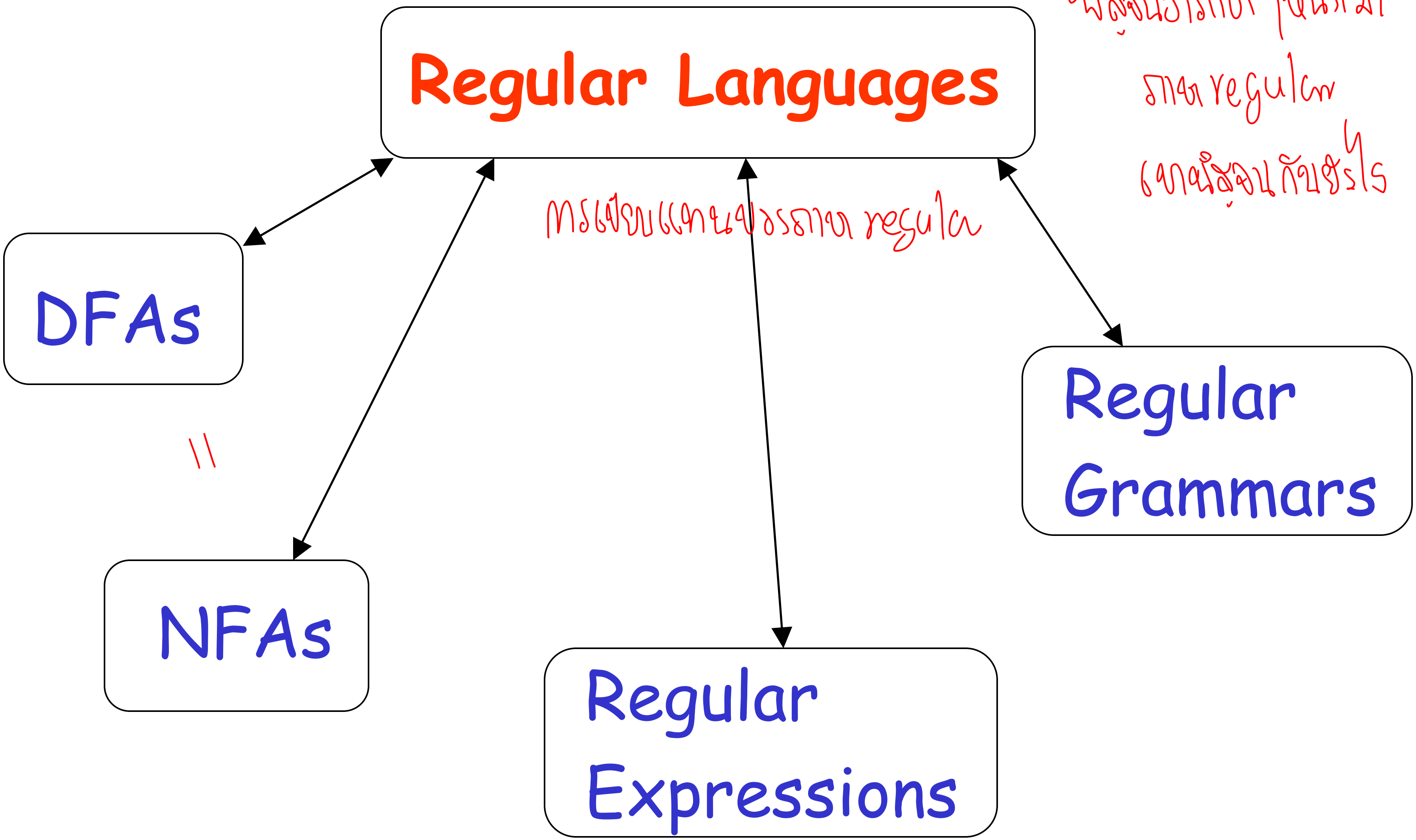
Standard Representations of Regular Languages

จะพิสูจน์ว่าภาษาๆ หนึ่งเป็นไม่ใช่
ภาษา regular
เราพิสูจน์กับอะไร



When we say: We are given a Regular Language L

We mean: Language L is in a standard representation

Elementary Questions

about

Regular Languages

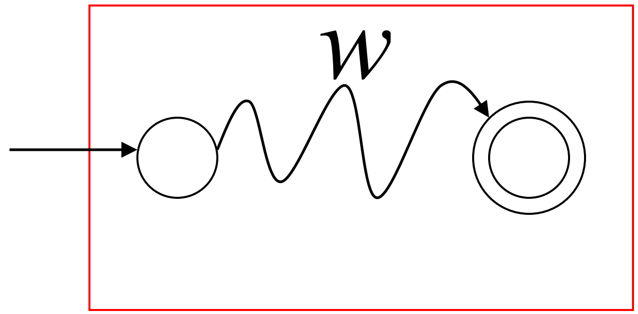
คำถามพื้นฐานเกี่ยวกับ regular language

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 ?

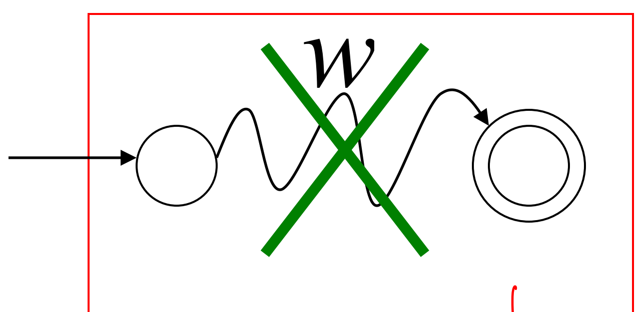
DFA



$w \in L$

ถ้าเป็น

DFA



ไม่เป็นสมาชิก

$w \notin L$

เข้าได้หรือ อ่านไม่หมด

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

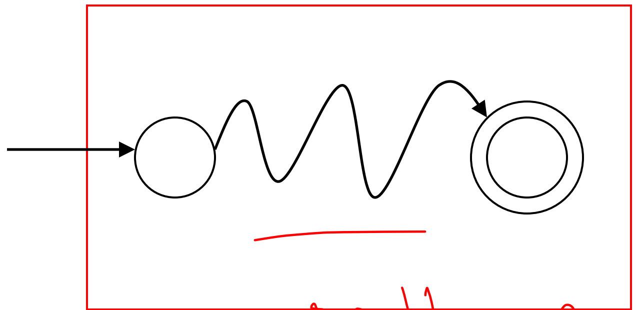
เซตที่ไม่มีภาษาเป็นนาง

Answer: Take the DFA that accepts L

ถ้าไม่มีทางไปสู่ทางเดินได้

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

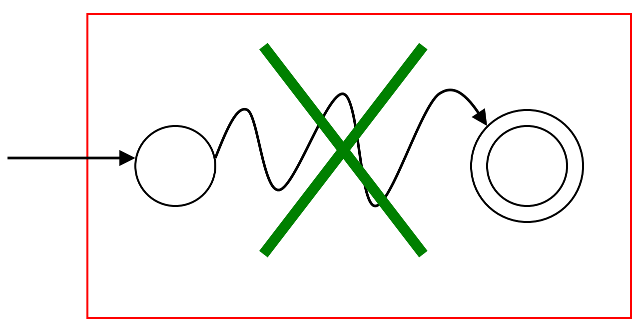
DFA



$$L \neq \emptyset$$

มี path → ต้องมี string 1 ตัว

DFA



$$L = \emptyset$$

Question: Given regular language L

how can we check

if L is finite?

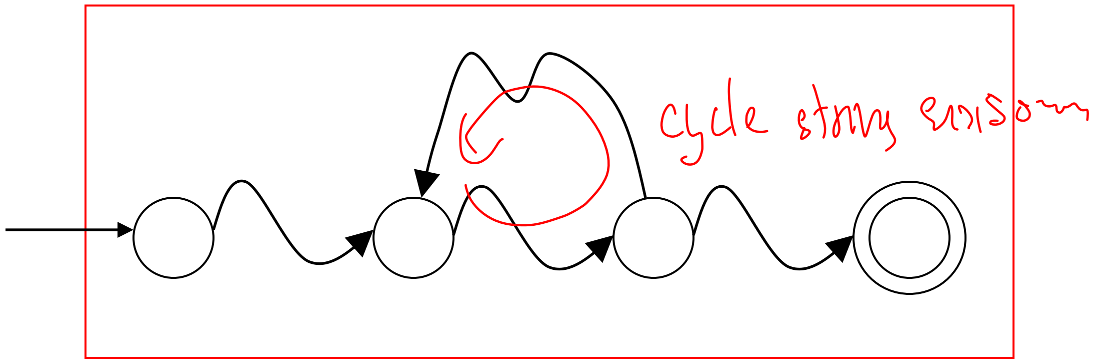
มี walk ที่เป็น cycle ไหม

Answer: Take the DFA that accepts L

Check if there is a walk with cycle

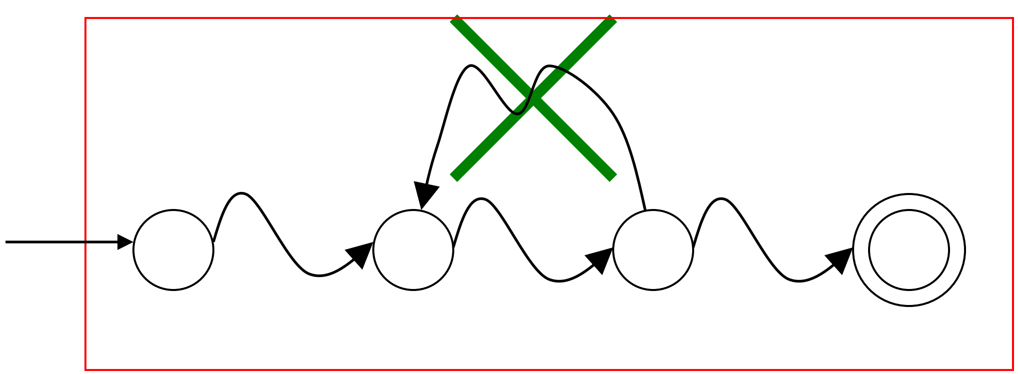
from the initial state to a final state

DFA



L is infinite

DFA



L is finite

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

๒ ภาษาเลย ซ้อนทับกันสนิท

เป๊ะ

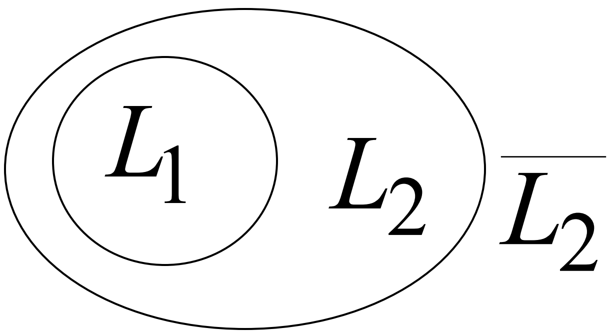
$L_1 L_2$

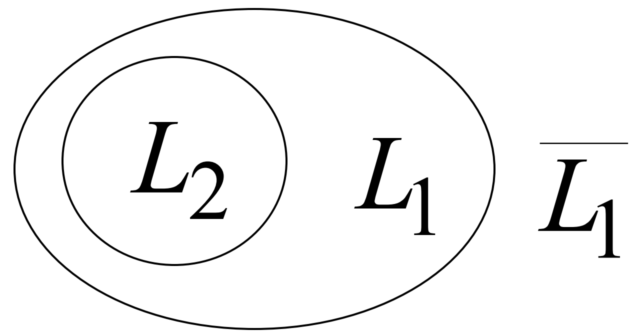
ทันที

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

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

$L_1 \cap \overline{L_2} = \varnothing$ **and** $\overline{L_1} \cap L_2 = \varnothing$





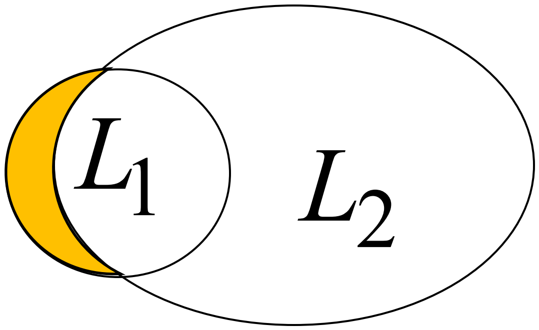
$L_1 \subseteq L_2$ $\qquad$ $L_2 \subseteq L_1$

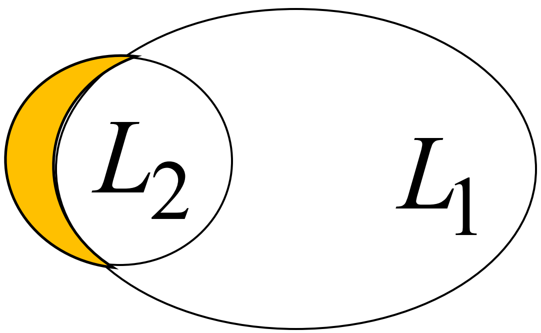
$$L_1 = L_2$$

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

$\Downarrow$

$L_1 \cap \overline{L_2} \neq \varnothing$ or $\overline{L_1} \cap L_2 \neq \varnothing$





$L_1 \not\subset L_2$ $\quad$ $L_2 \not\subset L_1$

$\Downarrow$

$$L_1 \neq L_2$$

Non-regular languages

Non-regular languages

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

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

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

ก็แค่สร้างไม่ได้เองหล่ะ?

Problem: this is not easy to prove

หา DFA ไม่ได้ง่ายๆนะ

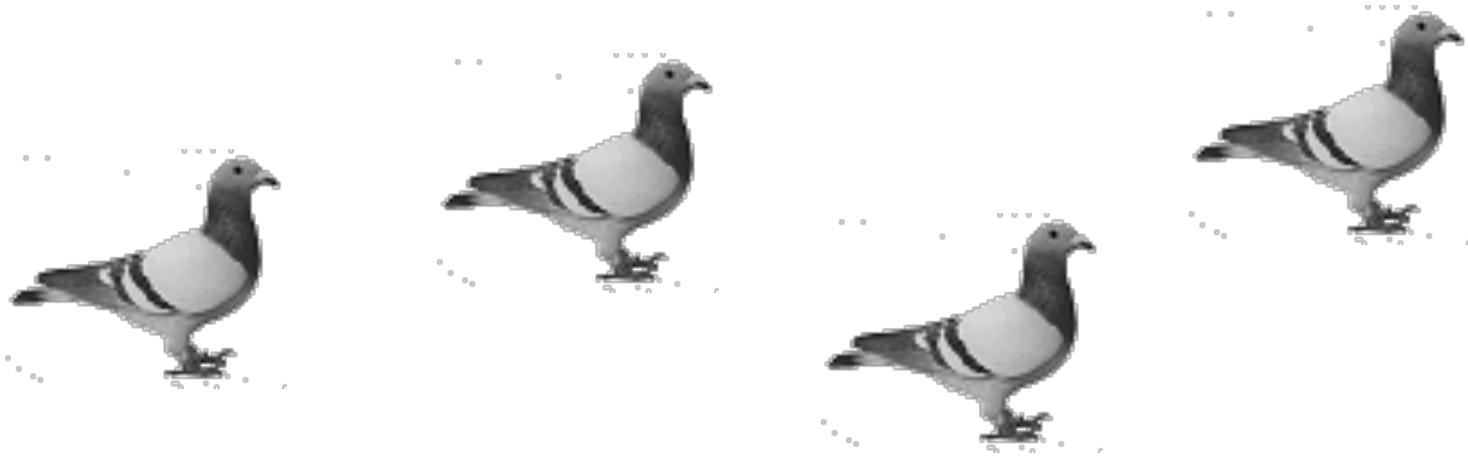
Solution: the Pumping Lemma !!!

การพิสูจน์ว่า L is not regular

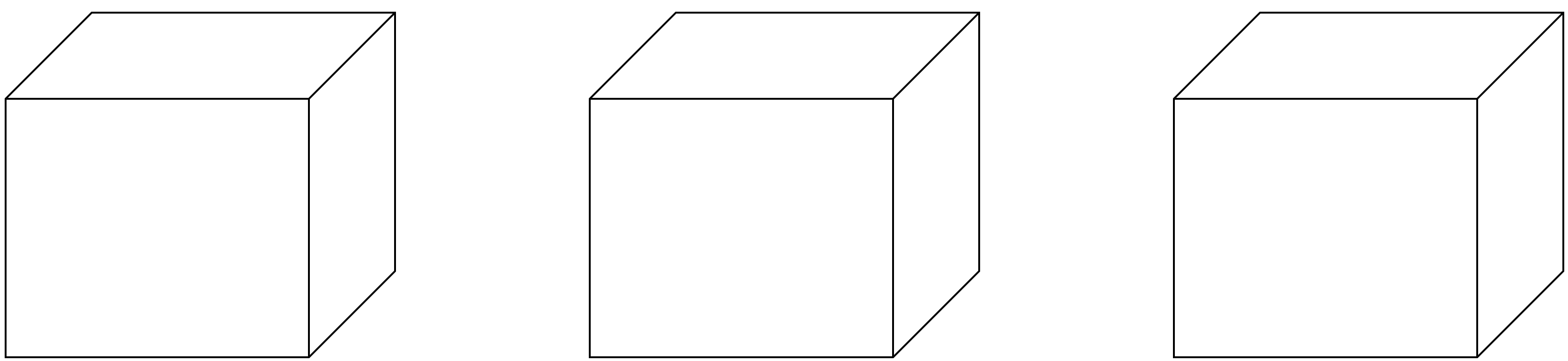
ซ้ายที่สุด → ใบตริงที่สุด

The Pigeonhole Principle

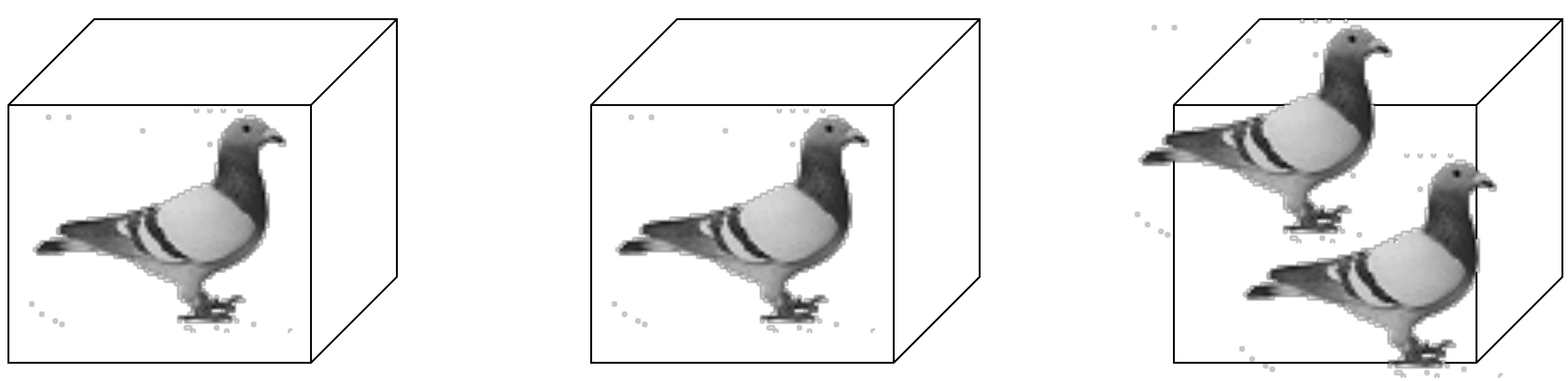
4 pigeons



3 pigeonholes

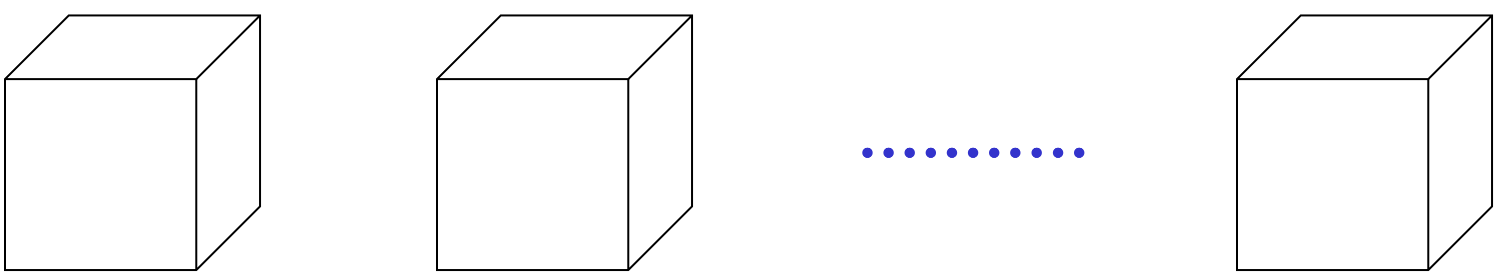


A pigeonhole must contain at least two pigeons



n pigeons

m pigeonholes $\qquad n > m$



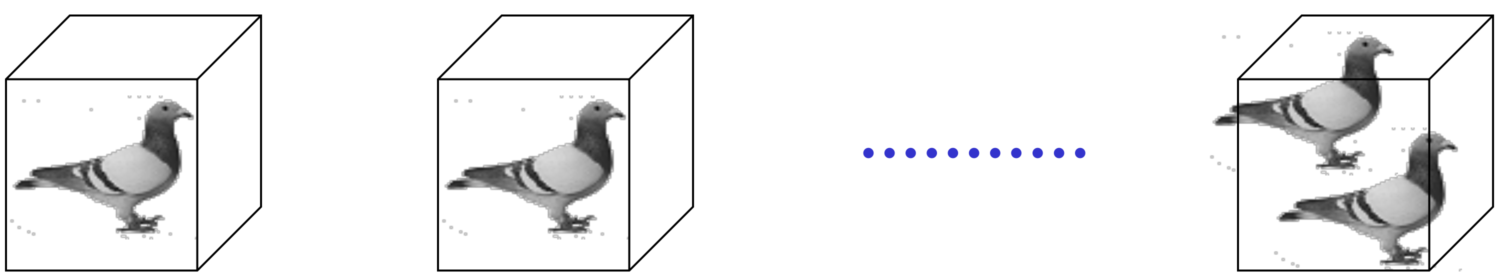
The Pigeonhole Principle

n pigeons

m pigeonholes

$$n > m$$

There is a pigeonhole with at least 2 pigeons

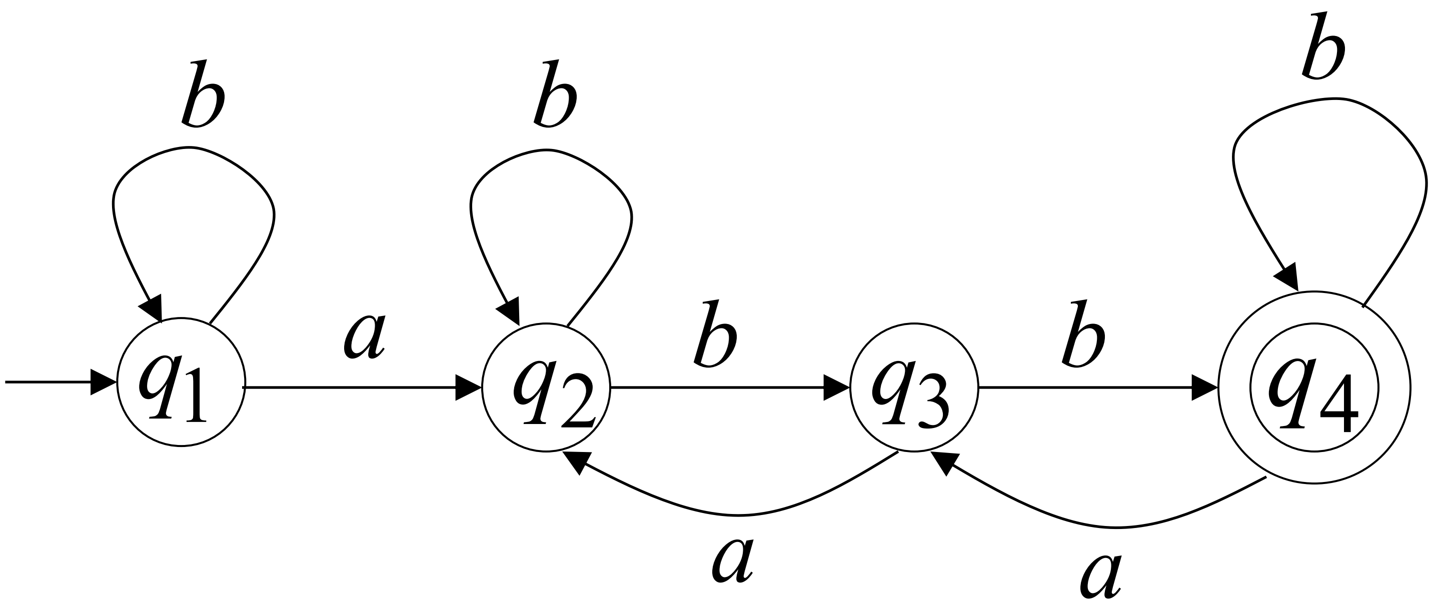


The Pigeonhole Principle

and

DFAs

DFA with 4 states

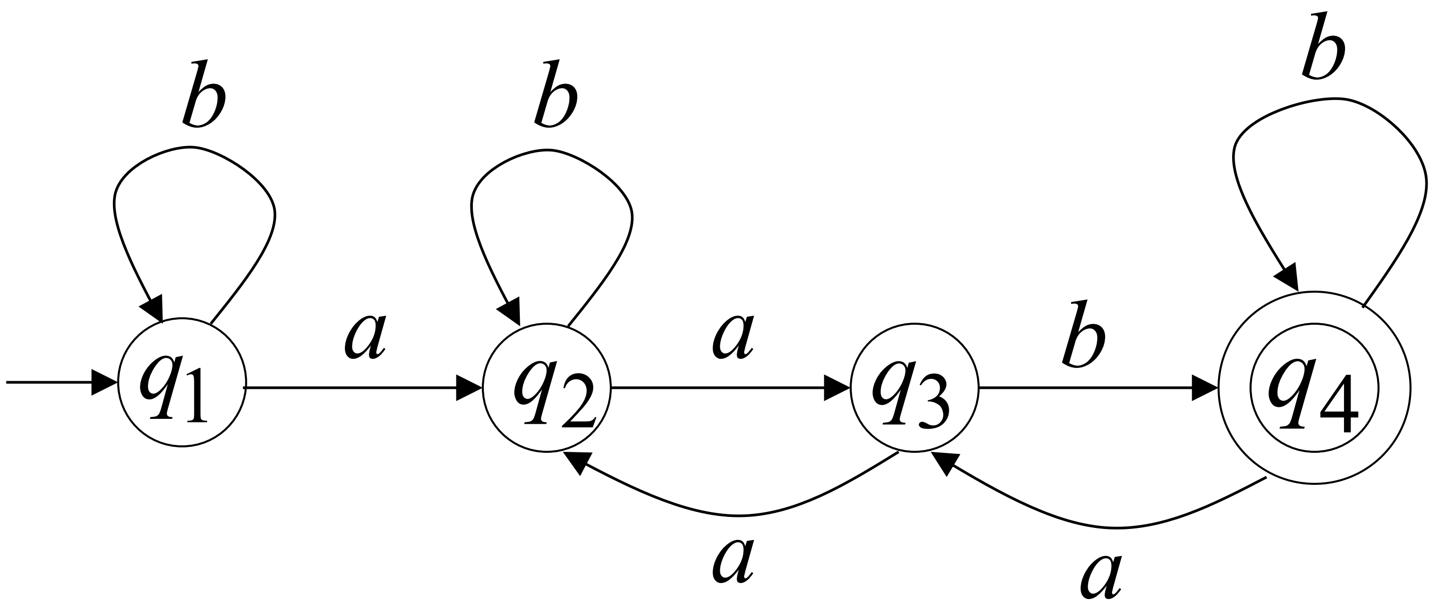


In walks of strings: a

aa

aab

no state is repeated



In walks of strings: $aabb$

$bbaa$

$abbabb$

$abbbabbabb...$

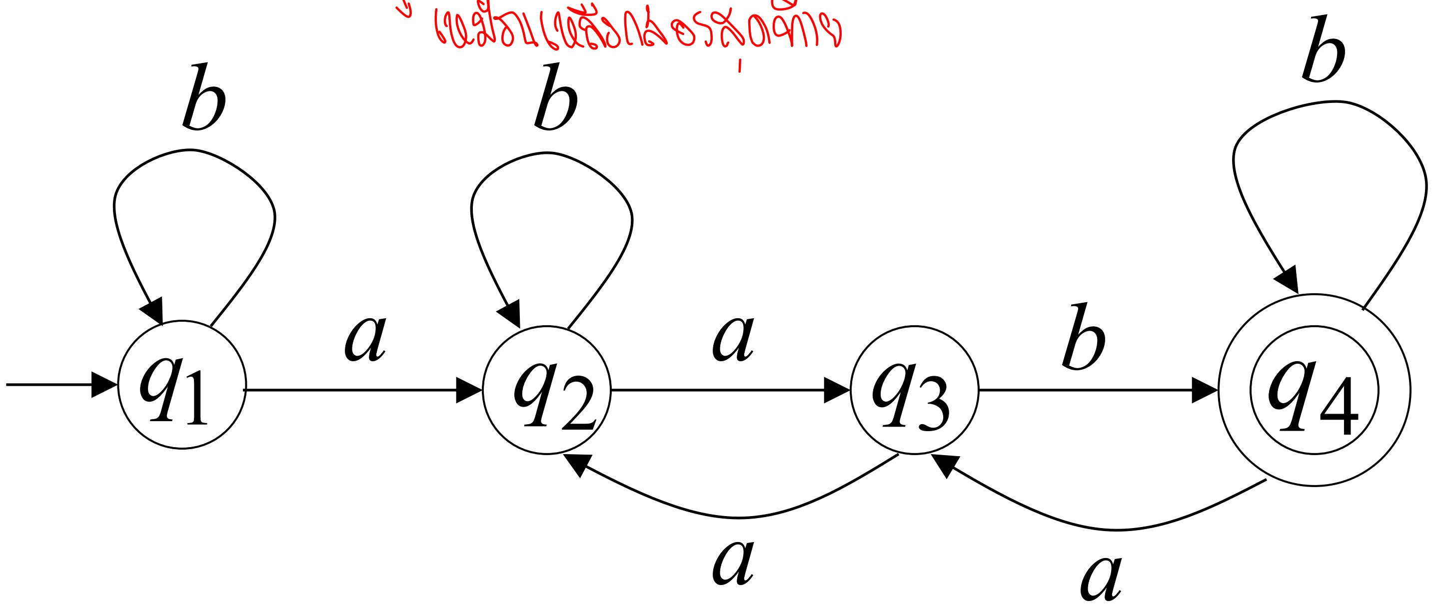
a state is repeated

ถ้าความยาวของstringมีเยอะเกิน

↓

stateต้องวนซ้ำ

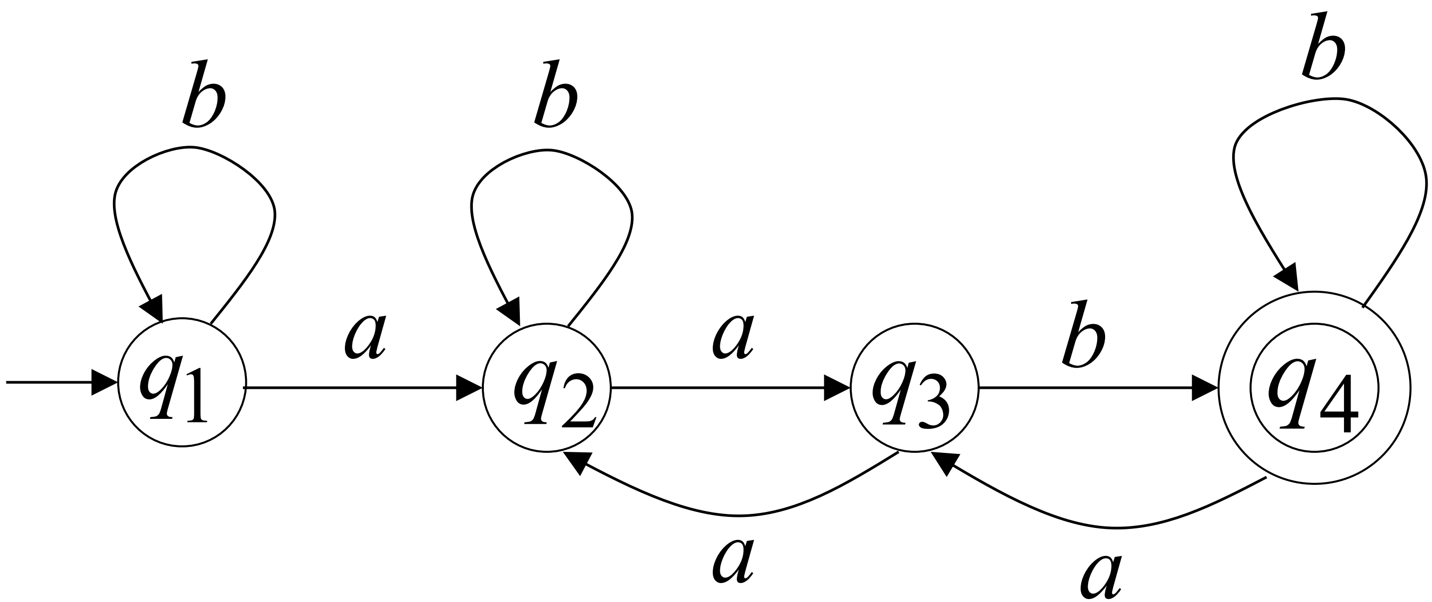
↓ เหมือนเหลือกลอรสุดท้าย



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

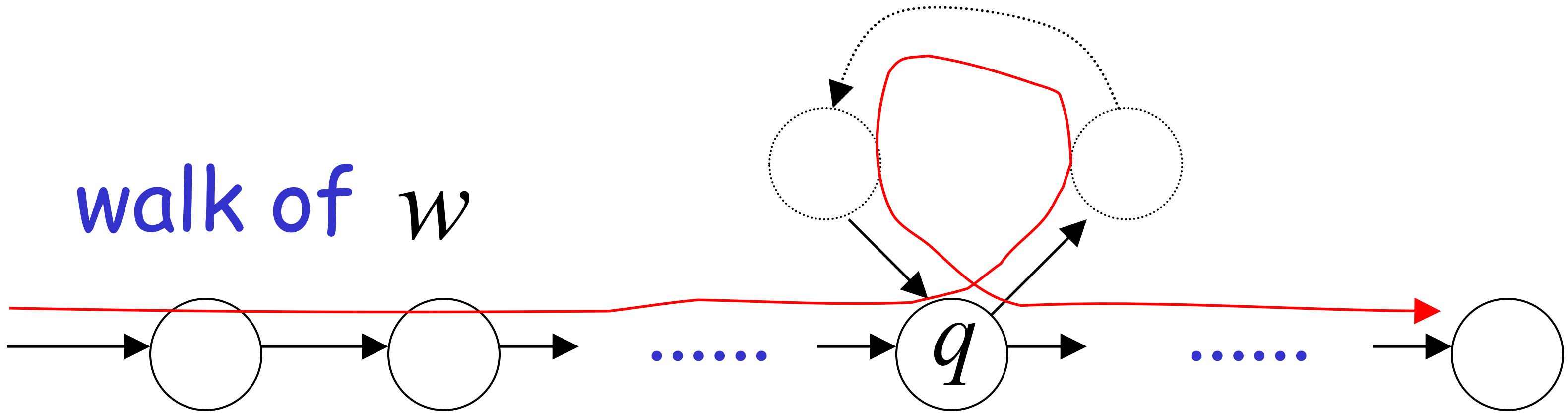


In general, for any DFA:

String w has length $\geq$ number of states

A state q must be repeated in the walk of w

walk of w

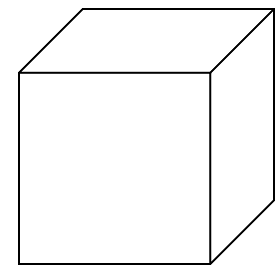


Repeated state

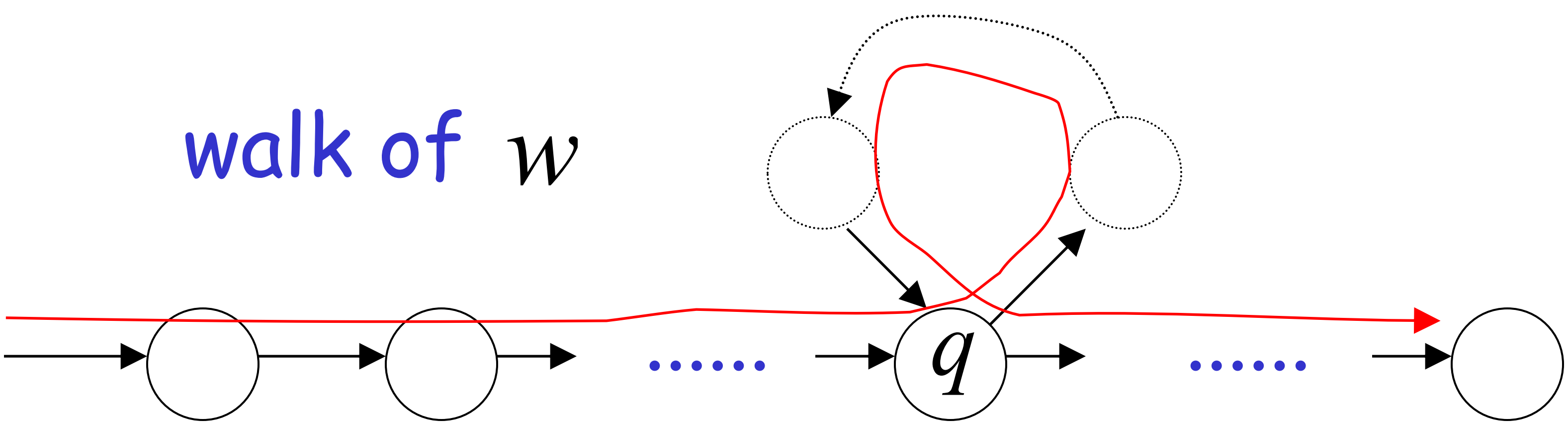
In other words for a string w:

$\xrightarrow{a}$ transitions are pigeons

q states are pigeonholes



walk of w



Repeated state

เชิงเชื่อมหากัน

The Pumping Lemma

การทำซ้ำ เอาคุณไปปั๊ม

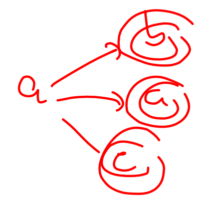
ทฤษฎี

Take an infinite regular language L

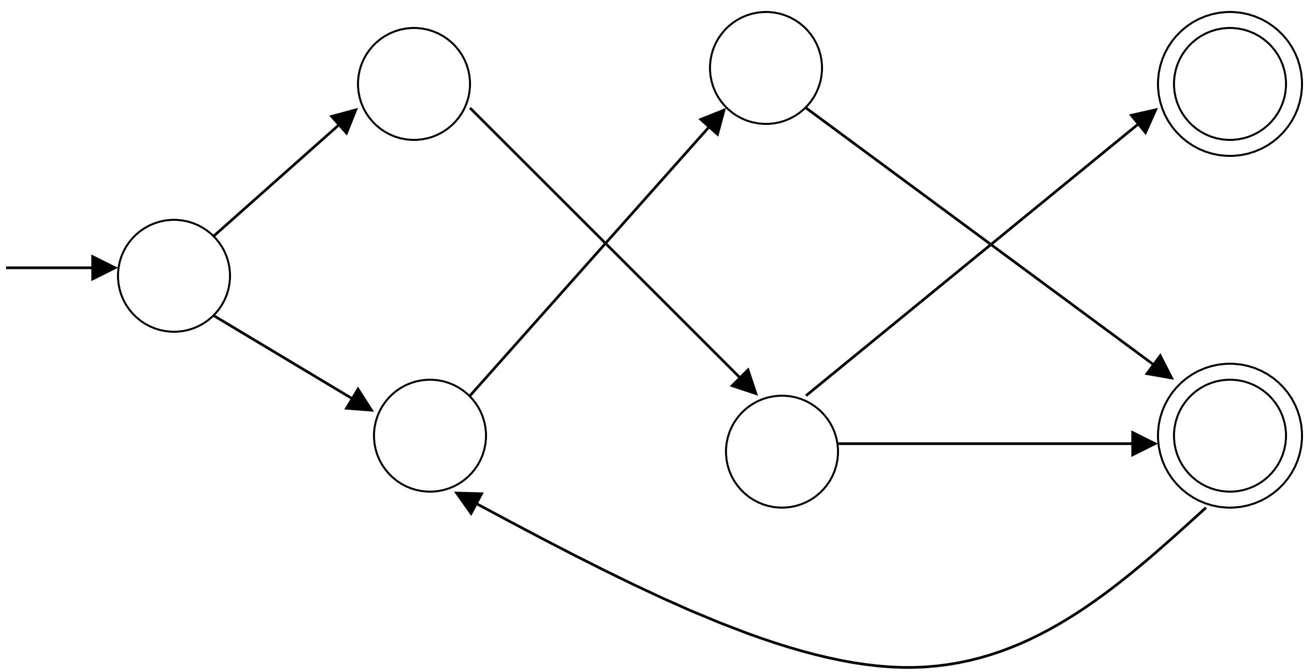
finite เป็น regular อยู่แล้ว

เขียนเป็นได้เสมอ

aa ac ab



There exists a DFA that accepts L



จำนวนสเตทใน DFA

m
states

Take string w with $w \in L$

There is a walk with label w:

walk w

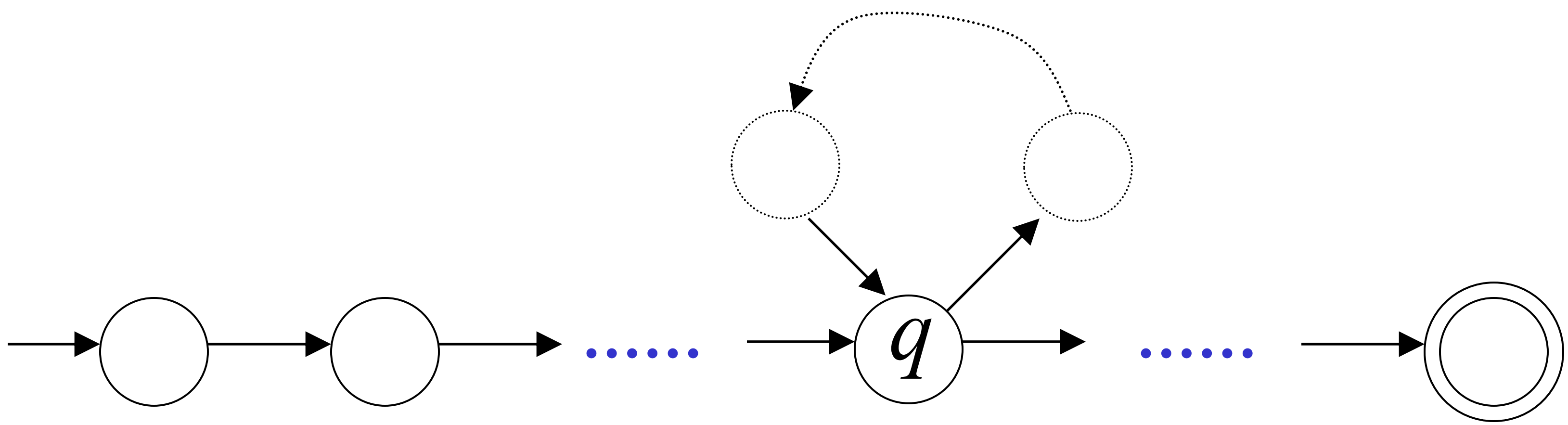
If string w has length $|w| \geq m$ (number of states of DFA)

ความยาวของสตริง ≥ จำนวนสเตต

จะมี state บาง state ที่ถูกซ้ำ

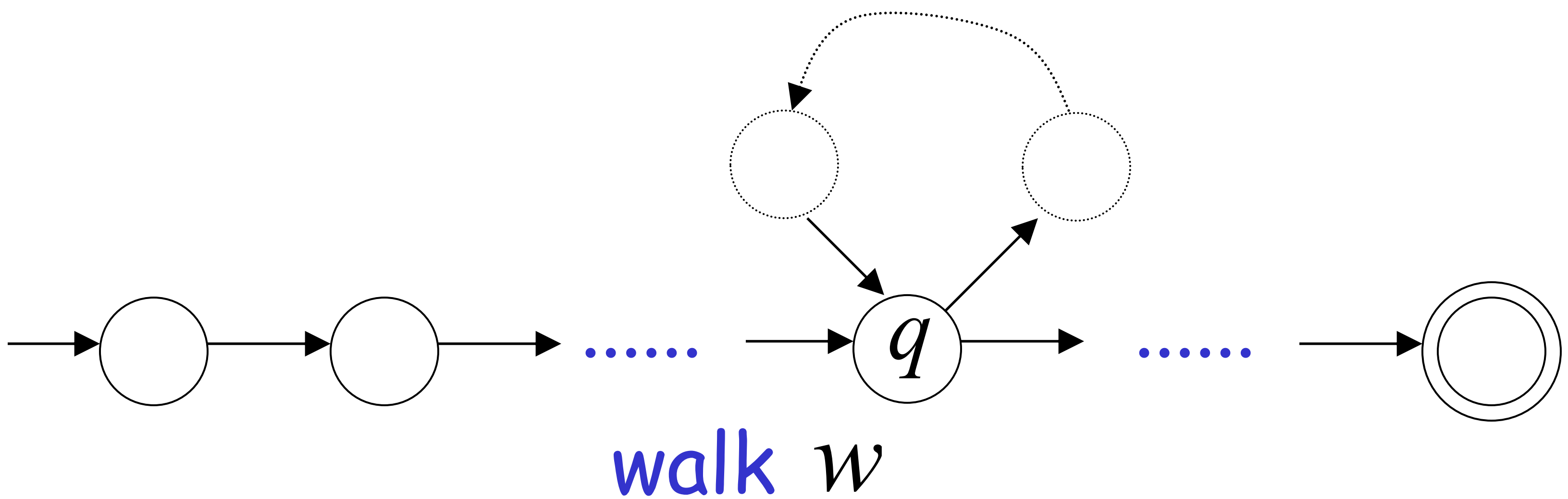
then, from the pigeonhole principle:

a state is repeated in the walk w



walk w

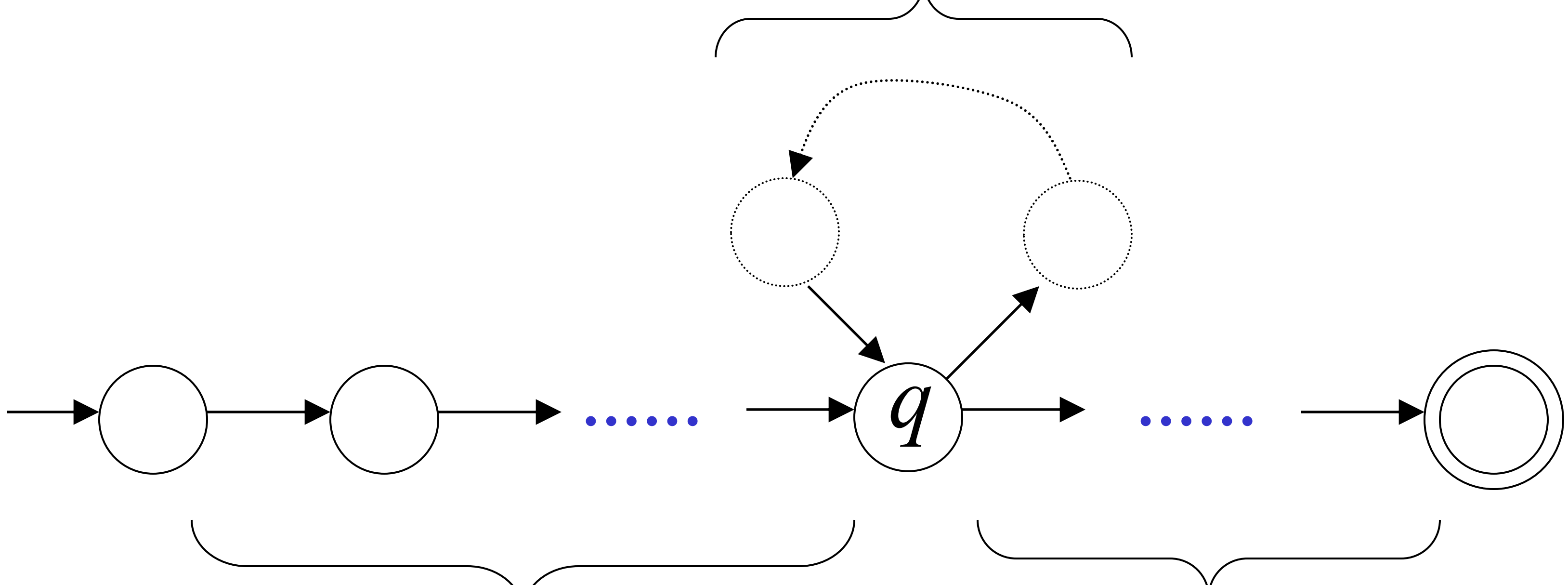
Let q be the first state repeated in the walk of w



แบ่ง 3 ก้อน

Write $w = x\,y\,z$

y string ซ้ำๆ



x ตั้งแต่ initial ไปถึง state q

z ส่วนของเหลือจาก q → final

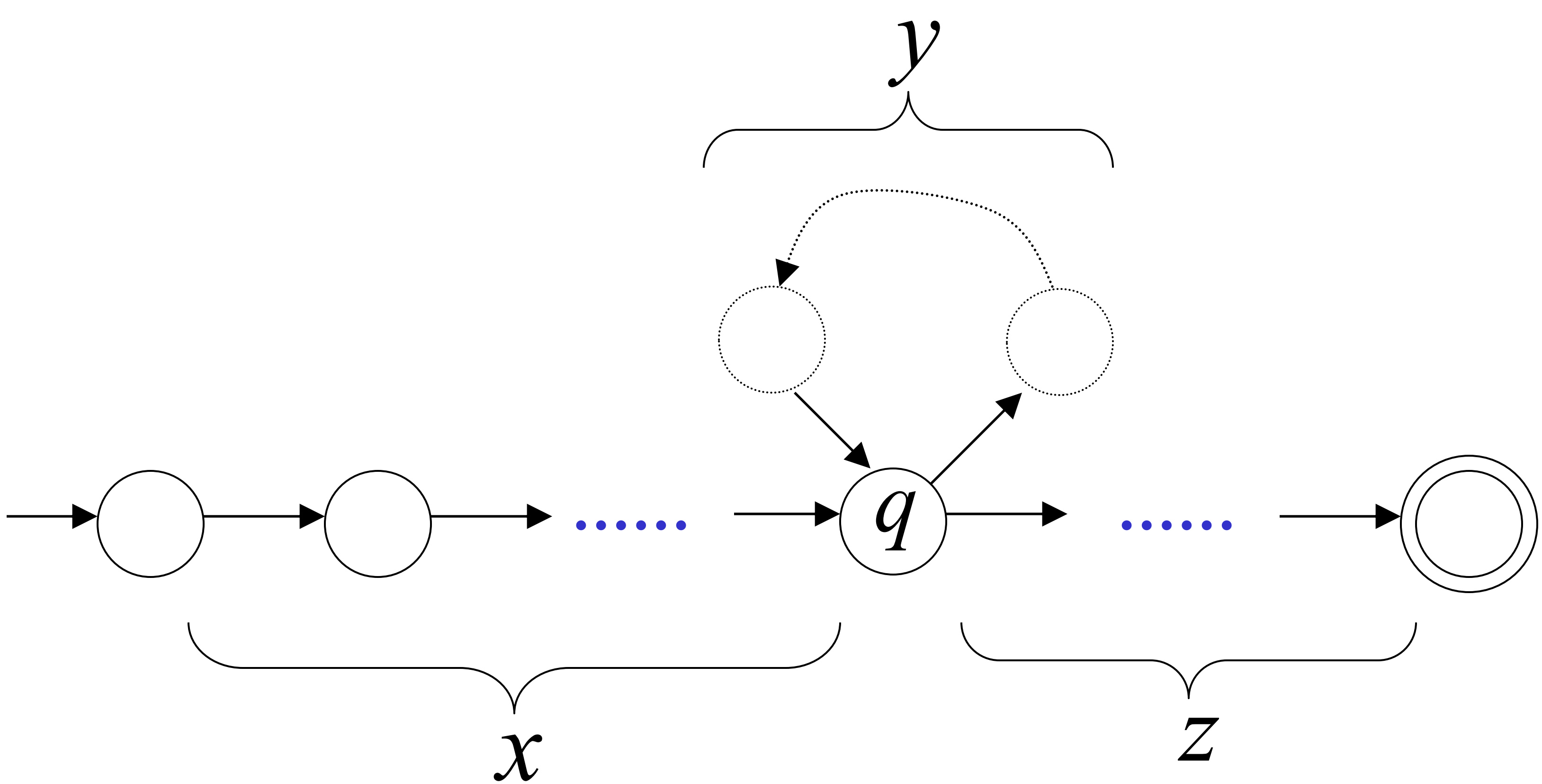
Observations: ① length $|x\,y| \leq m$ number of states of DFA

เงื่อนไข

จำนวนstateที่ใช้จริงไม่ถึง z

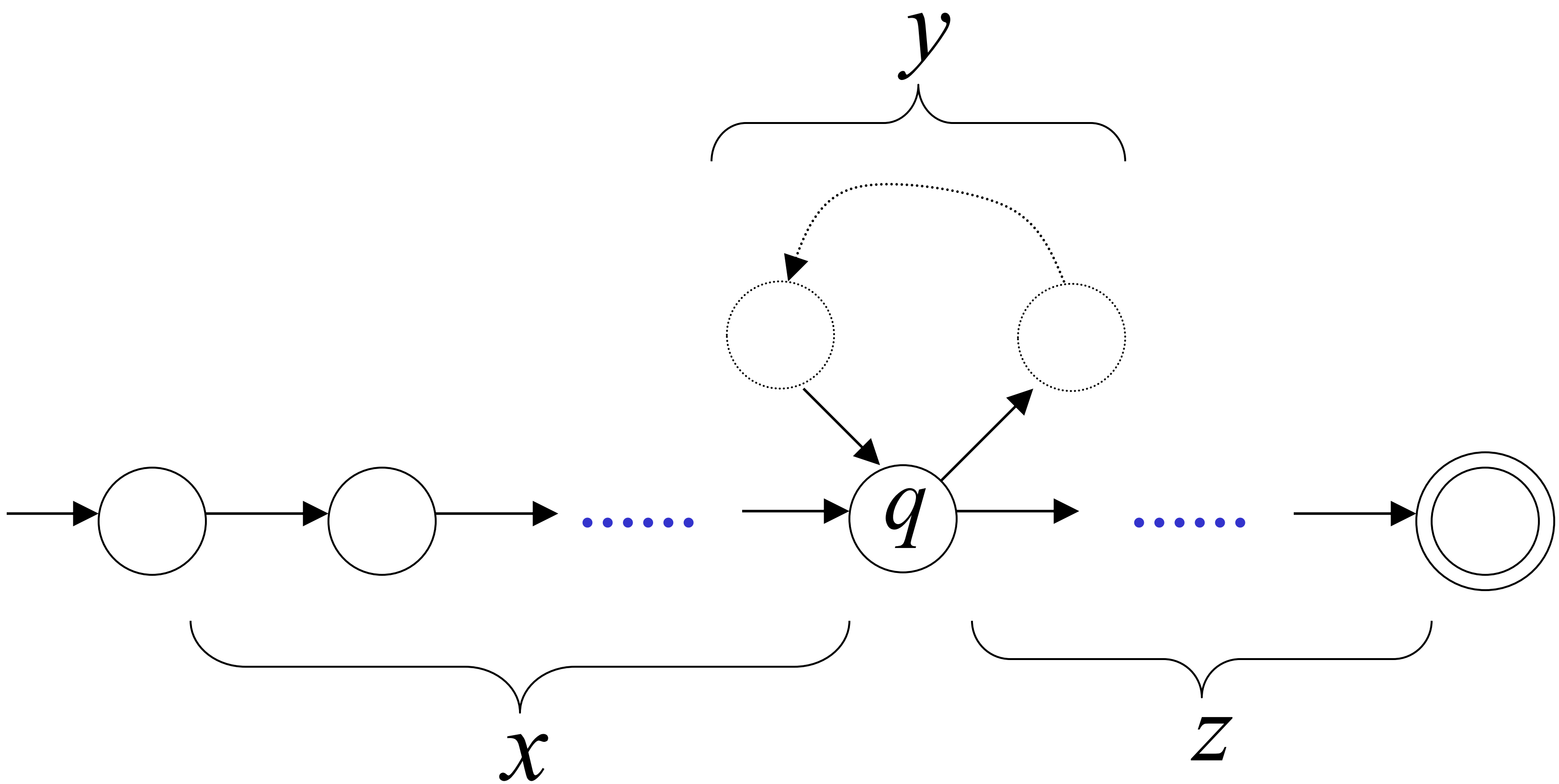
② length $|y| \geq 1$

สามารถวนซ้ำได้



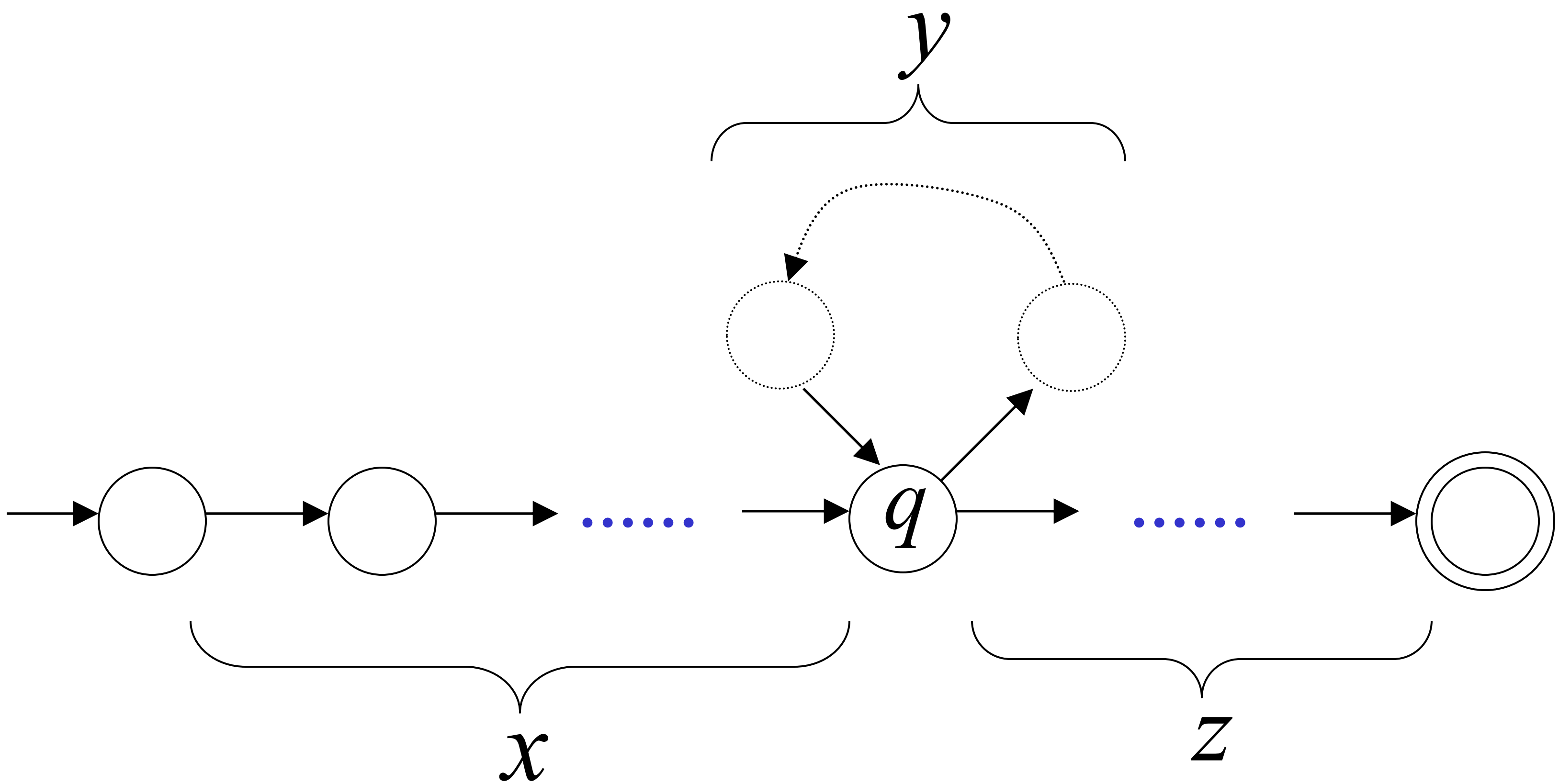
Observation: 3 The string $x\,z$
is accepted

กรณีที่ไม่ซ้ำไร

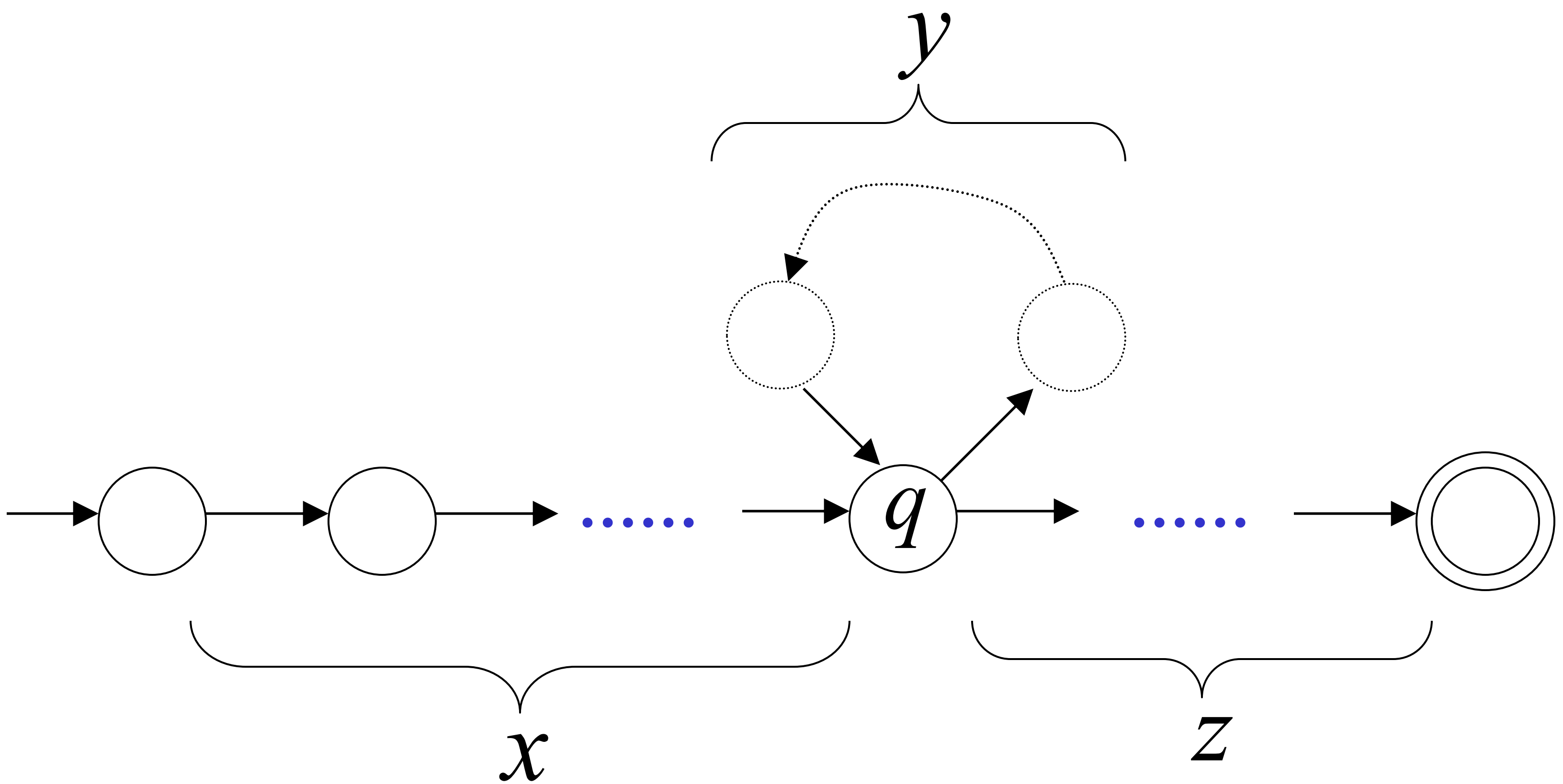


Observation: The string $x\,y\,y\,z$ is accepted

ส่วนซ้ำแค่ไหนก็ได้



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

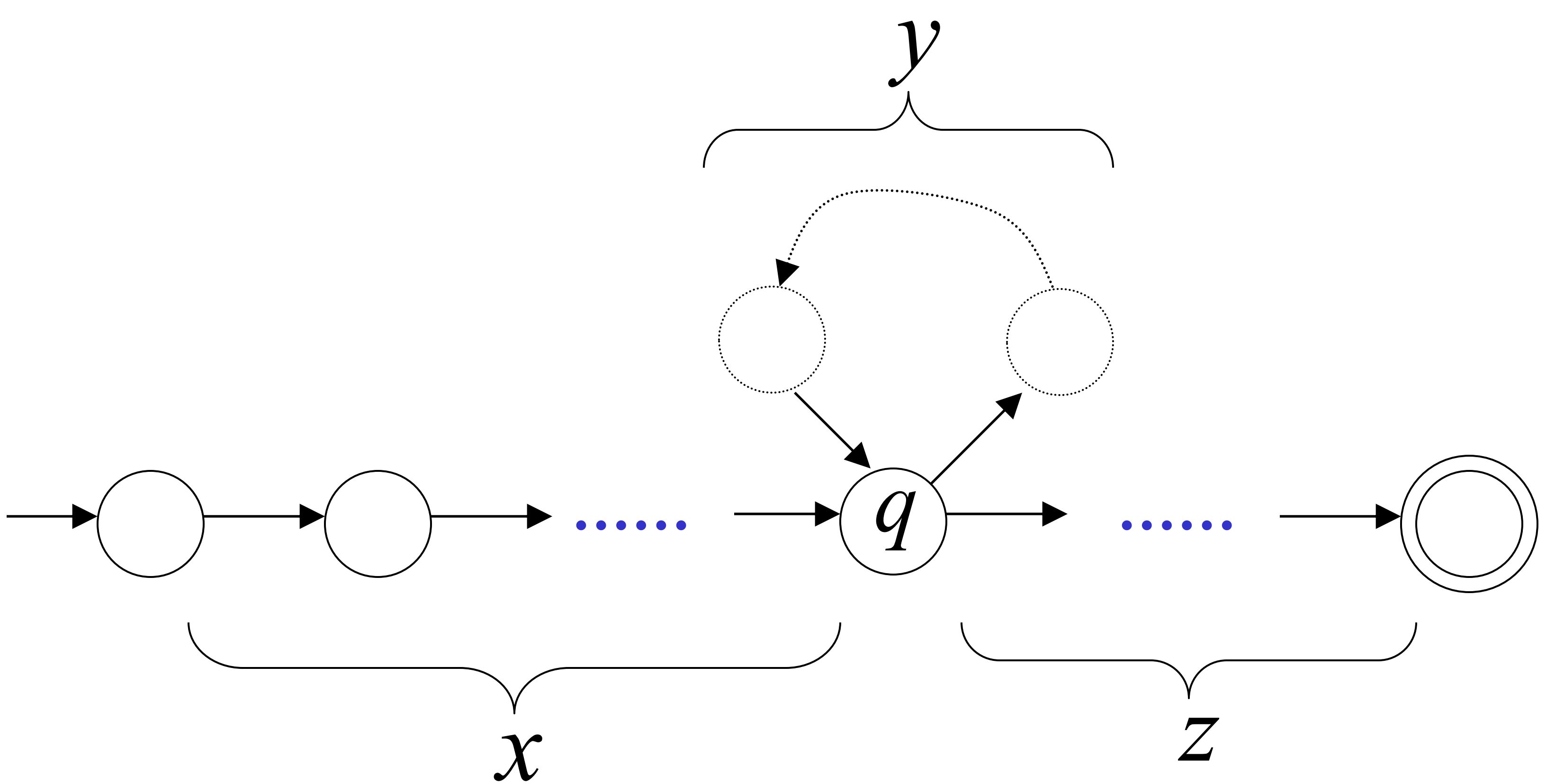


In General:

The string $x\ y^i\ z$ pump ก็ยอมรับได้

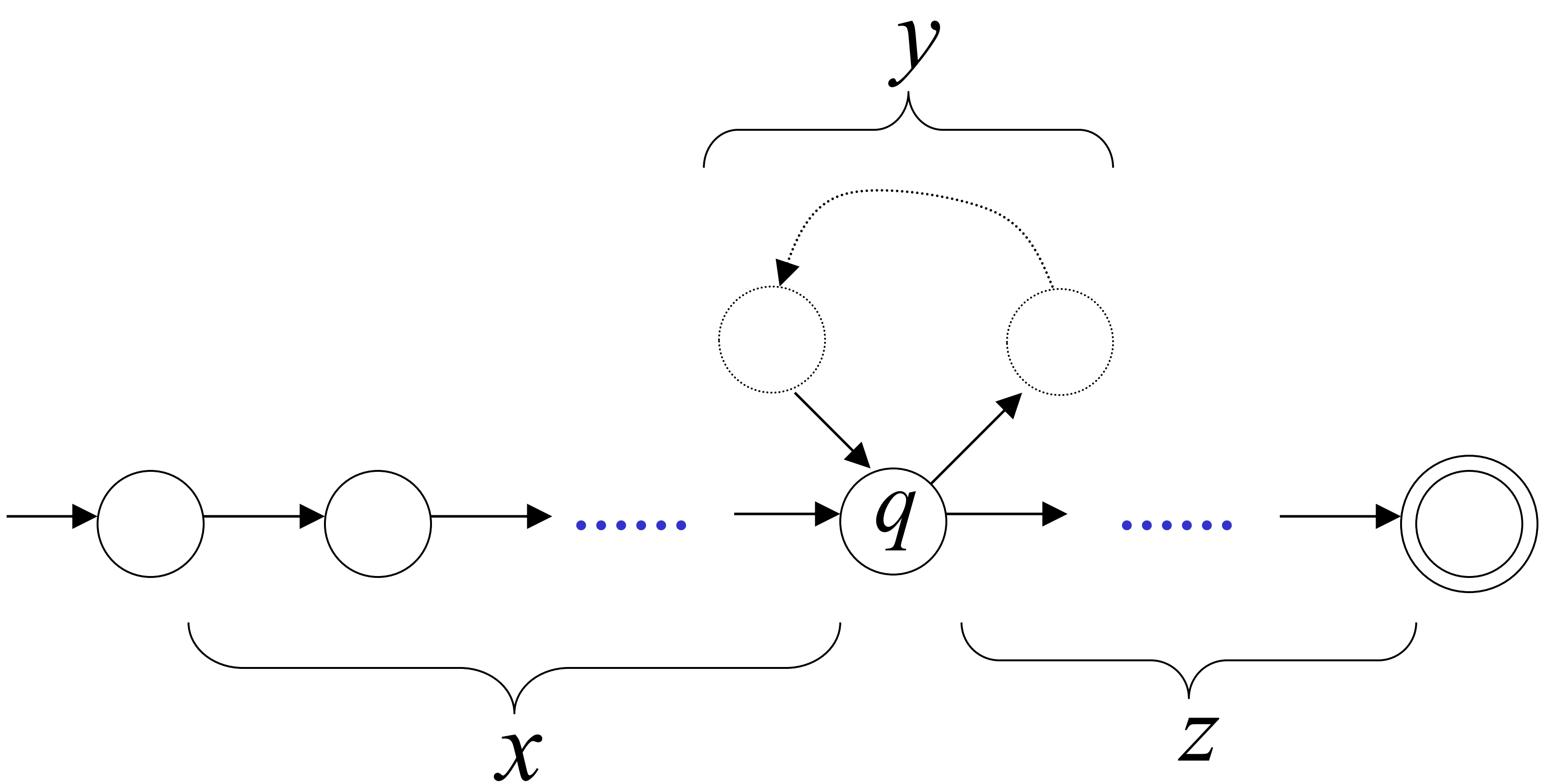
pumpไป ได้ตลอด

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



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

Language accepted by the DFA



In other words, we described:

The Pumping Lemma !!!

The Pumping Lemma:

สรุป → contradiction prov

assum
• Given a infinite regular language L

จะ L is not Regular

prove L ⇒ not regular เรา contra L ⇒ regular อีกคน prove

p ⇒ true
ไม่ set p ⇒ false
ถ้าเท่าได้ ต้อง สมมุติ

• there exists an integer m

เมื่อที่เสนอ L = regular → มีจำนวน state m

เราเล่น
• for any string $w \in L$ with length $|w| \geq m$

① เลือก

แบบ ผลบวก

pegeon hole principle

• ~~we~~ ตรงเขา can write $w = x\,y\,z$

② แบ่ง → เรียงไปเขา

• with $|x\,y| \leq m$ and $|y| \geq 1$

③ เราเล่น

เลือกค่า i

• such that: $x\,y^i\,z \in L$ $\quad i = 0, 1, 2, \ldots$

เราต้องหาตรง $\notin L$ → เราต้องการ prove ได้

ถ้าอยู่ → เป็น regular

ออกจากไม่อยู่ → ไม่เป็น regula

ไม่สอดคล้องกับ prove ตาม assume ไว้

L is not regular

Applications

of

the Pumping Lemma

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

is not regular

Proof: Use the Pumping Lemma

ก่อนจะใช้ → พิสูจน์ไปก่อนว่าหา NFA, DFA, regular exp, reg grammar ไม่ได้

เป็นแค่เครื่องมือ

ต้องพิสูจน์ในระดับหนึ่งว่ามันไม่ใช่ regular

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

Assume for contradiction

that L is a regular language สมมติฐาน

Since L is infinite

we can apply the Pumping Lemma

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

ต้องมี

Let m be the integer in the Pumping Lemma

We

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

length $|w| \geq m$

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

เลือกมาดีๆ ถ้าหากเงื่อนไขเลือก บางภาษาหาค่าไม่เจอนับ pump เรื่อยๆ ไม่หลุด

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

ตรงข้ามเลือก

From the Pumping Lemma

เรื่อยไป

it must be that length ① $|x\,y| \le m,$ ② $|\,y| \ge 1$

$$xyz = a^m b^m \;=\; \overbrace{a\ldots a\,a\ldots a\,a\ldots a}^{m}\overbrace{b\ldots b}^{m}$$

$$\underbrace{a\ldots a}_{x}\,\underbrace{a\ldots a}_{y}\,\underbrace{a\ldots ab\ldots b}_{z}$$

xy ไม่เกิน m ตัว

เลือก y มา

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

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

ดึ้งกน

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

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

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

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

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

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

ทำให้จำนวน a ทั้งหมดเกินมา k ตัว

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!!!

contra

↓

พิสูจน์ว่าเป็น reg ไม่ได้

↓

L is not reg

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

Conclusion: L is not a regular language

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