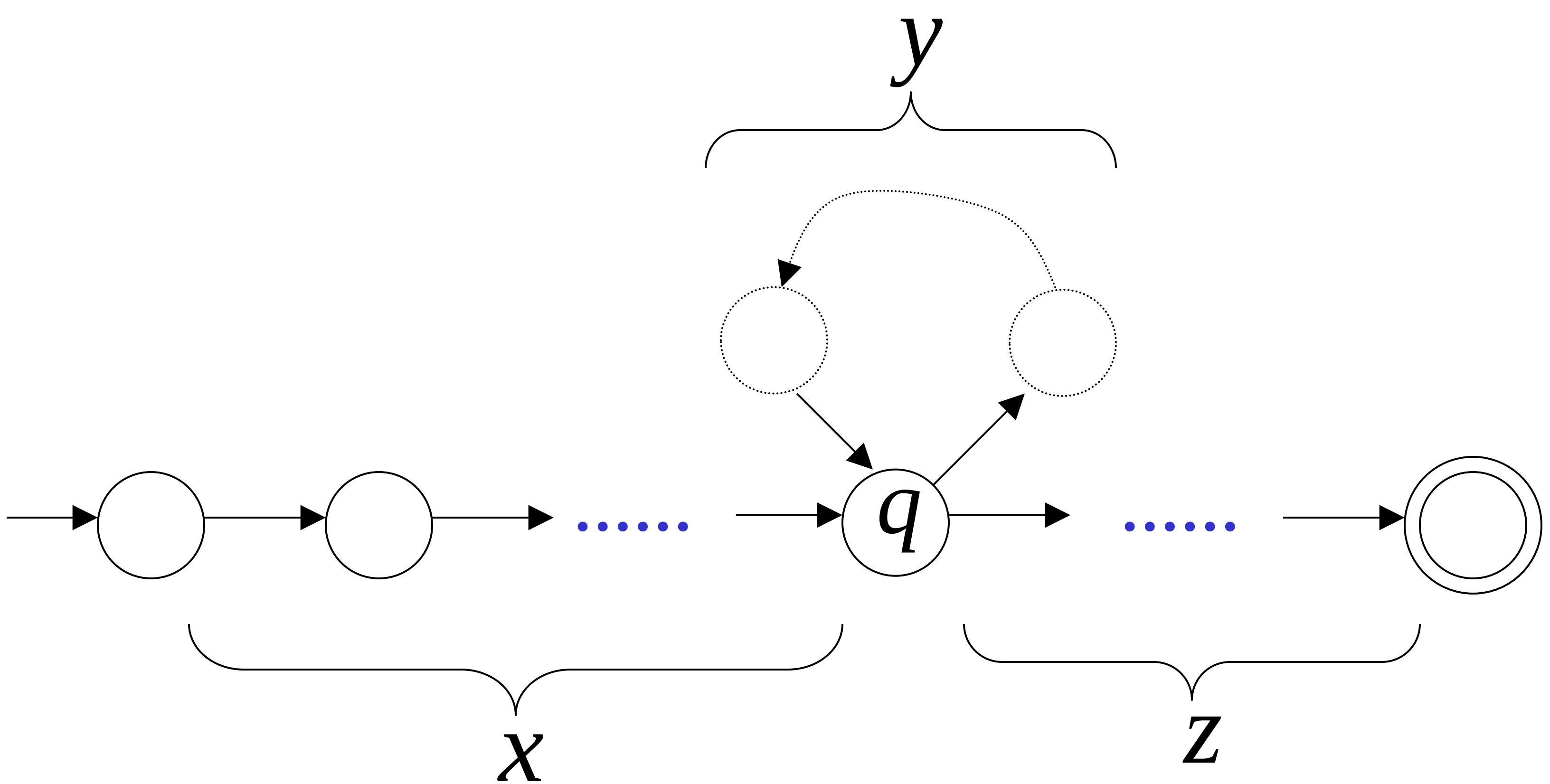
length $|y| \geq 1$



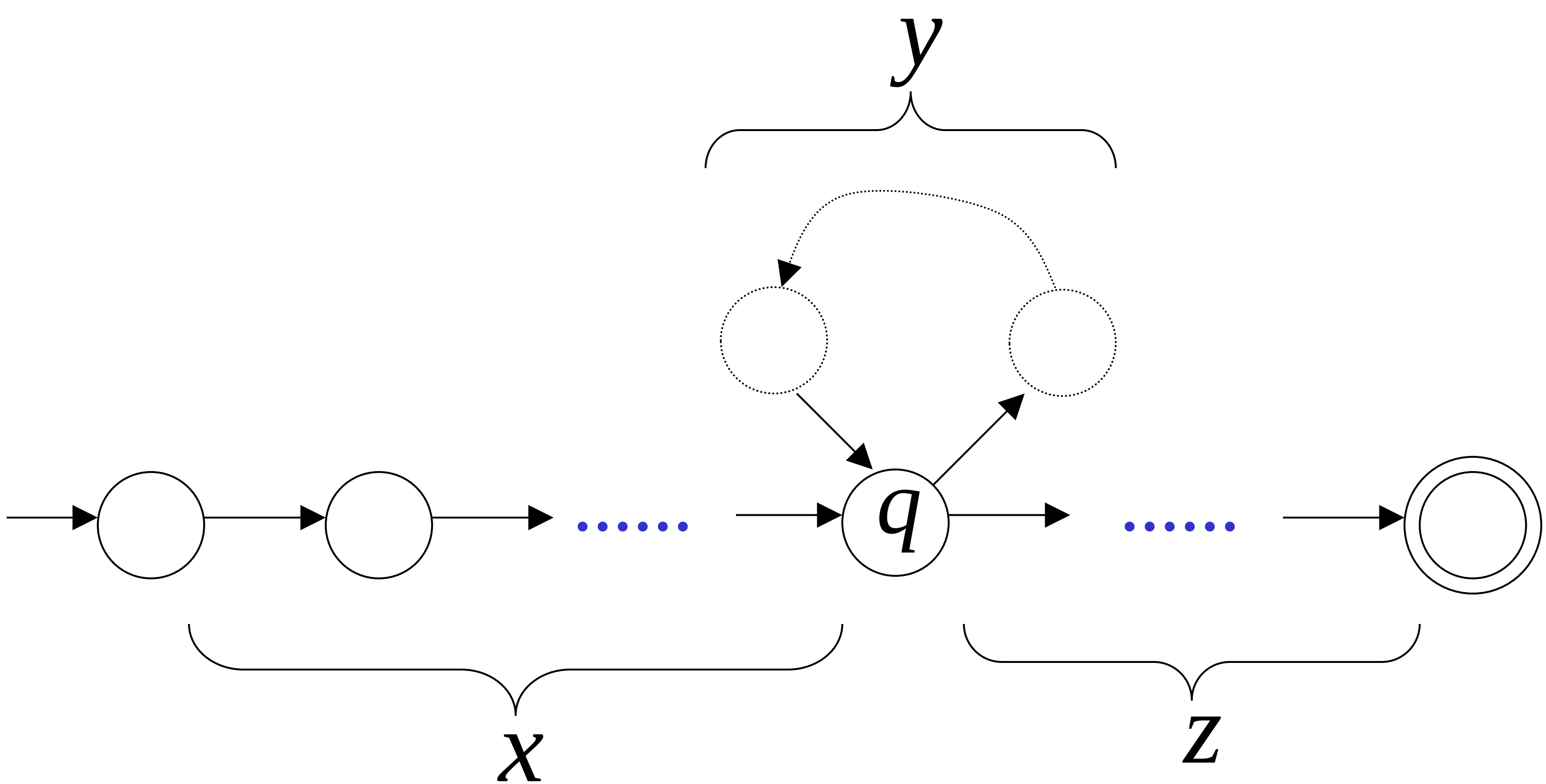
Observation: The string $x\,z$ is accepted



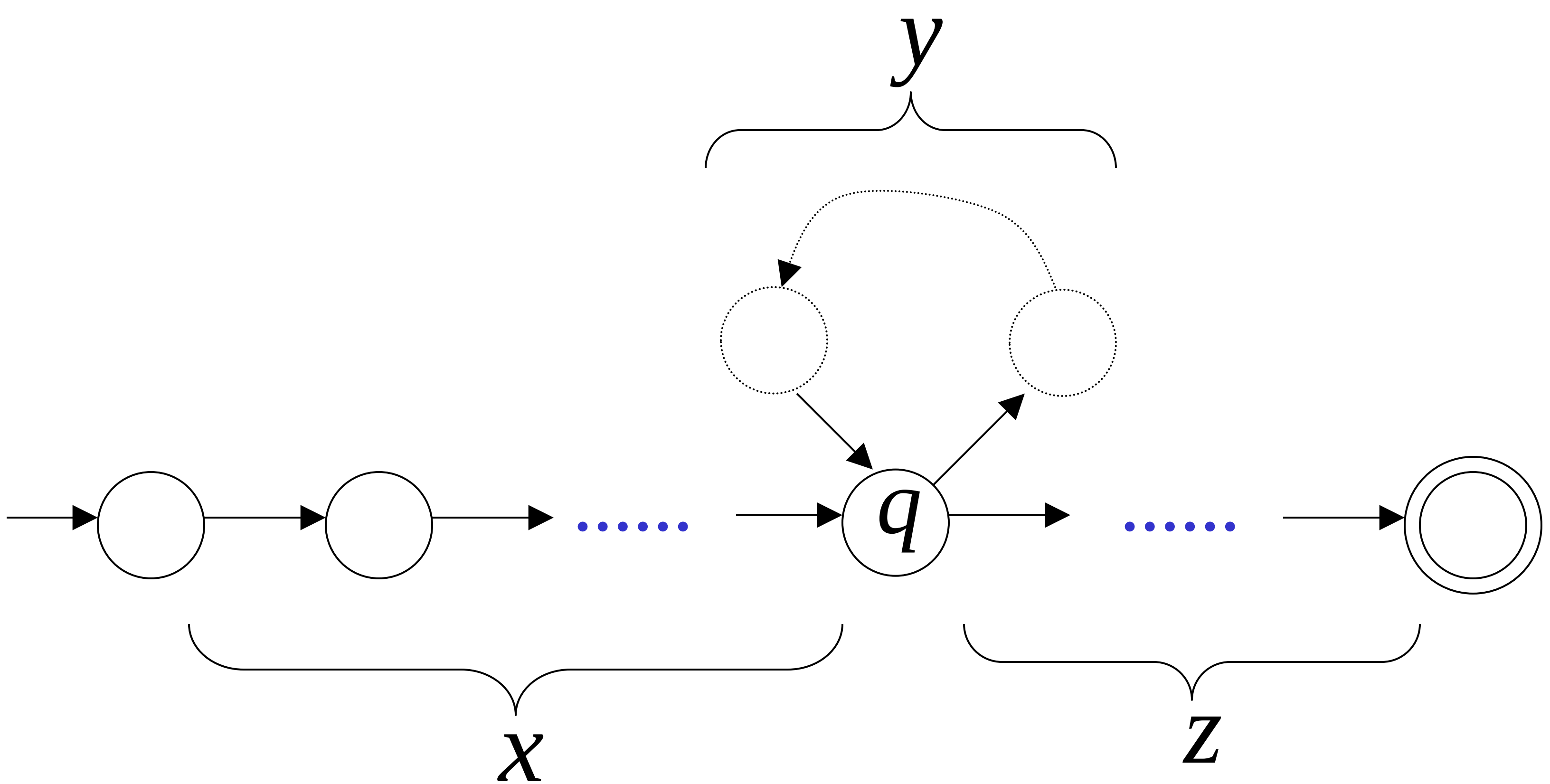
Observation: The string $x\ y\ y\ z$ is accepted



Observation: The string $x\,y\,y\,y\,z$ is accepted

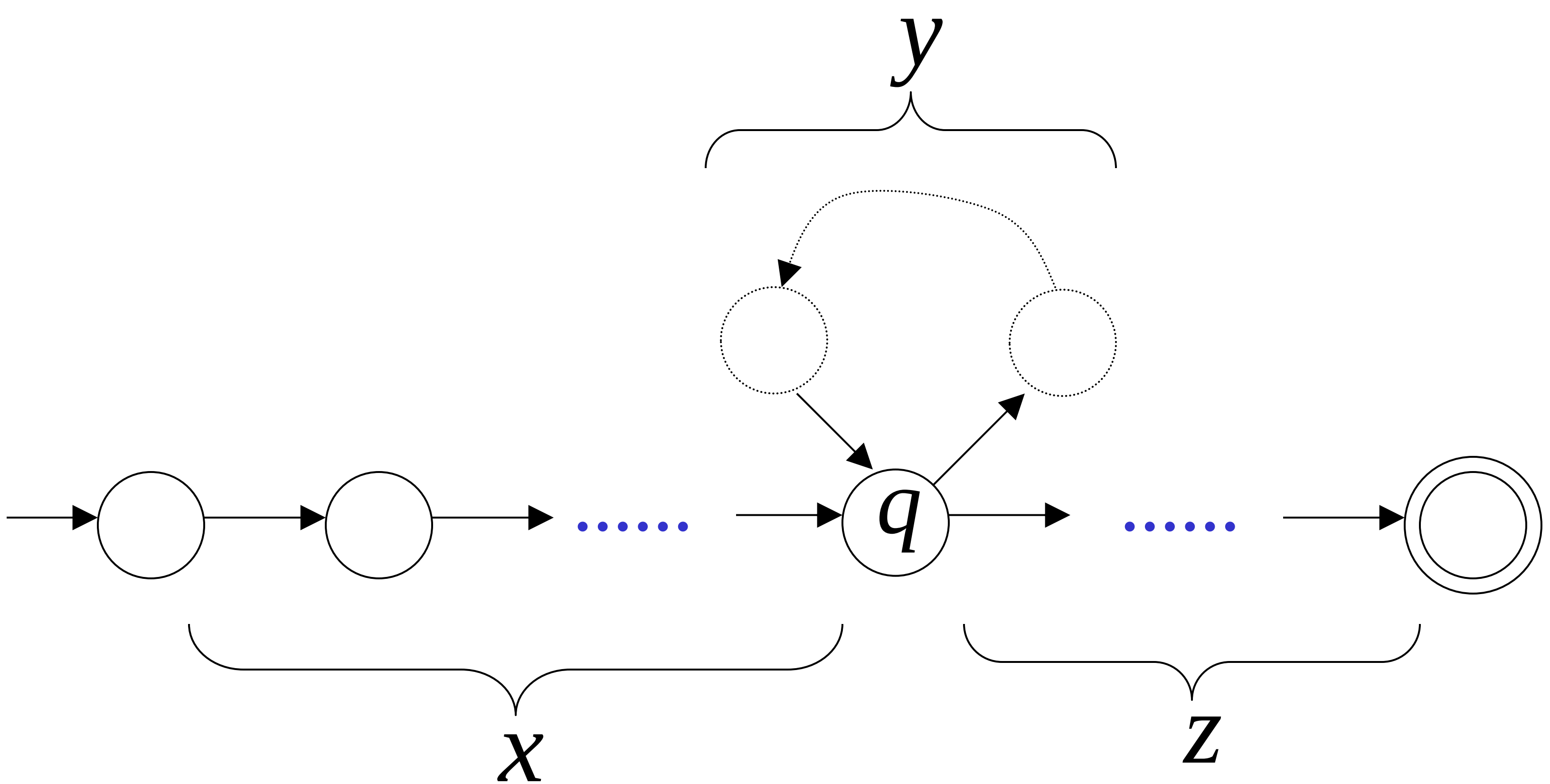


In General: The string $x\,y^i\,z$ is accepted $i = 0, 1, 2, \ldots$



In General: $x\,y^i\,z \in L \qquad i = 0, 1, 2, \ldots$

Language accepted by the DFA



In other words, we described:

The Pumping Lemma !!!

The Pumping Lemma:

- Given a infinite regular language L
- there exists an integer m
- for any string $w \in L$ with length $|w| \geq m$
- we can write $w = x\,y\,z$
- with $|x\,y| \leq m$ and $|y| \geq 1$
- such that: $x\,y^i\,z \in L \qquad i = 0, 1, 2, \ldots$

Applications

of

the Pumping Lemma

Theorem: The language $L = \{a^n b^n : n \geq 0\}$

is not regular

Proof: Use the Pumping Lemma

$$L = \{a^n b^n : n \geq 0\}$$

Assume for contradiction
that L is a regular language

Since L is infinite
we can apply the Pumping Lemma

$$L = \{a^n b^n : n \geq 0\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$

length $|w| \geq m$

We pick $w = a^m b^m$

Write: $a^m b^m = x\, y\, z$

From the Pumping Lemma

it must be that length $|x\, y| \le m, \quad |y| \ge 1$

$$xyz = a^m b^m = \underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...ab...b}_{z}$$

$$\overbrace{a...aa...aa...a}^{m}\overbrace{b...b}^{m}$$

Thus: $y = a^k, \quad k \ge 1$

$$x\ y\ z = a^m b^m \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\ y^i\ z \in L$

$$i = 0, 1, 2, \ldots$$

Thus: $x\ y^2\ z \in L$

$$x\ y\ z = a^m b^m \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\ y^2\ z \ \in\ L$

$$xy^2z = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...a}_{y}a...a}^{m+k}\overbrace{b...b}^{m} \in L$$

where $z = a...ab...b$.

Thus: $a^{m+k}b^m \in L$

$$a^{m+k}b^m \in L \qquad k \geq 1$$

BUT: $L = \{a^n b^n : n \geq 0\}$

$$\Downarrow$$

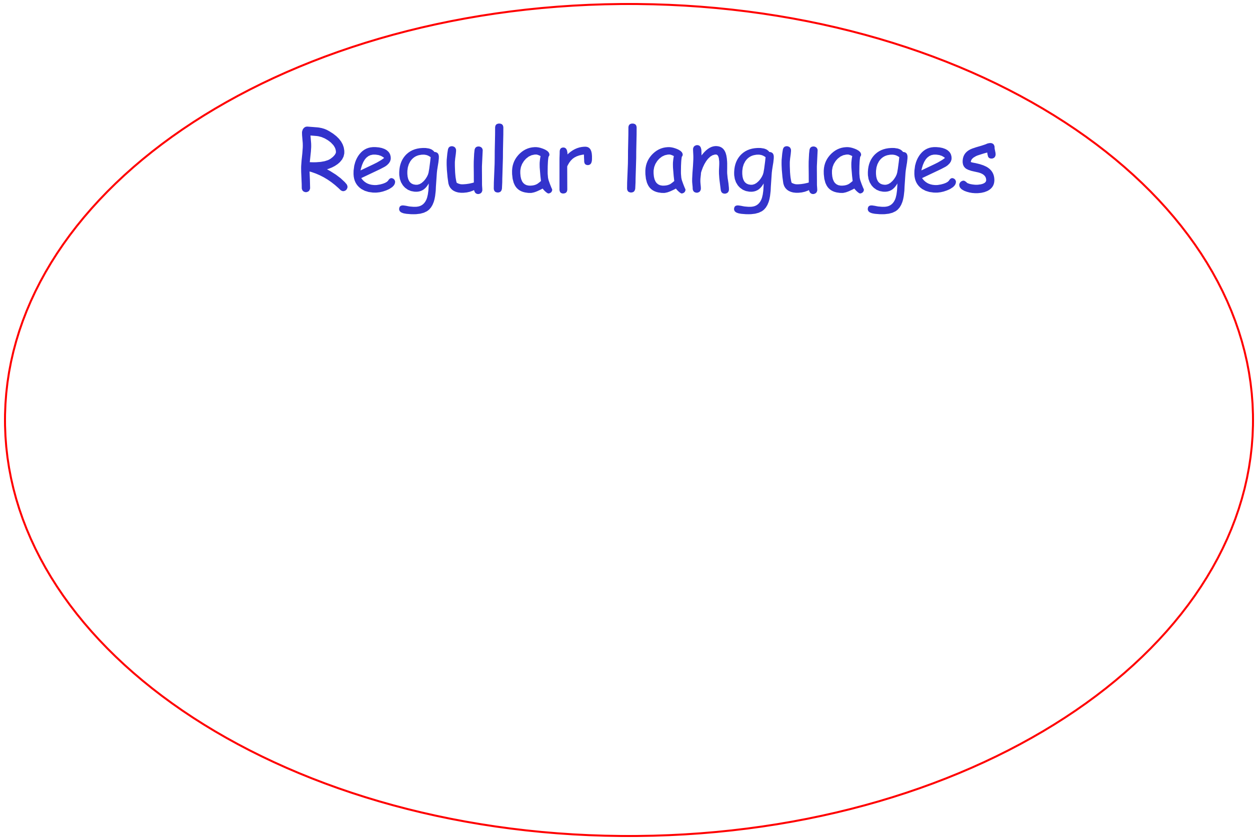
$$a^{m+k}b^m \notin L$$

CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language

Non-regular languages $\{a^n b^n : \quad n \geq 0\}$



Non-regular languages $L = \{vv^R : v \in \Sigma^*\}$

Regular languages

Theorem: The language

$$L = \{vv^R : v \in \Sigma^*\} \qquad \Sigma = \{a, b\}$$

is not regular

Proof: Use the Pumping Lemma

$$L = \{vv^R : v \in \Sigma^*\}$$

Assume for contradiction that L is a regular language

Since L is infinite we can apply the Pumping Lemma

$$L = \{vv^R : v \in \Sigma^*\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$ and

length $|w| \geq m$

We pick $w = a^m b^m b^m a^m$

Write $a^m b^m b^m a^m = x\,y\,z$

From the Pumping Lemma
it must be that length $|\,x\,y\,| \leq m, \quad |\,y\,| \geq 1$

$$xyz = \underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{...ab...bb...ba...a}_{z}$$

$$\overbrace{a...aa...a...a}^{m}\overbrace{b...b}^{m}\overbrace{b...b}^{m}\overbrace{a...a}^{m}$$

Thus: $y = a^k, \quad k \geq 1$

$$x\ y\ z = a^m b^m b^m a^m \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\ y^i\ z \ \in\ L$

$$i = 0, 1, 2, \ldots$$

Thus: $x\ y^2\ z \ \in\ L$

$x\ y\ z = a^m b^m b^m a^m$ $\qquad y = a^k, \quad k \geq 1$

From the Pumping Lemma: $\quad x\ y^2\ z \ \in\ L$

$$xy^2z = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...a}_{y}a...a}^{m+k}\overbrace{b...b}^{m}\overbrace{b...b}^{m}\overbrace{a...a}^{m} \in L$$

(where $z = a...ab...bb...ba...a$)

Thus: $\quad a^{m+k} b^m b^m a^m \ \in L$

$$a^{m+k} b^m b^m a^m \quad \in L \qquad\qquad k \geq 1$$

BUT: $$L = \{vv^R : v \in \Sigma^*\}$$

$$\Downarrow$$

$$a^{m+k} b^m b^m a^m \quad \notin L$$

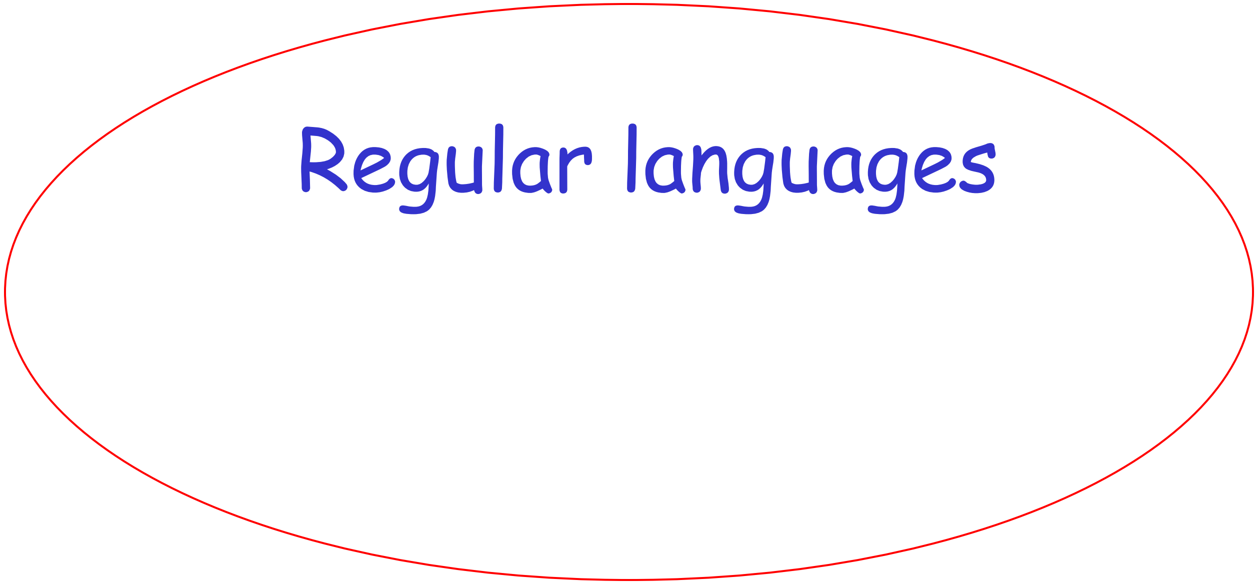
CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language

Non-regular languages

$$L = \{a^n b^l c^{n+l} : \; n, l \geq 0\}$$



Theorem: The language

$$L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$$

is not regular

Proof: Use the Pumping Lemma

$$L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$$

Assume for contradiction
that L is a regular language

Since L is infinite
we can apply the Pumping Lemma

$$L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$ and

length $|w| \geq m$

We pick $w = a^m b^m c^{2m}$

Write $a^m b^m c^{2m} = x\ y\ z$

From the Pumping Lemma
it must be that length $|x\ y| \leq m, \quad |y| \geq 1$

$$xyz = \underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...ab...bc...cc...c}_{z}$$

$$\overbrace{a...aa...aa...a}^{m}\overbrace{b...b}^{m}\overbrace{c...cc...c}^{2m}$$

Thus: $y = a^k, \quad k \geq 1$

$$x\ y\ z = a^m b^m c^{2m} \qquad y = a^k, \quad k \geq 1$$

From the Pumping Lemma: $x\ y^i\ z \in L$

$$i = 0, 1, 2, \ldots$$

Thus: $x\ y^0\ z = xz \in L$

$$x\ y\ z = a^m b^m c^{2m} \qquad y = a^k, \quad k \ge 1$$

From the Pumping Lemma: $xz \in L$

$$xz = \underbrace{\overbrace{a...a}^{}}_{x}\!\!\underbrace{\overbrace{a...a}^{m-k}\overbrace{b...b}^{m}\overbrace{c...cc...c}^{2m}}_{z} \in L$$

Thus: $a^{m-k} b^m c^{2m} \in L$

$$a^{m-k} b^m c^{2m} \in L \qquad k \geq 1$$

BUT: $L = \{a^n b^l c^{n+l} : \ n, l \geq 0\}$

$$\Downarrow$$

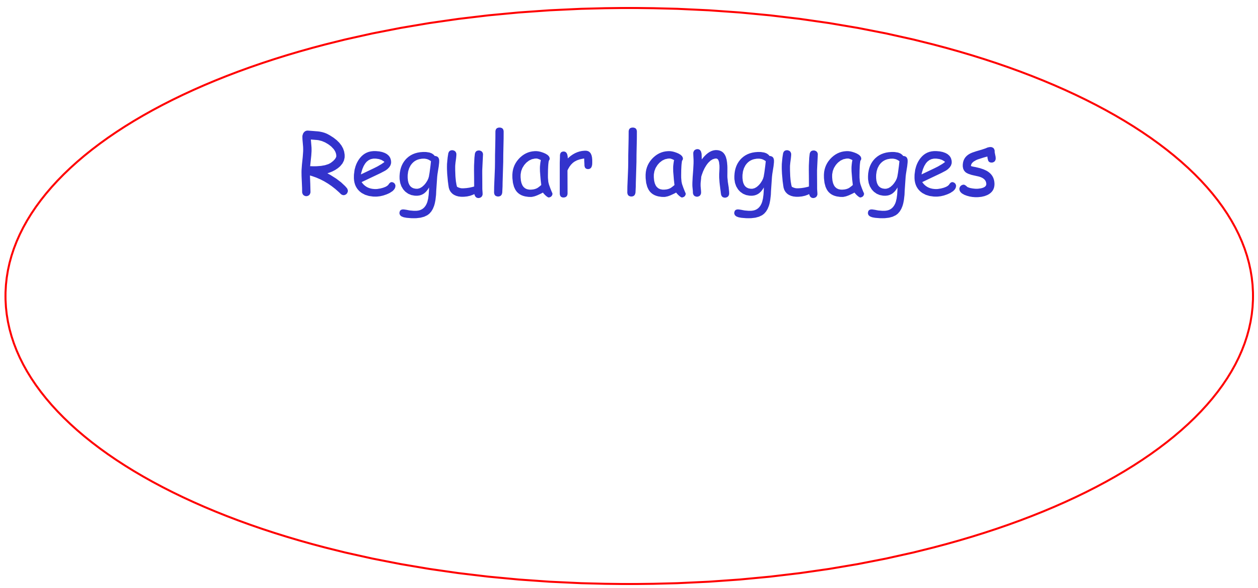
$$a^{m-k} b^m c^{2m} \notin L$$

CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language

Non-regular languages $L = \{a^{n!} : \ n \geq 0\}$



Theorem: The language $L = \{a^{n!} : \ n \geq 0\}$

is not regular

$$n! \ = \ 1 \cdot 2 \Lambda \ (n-1) \cdot n$$

Proof: Use the Pumping Lemma

$$L = \{a^{n!} : n \geq 0\}$$

Assume for contradiction
that L is a regular language

Since L is infinite
we can apply the Pumping Lemma

$$L = \{a^{n!} :\ n \geq 0\}$$

Let m be the integer in the Pumping Lemma

Pick a string w such that: $w \in L$

length $|w| \geq m$

We pick $w = a^{m!}$

Write $a^{m!} = x\ y\ z$

From the Pumping Lemma

it must be that length $|\,x\ y\,| \leq m, \quad |\,y\,| \geq 1$

$$xyz = a^{m!} = \underbrace{\overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}a...a}^{m}\overbrace{a...aa...a}^{m!-m}}_{z \text{ (from the third part)}}$$

$$xyz = a^{m!} = \underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...aa...aa...a}_{z}$$

Thus: $y = a^k, \quad 1 \leq k \leq m$

$$x\ y\ z = a^{m!} \qquad\qquad y = a^k, \quad 1 \le k \le m$$

From the Pumping Lemma: $x\ y^i\ z \in L$

$$i = 0, 1, 2, \ldots$$

Thus: $x\ y^2\ z \in L$

$$x\,y\,z = a^{m!} \qquad y = a^{k},\quad 1 \le k \le m$$

From the Pumping Lemma: $x\,y^2\,z \in L$

$$xy^2z = \overbrace{\underbrace{a...a}_{x}\underbrace{a...a}_{y}\underbrace{a...a}_{y}a...a}^{m+k}\overbrace{a...aa...a}^{m!-m} \in L$$

with z spanning $a...aa...aa...a$.

Thus: $a^{m!+k} \in L$

$$a^{m!+k} \in L \qquad 1 \le k \le m$$

Since: $L = \{a^{n!}: \ n \ge 0\}$

$\Downarrow$

There must exist p such that:

$$m! + k = p!$$

However: $\quad m!+k \leq m!+m \quad$ for $\quad m>1$

$$\leq m!+m!$$

$$< m!m+m!$$

$$= m!(m+1)$$

$$= (m+1)!$$

$$\Downarrow$$

$$m!+k < (m+1)!$$

$$\Downarrow$$

$m!+k \neq p! \quad$ for any $\quad p$

$$a^{m!+k} \in L \qquad 1 \le k \le m$$

BUT: $L = \{a^{n!} : n \ge 0\}$

$$\Downarrow$$

$$a^{m!+k} \notin L$$

CONTRADICTION!!!

Therefore: Our assumption that L is a regular language is not true

Conclusion: L is not a regular language