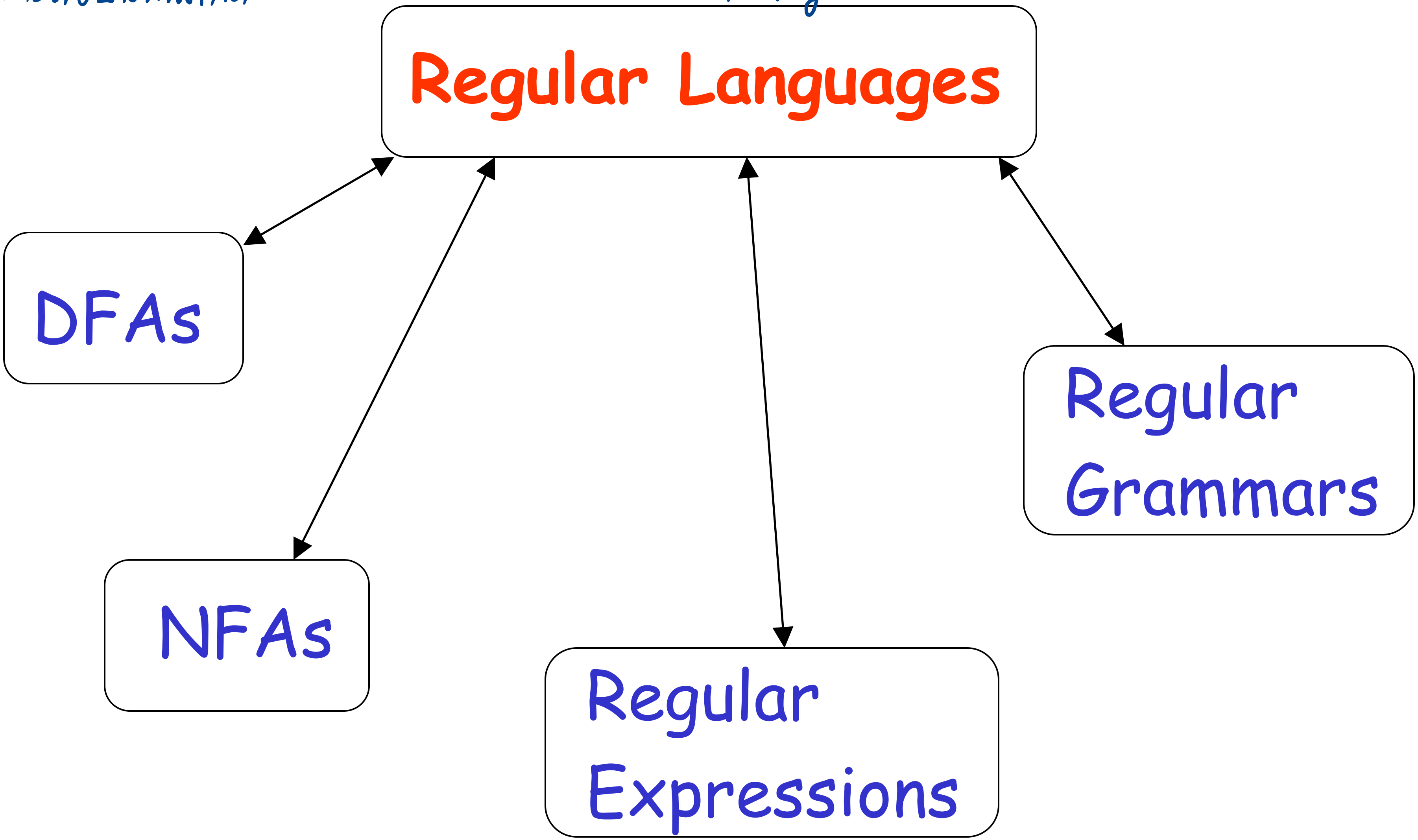
Standard Representations of Regular Languages

พิสูจน์ว่าเป็น
ภาษา reg
- อย่างใดอย่างหนึ่งก็ได้

วิธีการแทนภาษา reg.



When we say: We are given a Regular Language L

We mean: Language L is in a 4 วิธี หน้าที่แล้วได้ standard representation

Elementary Questions

about

Regular Languages

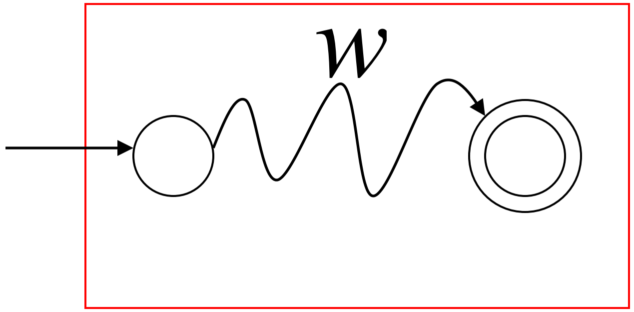
Membership Question

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

สร้าง ป้อน str w เข้าไป

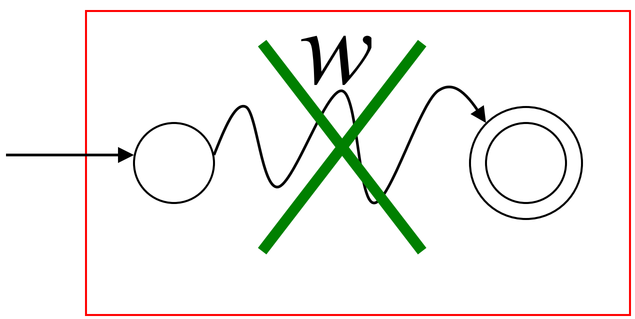
Answer: Take the DFA that accepts L
and check if w is accepted

DFA



$w \in L$

DFA



$w \notin L$

Question: Given regular language L

how can we check

if L is empty: $(L = \emptyset)$?

L เป็น ภาษาว่าง

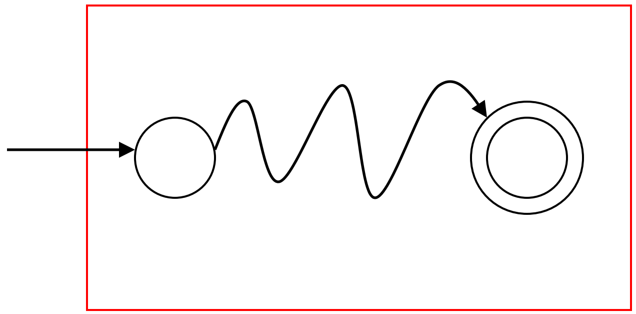
สร้าง

Answer: Take the DFA that accepts L

(ต้องไม่มี path ที่พาเราไป final state

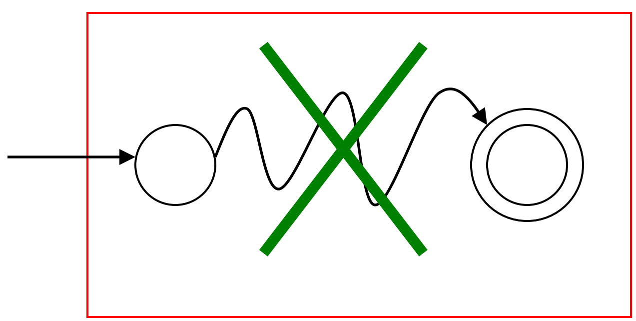
Check if there is any path from the initial state to a final state

DFA



$$L \neq \emptyset$$

DFA



$$L = \emptyset$$

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

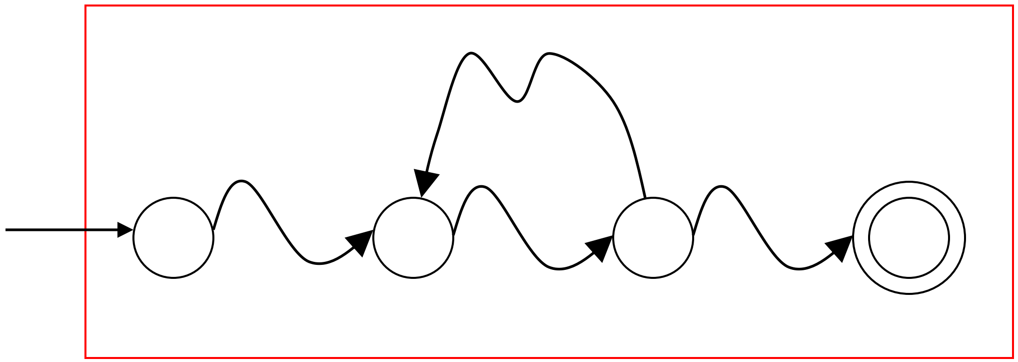
เป็นภาษาจำกัด

Answer: Take the DFA that accepts L

ถ้ามี คือไม่จำกัด

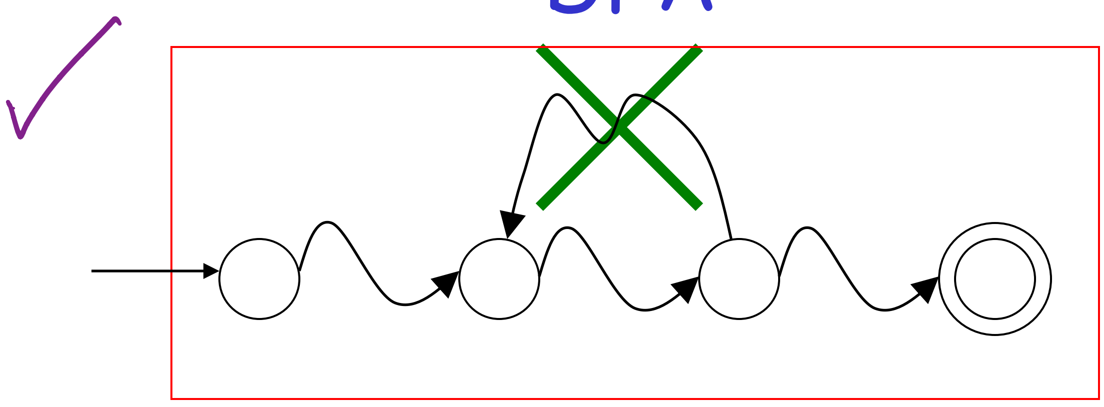
Check if there is a walk with cycle
from the initial state to a final state

DFA



L is infinite

DFA



L is finite

เช็ค $L_1 = L_2$

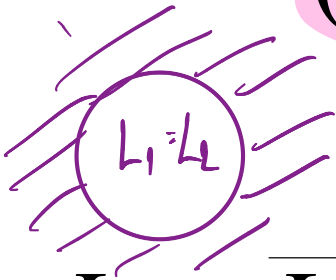
Question: Given regular languages L_1 and L_2 how can we check if $L_1 = L_2$?

L_1 = L_2 $\rightarrow$ $L_1 = L_2$ วงทับกันพอดี

ใช้คุณสมบัติ

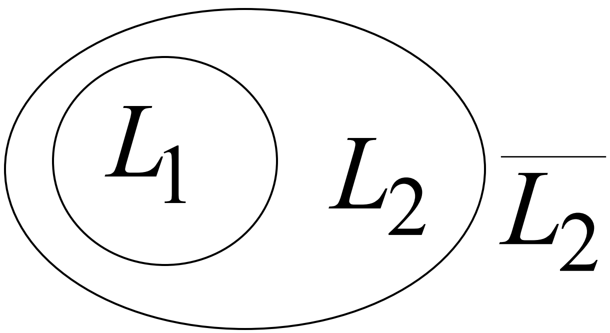
Answer: Find if $(L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2) = \underline{\varnothing}$

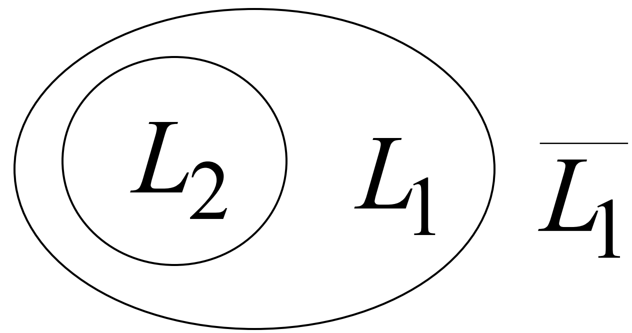
$\varnothing$ แสดงว่า $L_1 = L_2$



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

$L_1 \cap \overline{L_2} = \emptyset$ and $\overline{L_1} \cap L_2 = \emptyset$



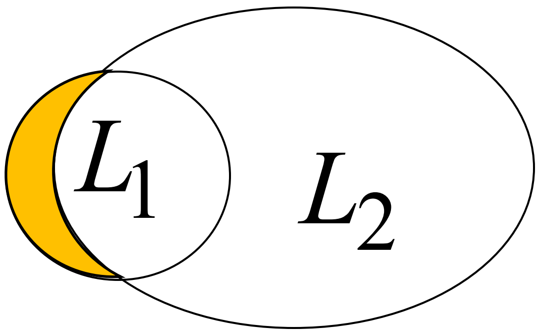


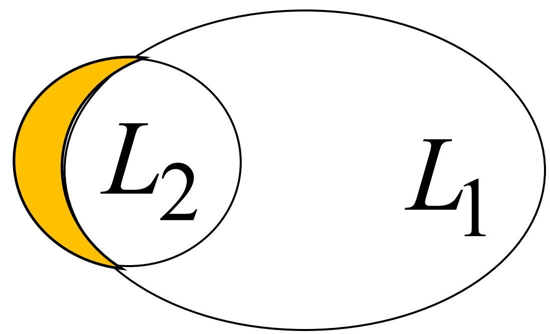
$L_1 \subseteq L_2$ $\qquad$ $L_2 \subseteq L_1$

$$L_1 = L_2$$

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

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





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

$$L_1 \neq L_2$$

Non-regular languages

Non-regular languages

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

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

$a^{100} b^*$

Regular languages

$$a^*b$$

$$b^*c+a$$

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

etc...

How can we prove that a language L is not regular? เราจะพิสูจน์ไงว่าเป็น non reg.

L is reg ← DFA, NFA, reg ex, reg gram

"ไม่ใช่วิธีที่ถูกต้อง"

Prove that there is no DFA that accepts L

Problem: this is not easy to prove

ใช้พิสูจน์ L ไม่ใช่ reg

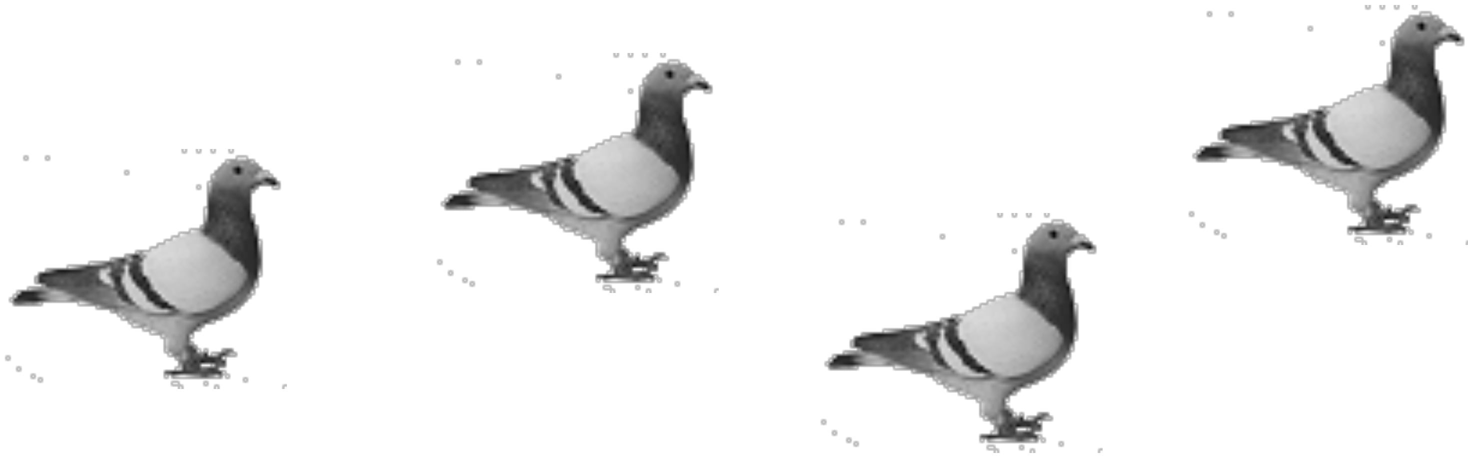
Solution: the Pumping Lemma !!!

พิเจี่ยนโฮ

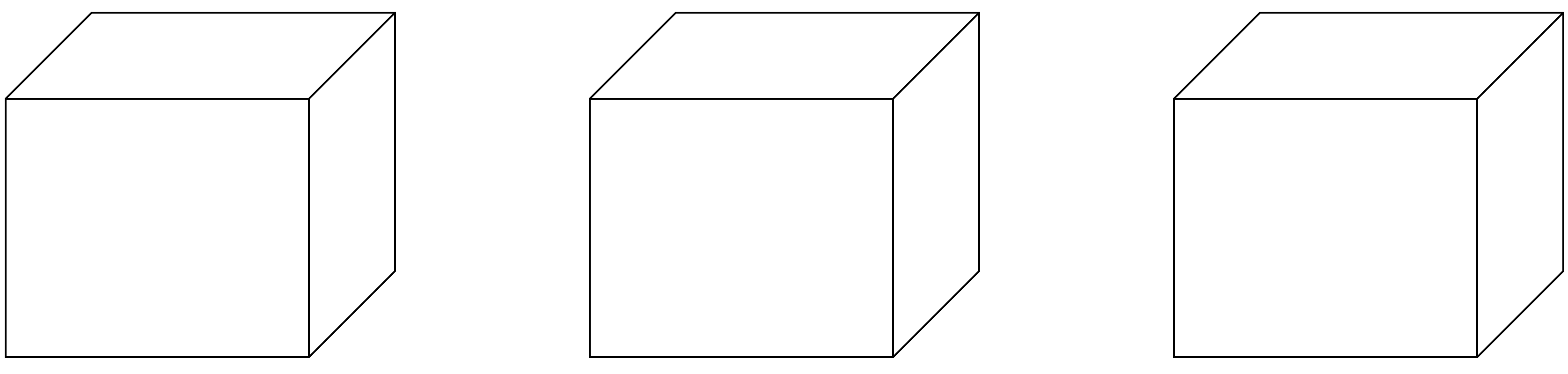
The Pigeonhole Principle

รัง นกพิราบ

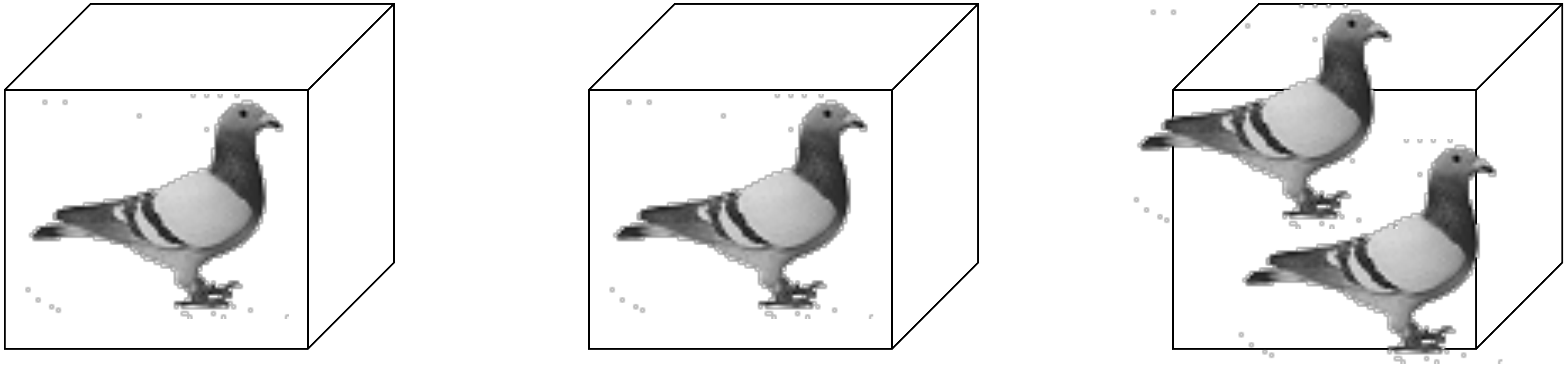
4 pigeons



3 pigeonholes



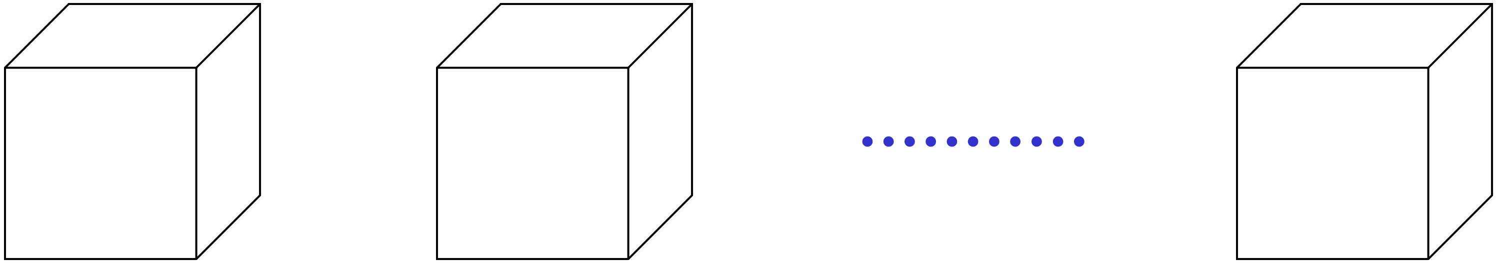
A pigeonhole must contain at least two pigeons



n pigeons

นก กล่อง

m pigeonholes $\quad n > m$



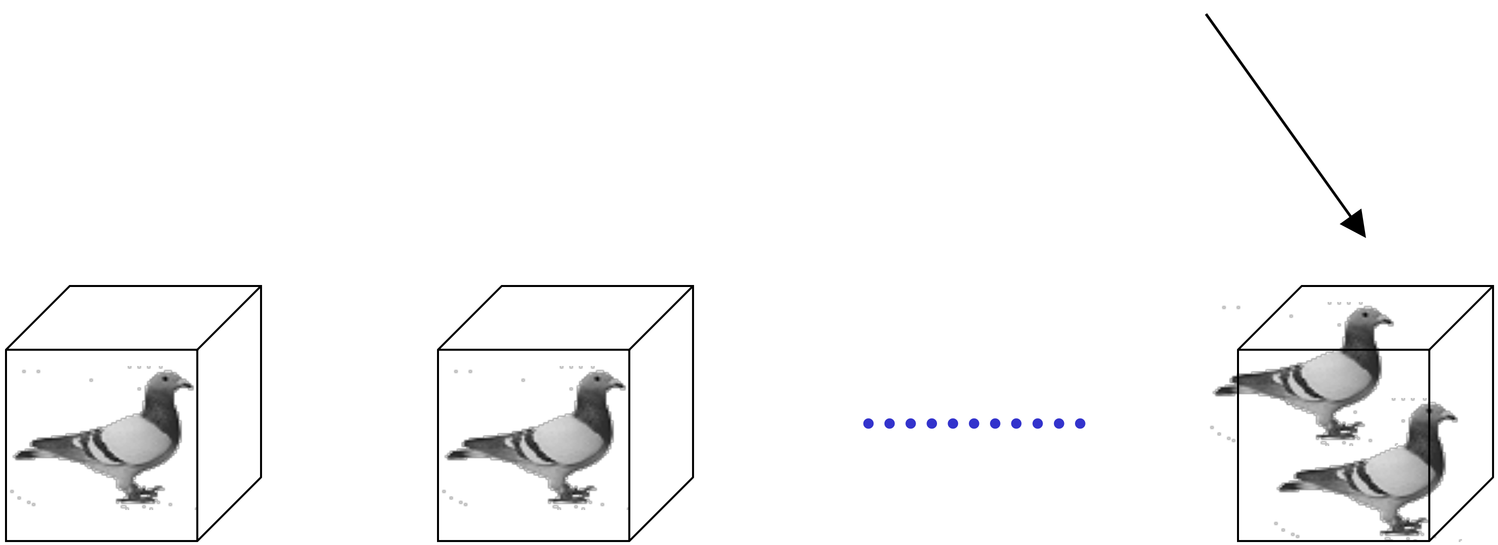
The Pigeonhole Principle

n pigeons

m pigeonholes

$$n > m$$

There is a pigeonhole with at least 2 pigeons

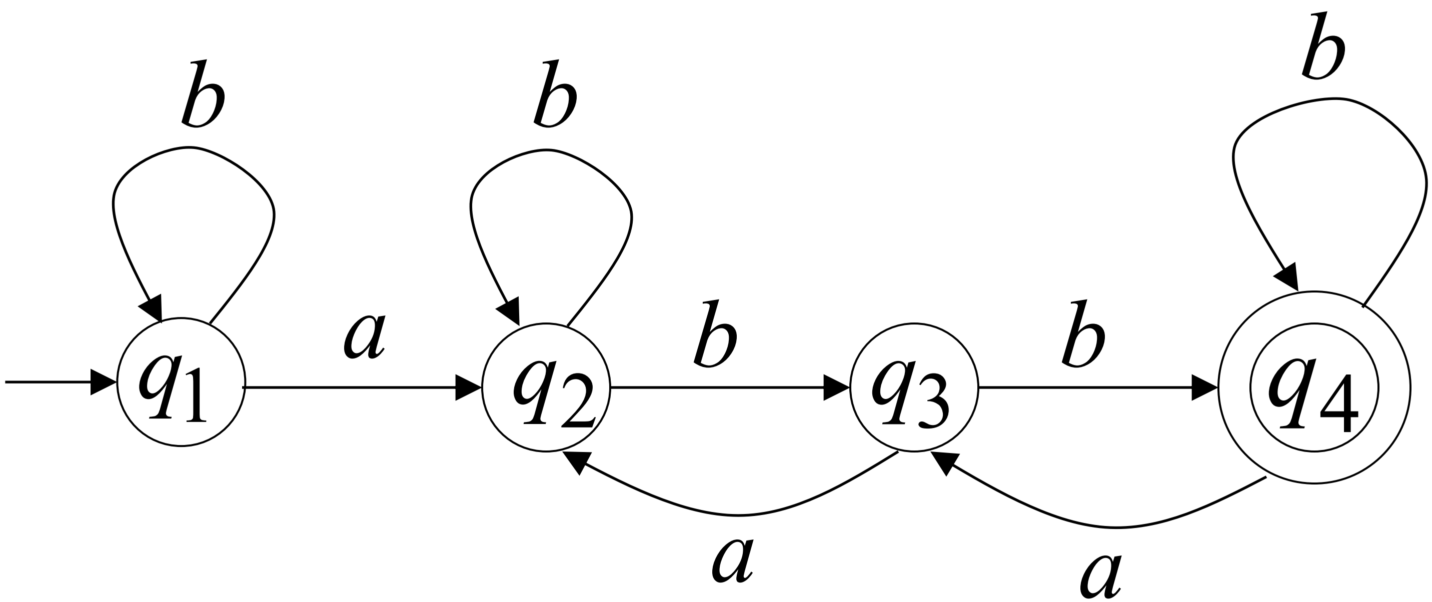


The Pigeonhole Principle

and

DFAs

DFA with 4 states

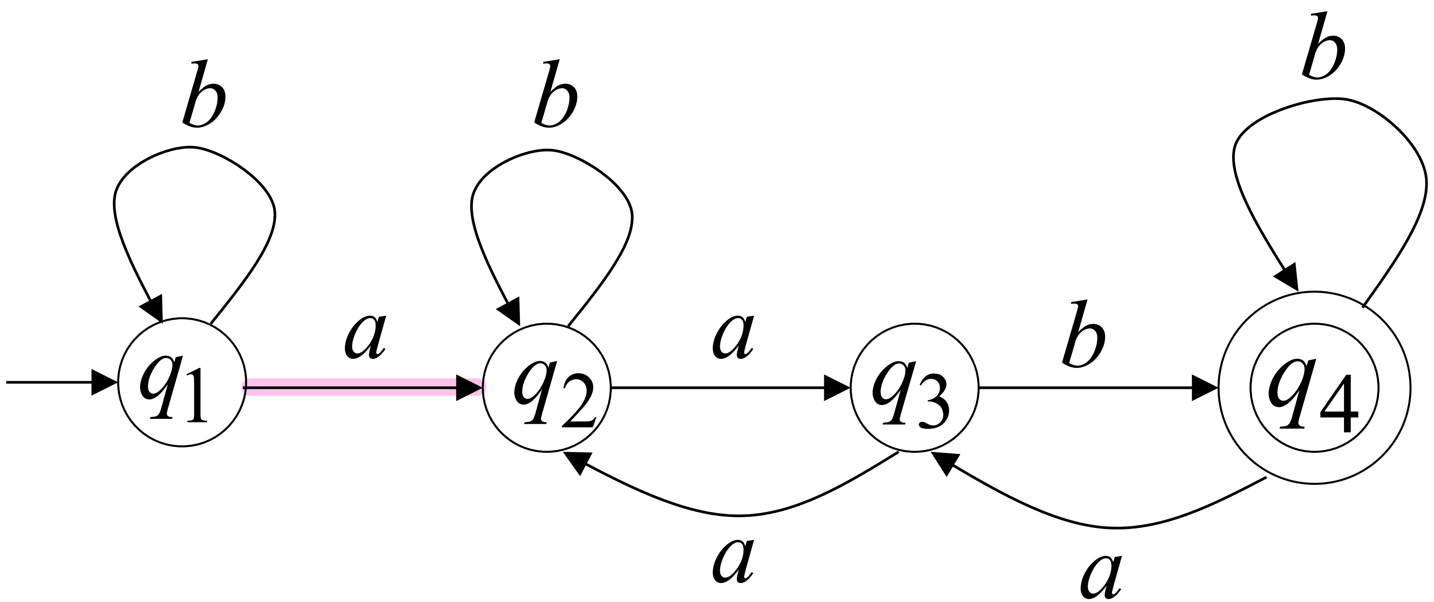


In walks of strings: a

aa

aab

no state
is repeated



In walks of strings: $aabb$

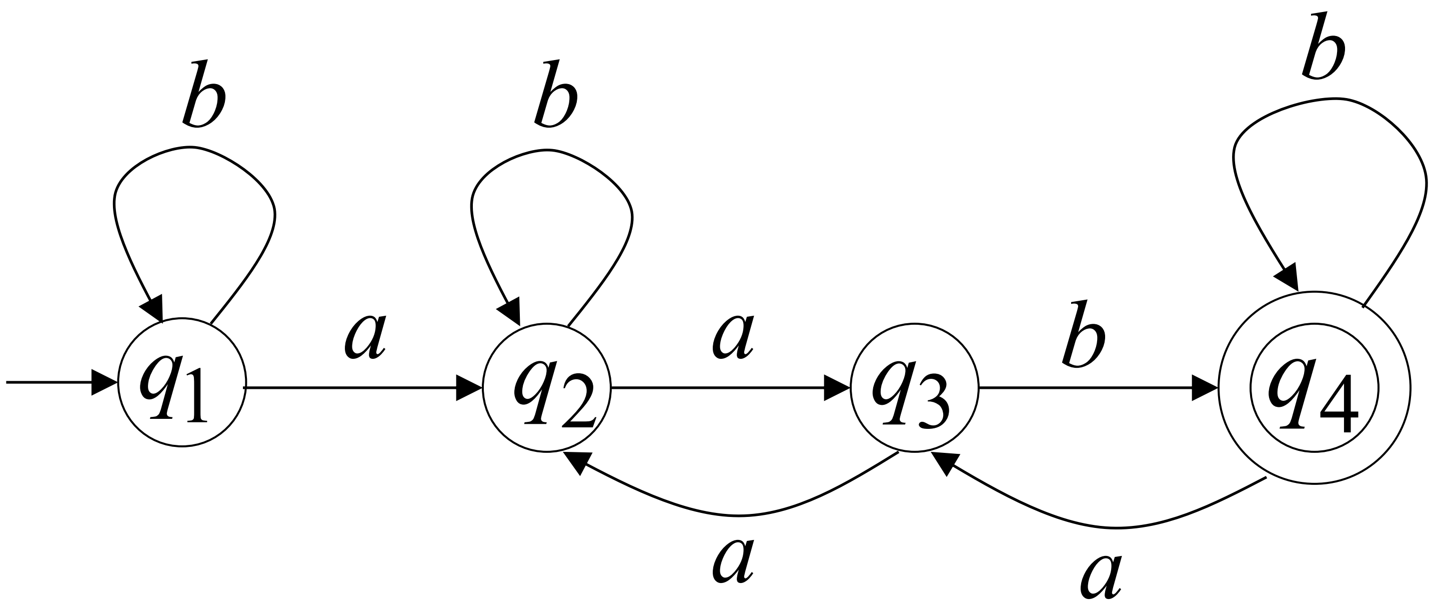
$bbaa$

$abbabb$

$abbbabbabb...$

a state is repeated

ความ str มีห
ยาวเกินอะไรซักอย่าง
มีหจะมีบางstate โดนซ้ำ

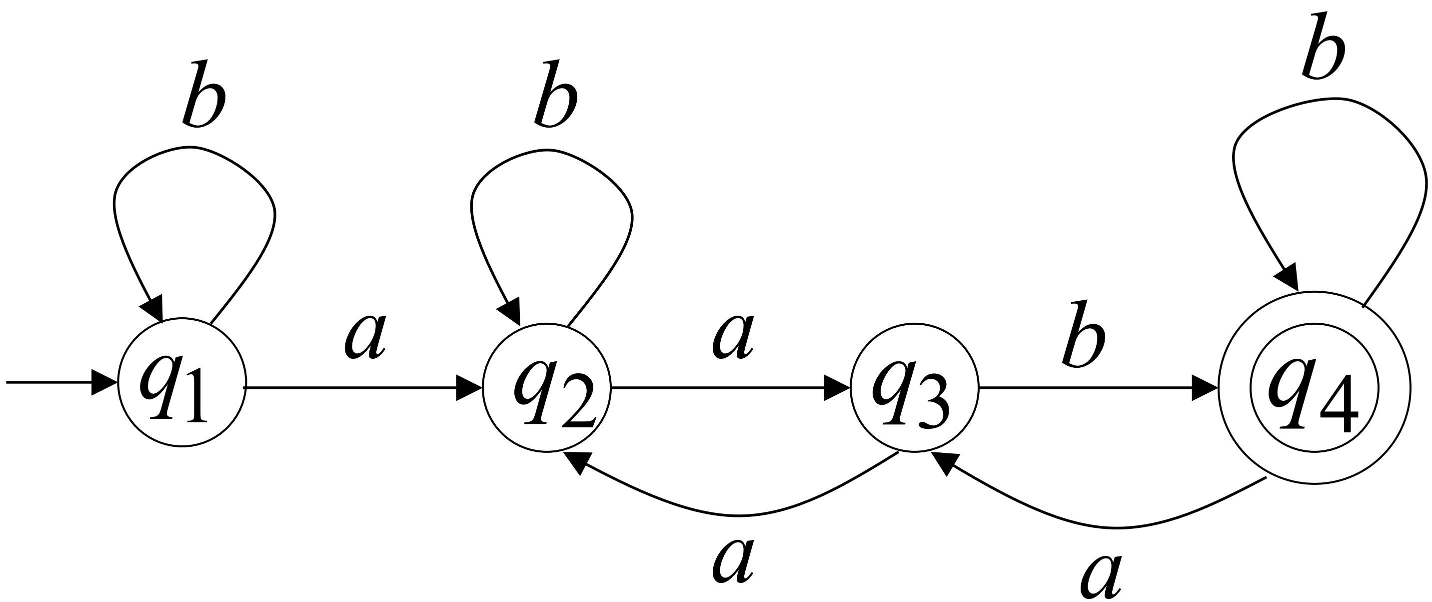


ความยาว str มากกว่าหรือเท่ากับ

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

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

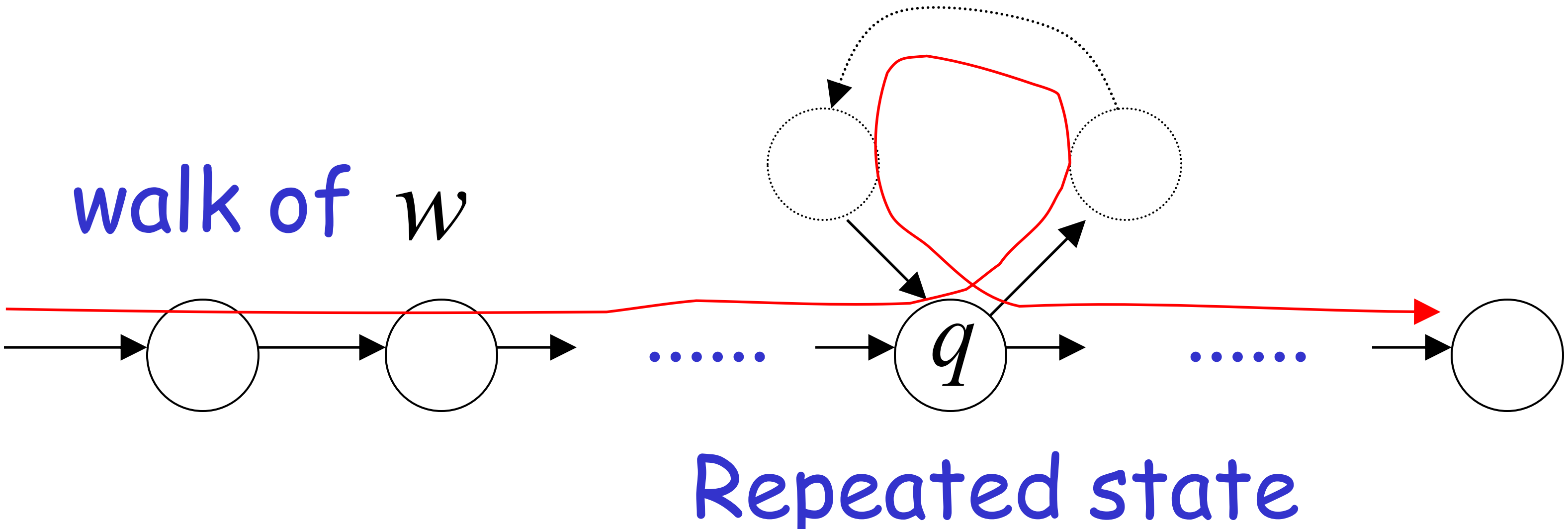
Thus, a state must be repeated



In general, for any DFA:

String w has length $\geq$ number of states

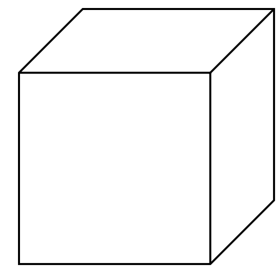
A state q must be repeated in the walk of w



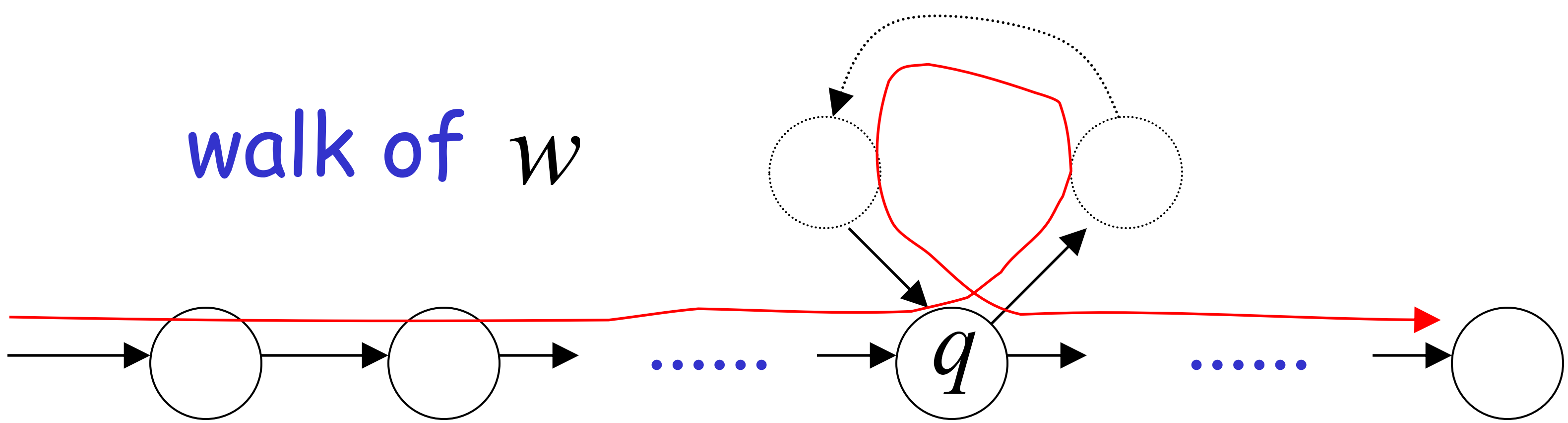
In other words for a string w:

$\xrightarrow{a}$ transitions are นกพิราบ pigeons

q states are รัง pigeonholes



walk of w



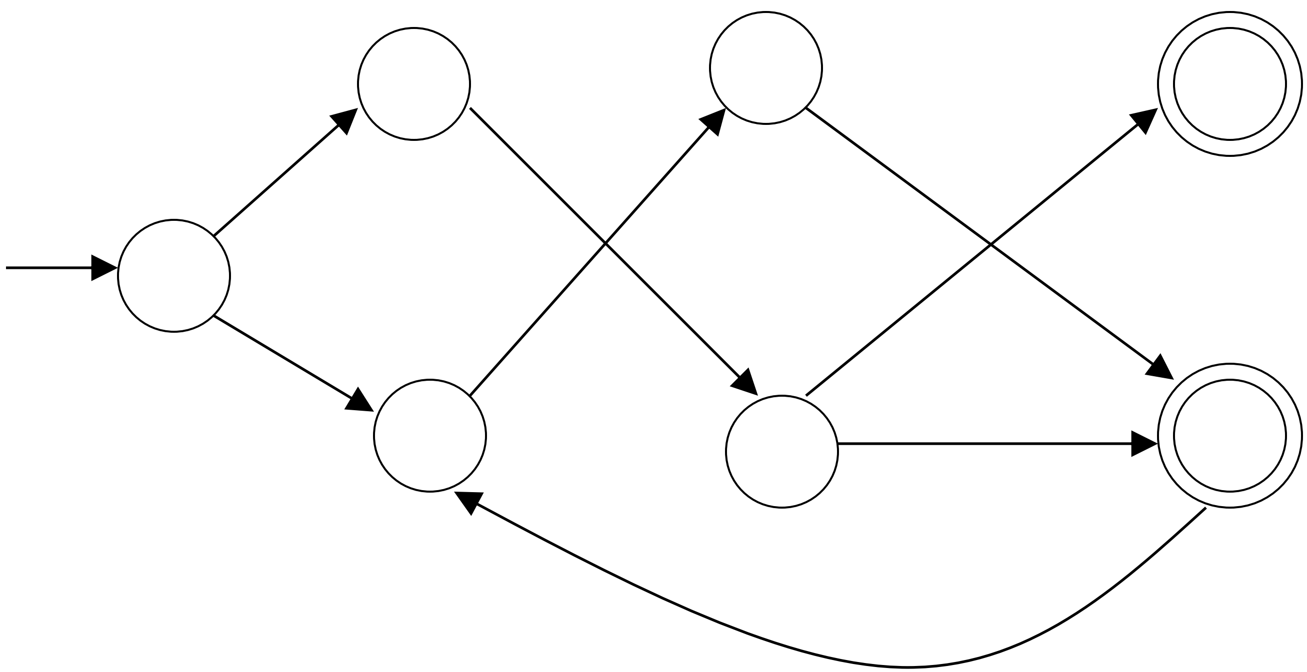
Repeated state

The Pumping Lemma

การทำซ้ำ str
repeat

Take an infinite regular language L

There exists a DFA that accepts L



m =
จำนวน states
ใน DFA ที่ยอมรับ
ภาษา L

Take string w with $w \in L$

ทำ initial state $\longrightarrow$ final state

There is a walk with label w:

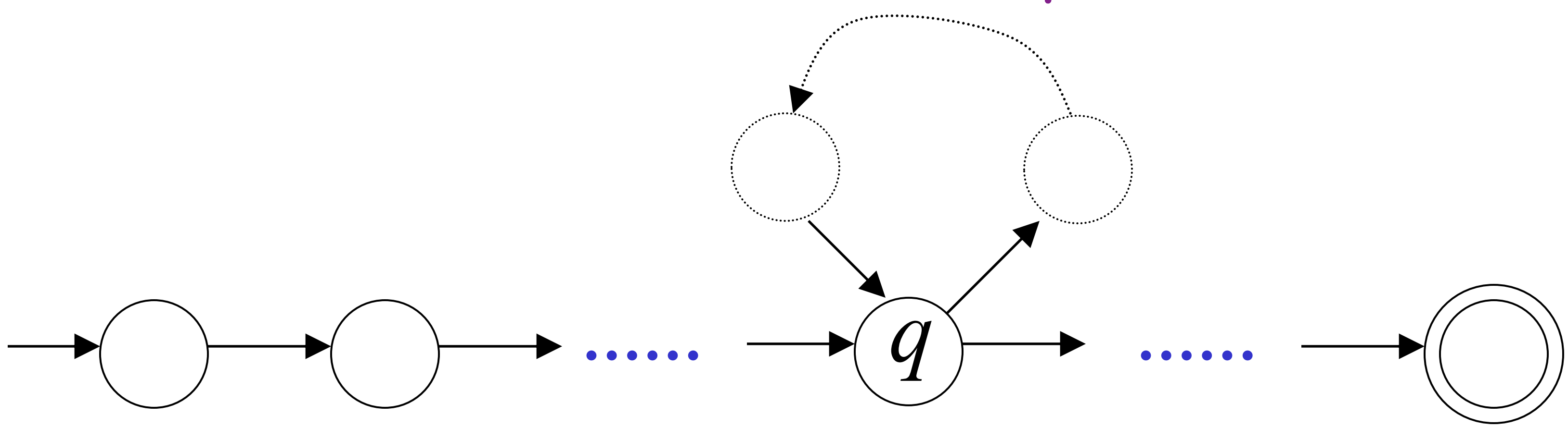
If string w has length $|w| \geq m$ (number of states of DFA)

then, from the pigeonhole principle:

ที่ไดหซ้ำ

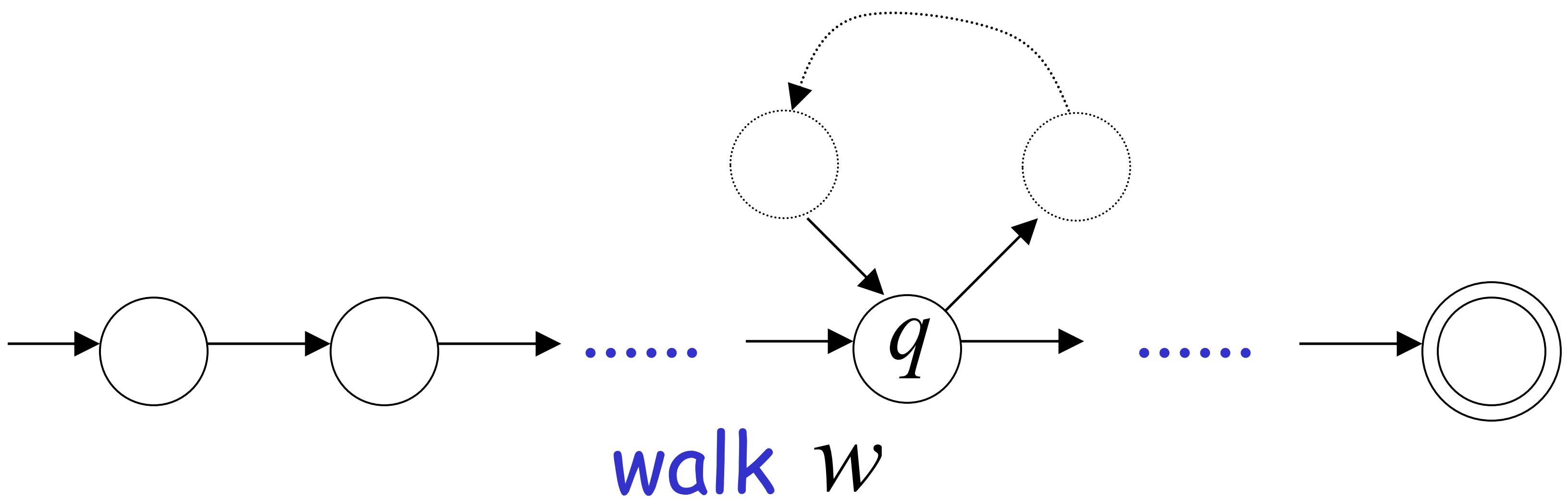
a state is repeated in the walk w

เช่น state q



walk w

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

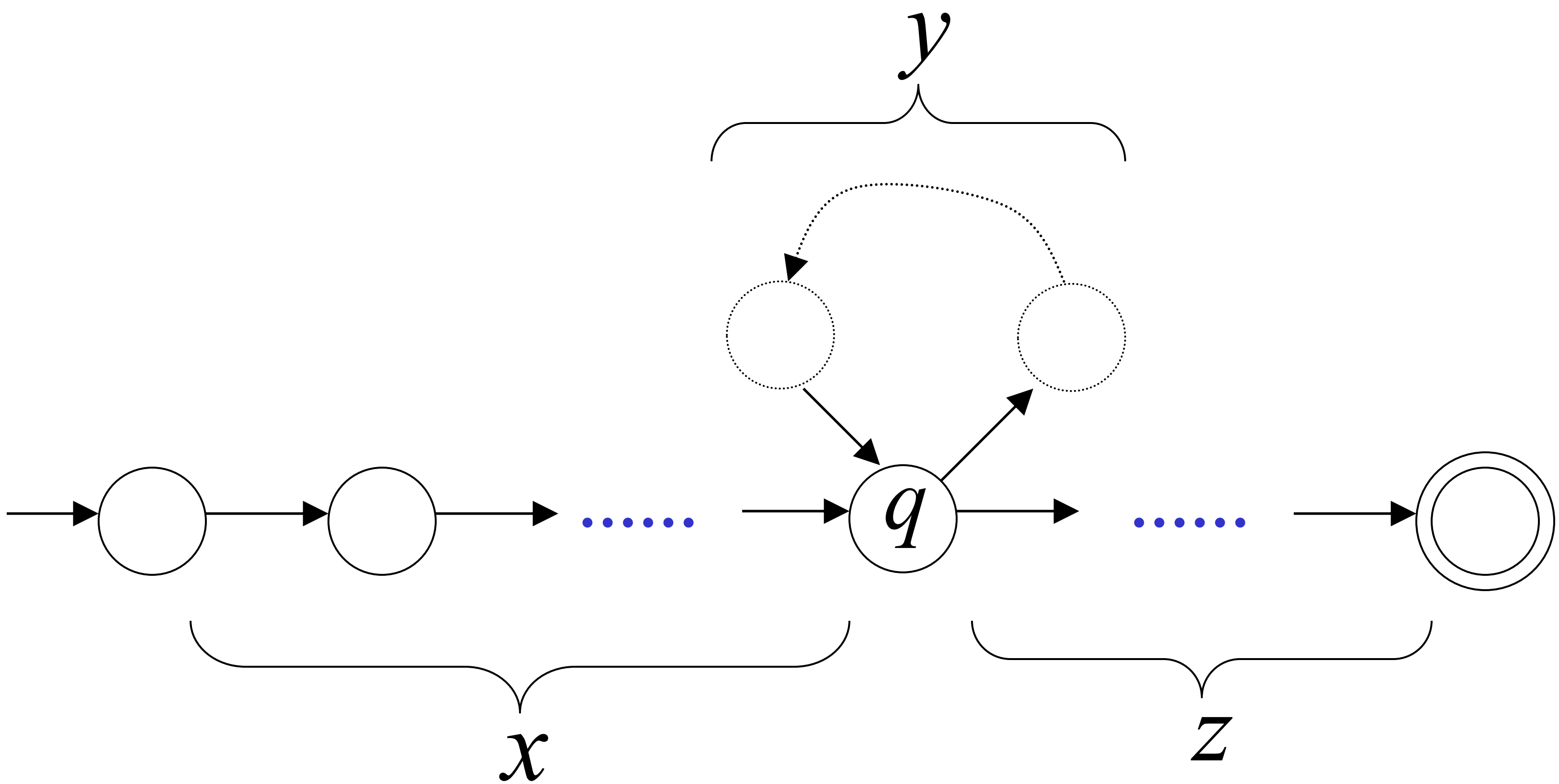


จำ

Str w แบ่งได้ 3 ส่วน

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

ส่วนของ Str ที่ถูกทำซ้ำ



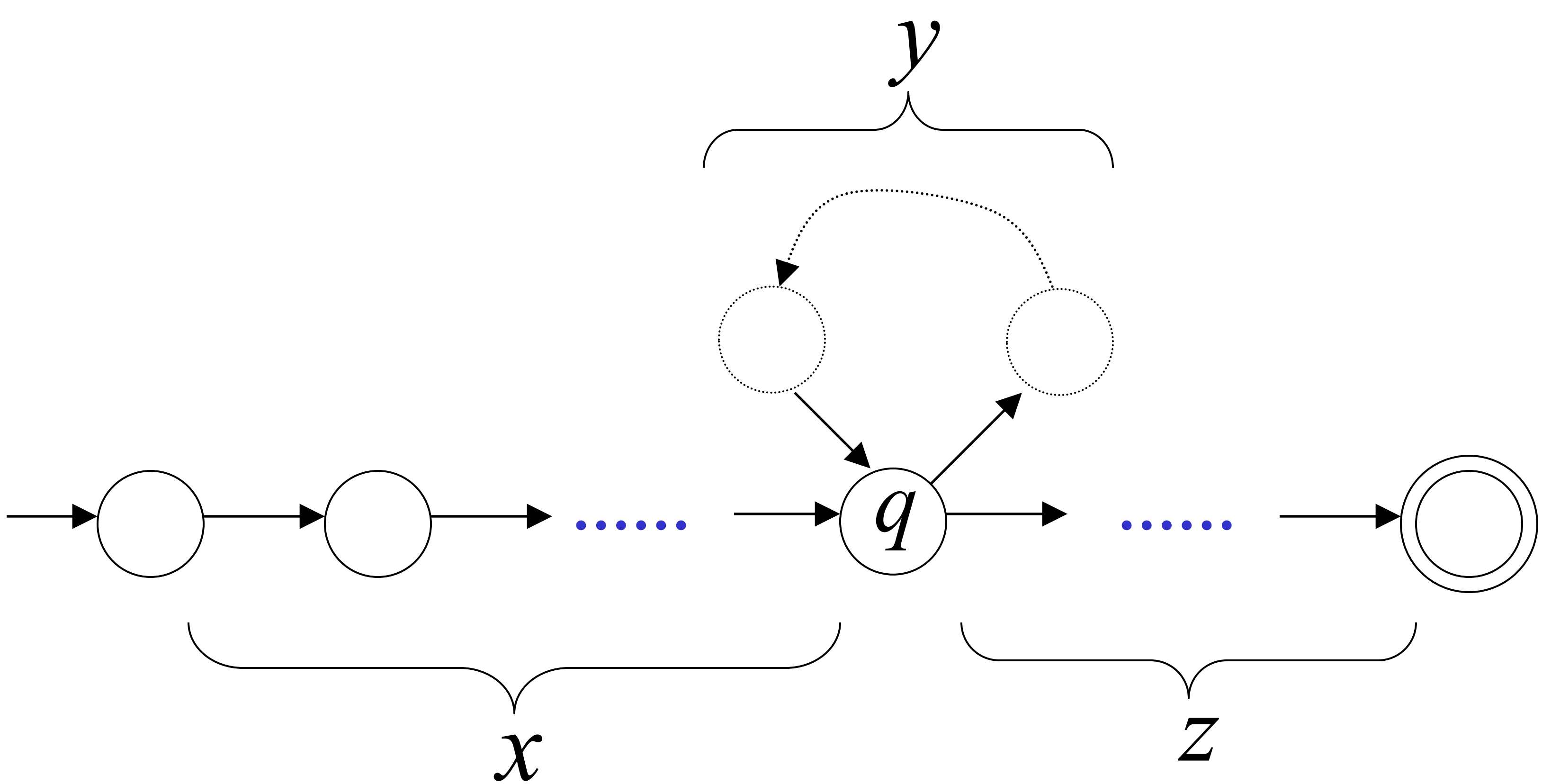
Observations: ① length $|x\,y| \le m$ number of states of DFA

จำเงื่อนไข

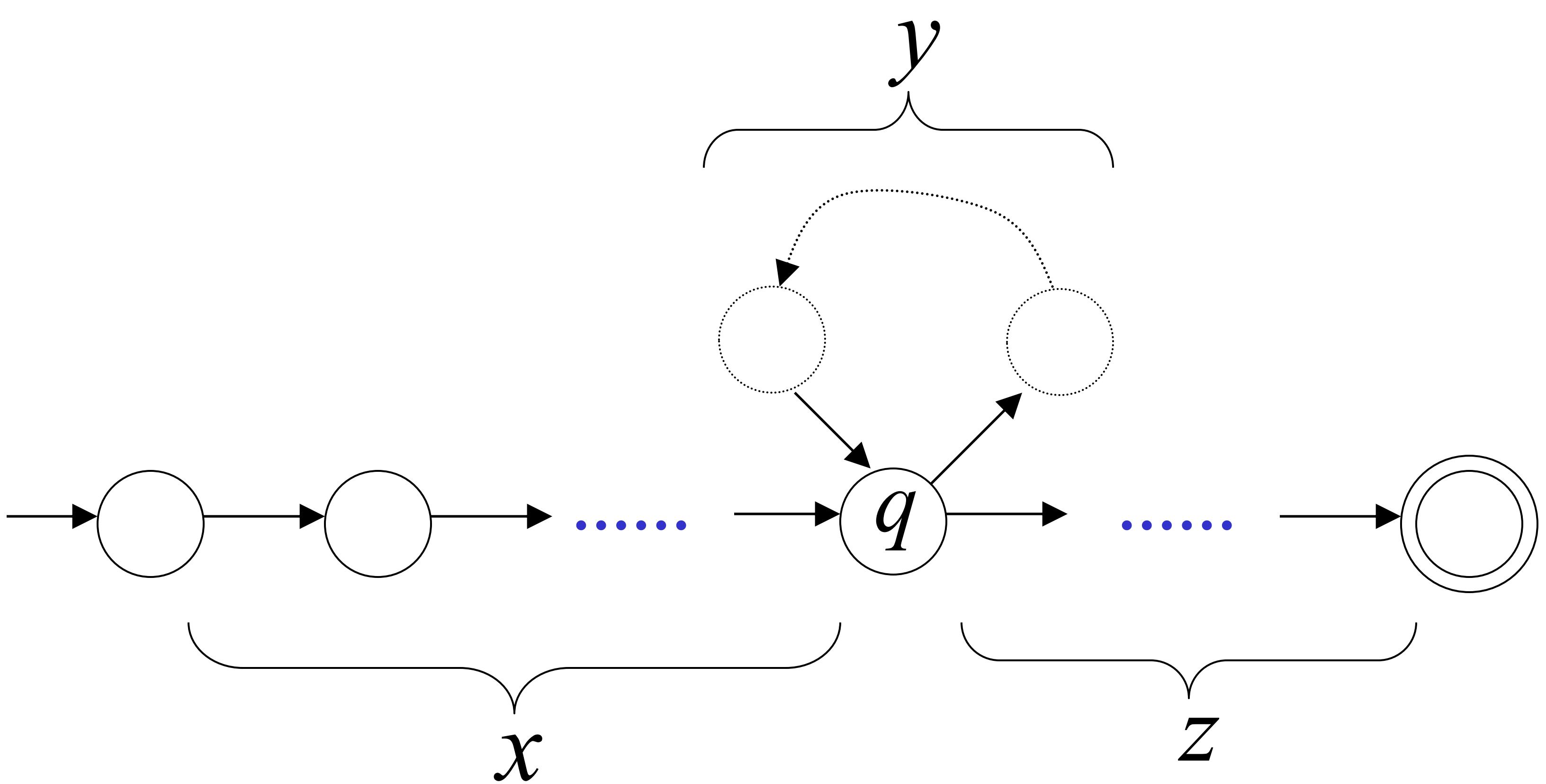
state ทั้งหมดใน DFA

มันจะได้ซ้ำได้

② length $|y| \ge 1$



Observation: The string $x\ z$ Str ที่สั้นที่สุด
is accepted คือ repeat y 0 ครั้ง

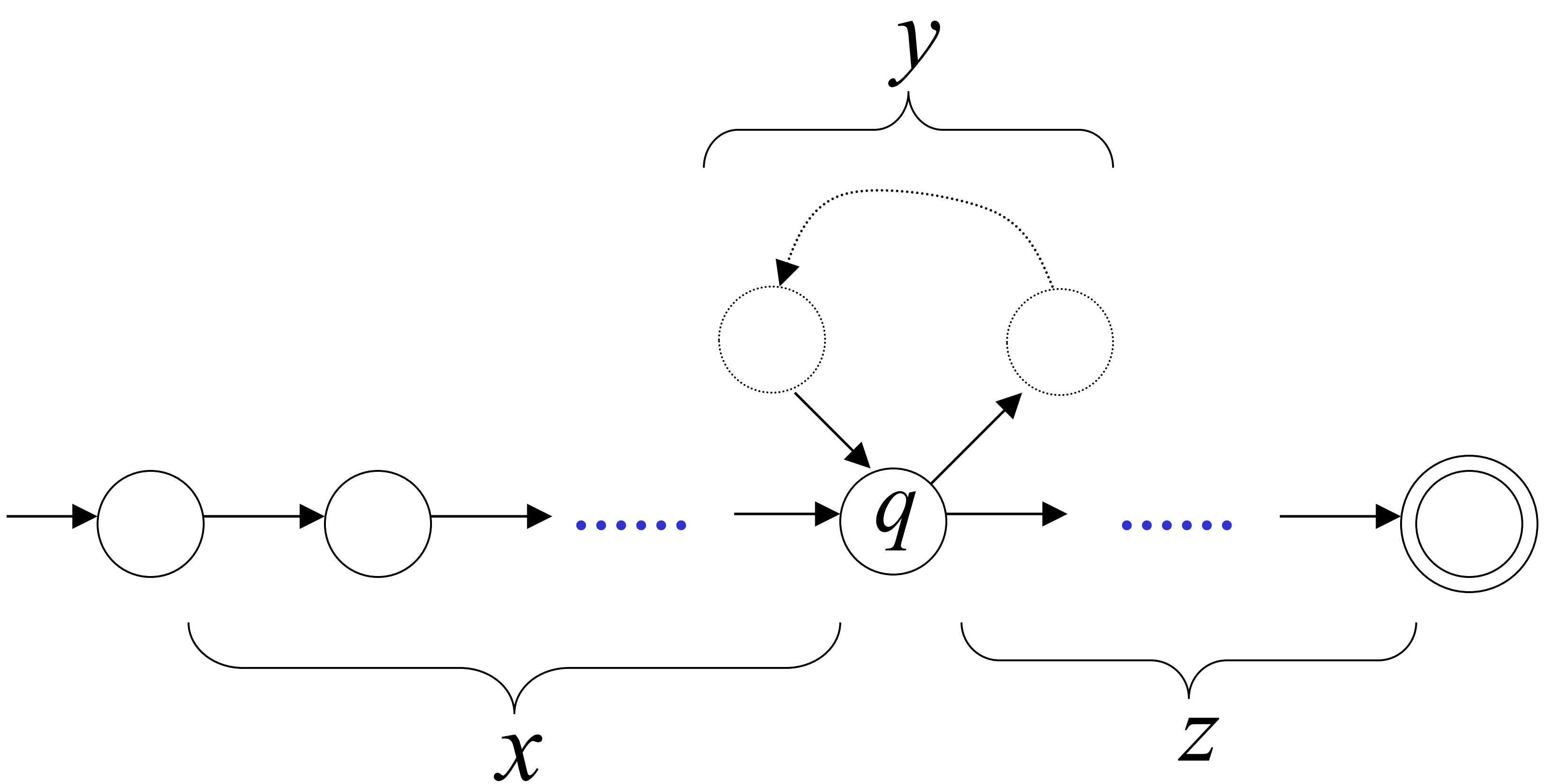


Observation: The string $x\,y\,y\,z$ is accepted

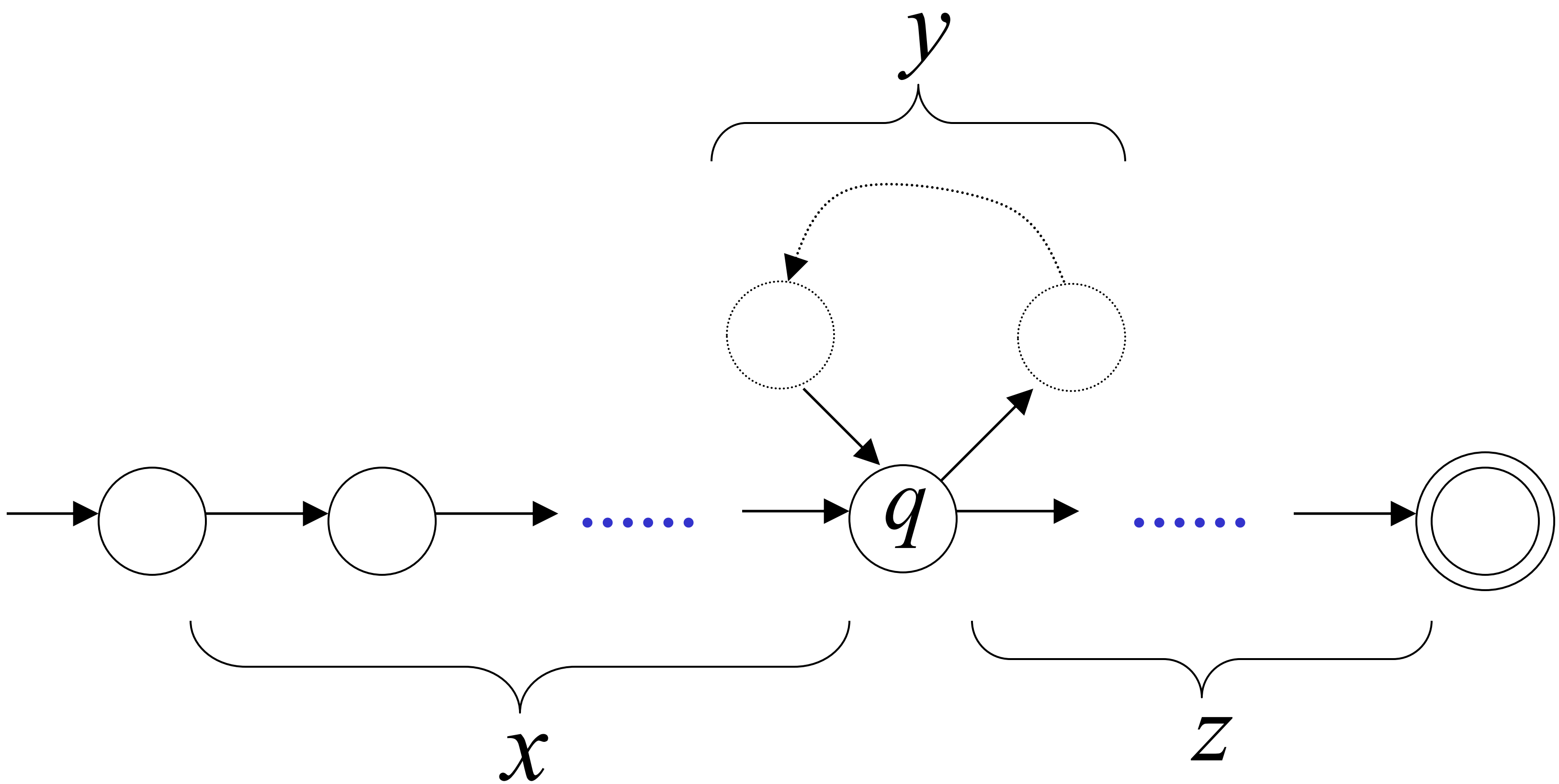
↳ rp. 2 ครั้ง

xyz

↳ repeat y 1 ครั้ง

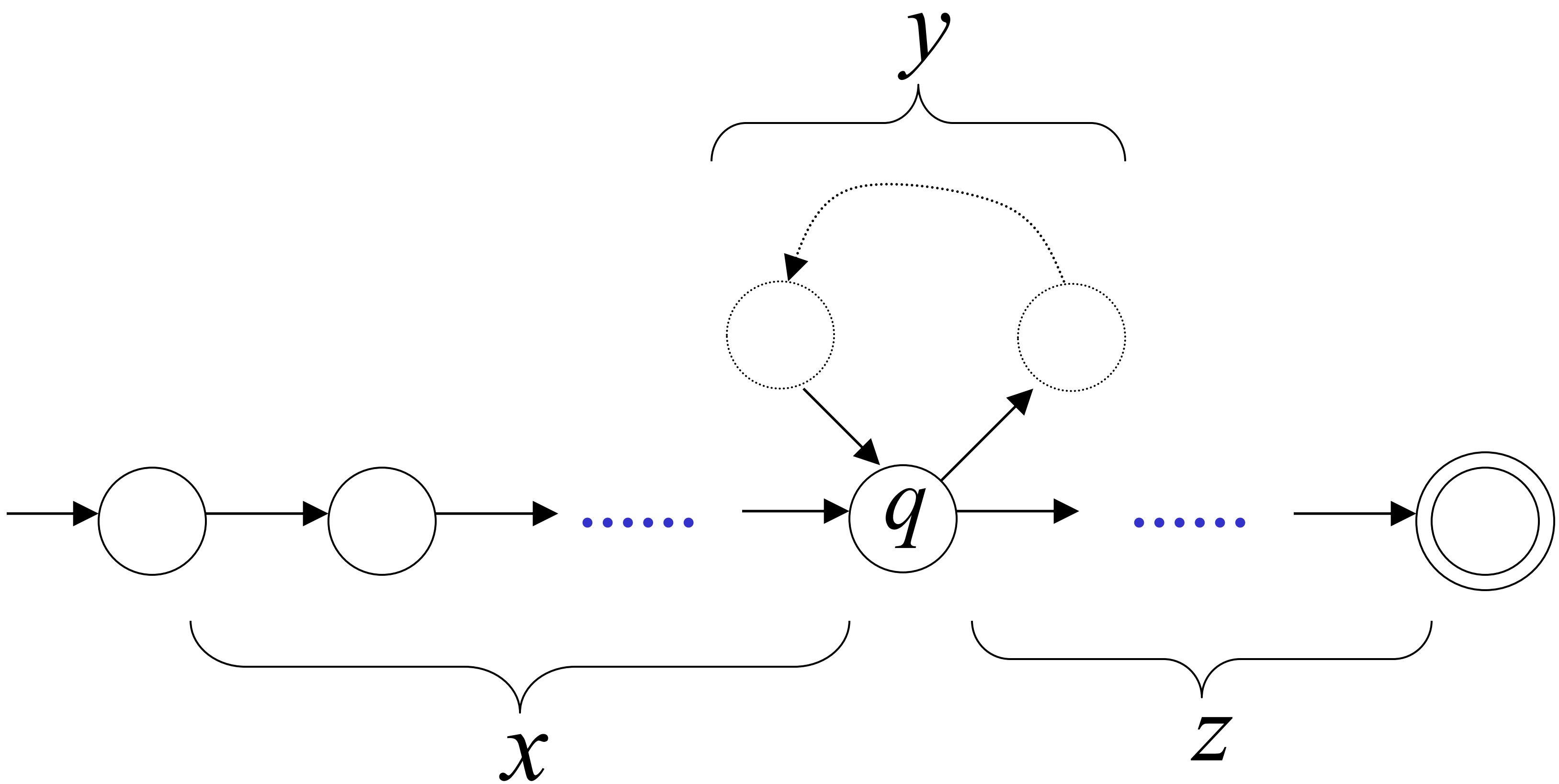


Observation: The string $x\ y\ y\ y\ z$ is accepted



In General: The string $x\,y^i\,z$ เขียนแบบนี้ได้

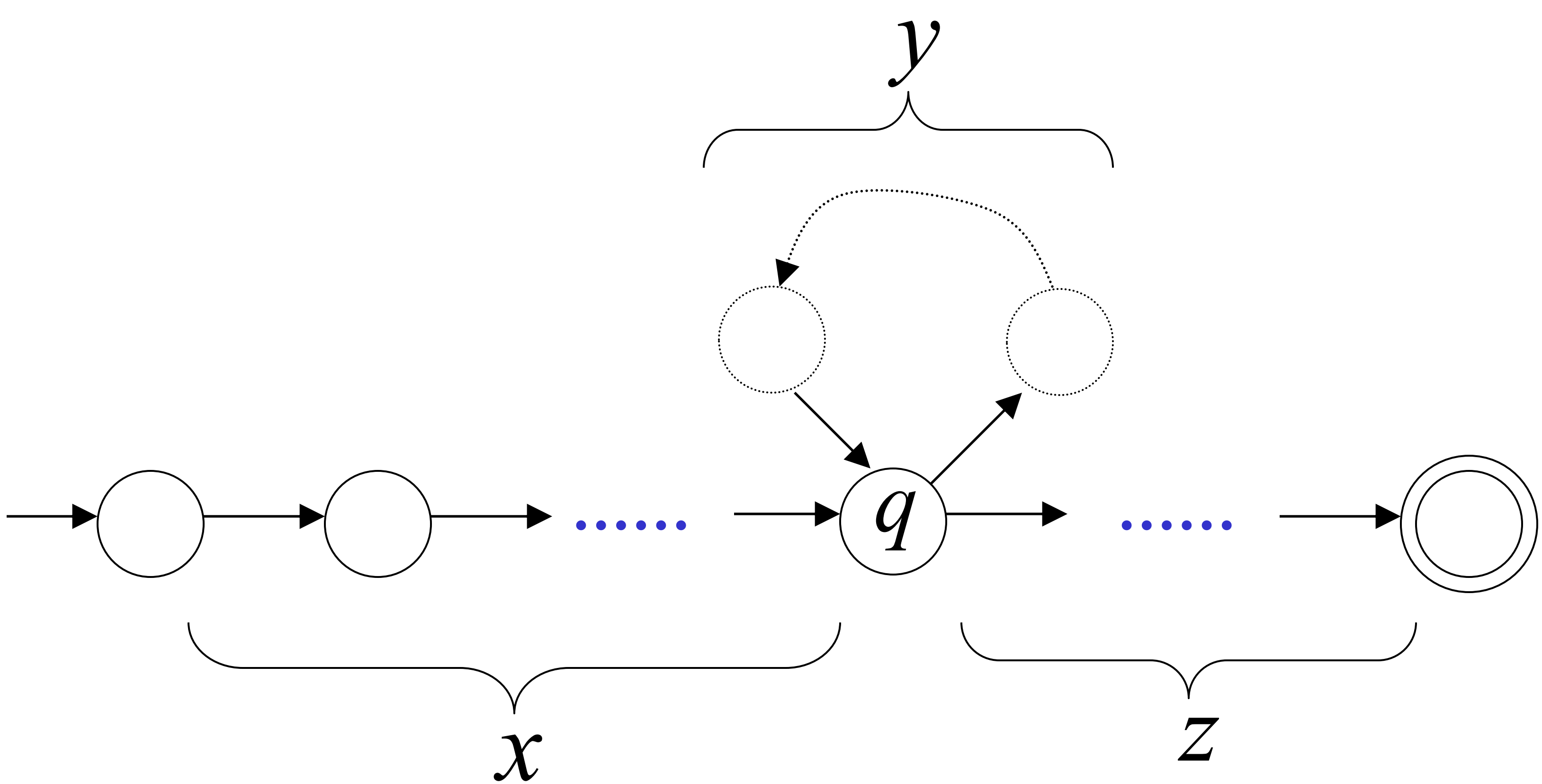
is accepted $i = 0, 1, 2, \ldots$



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

ไม่ว่าจะ plum กี่ครั้งก็ตาม

Language accepted by the DFA



In other words, we described:

The Pumping Lemma !!!

คือ Contradiction ← **The Pumping Lemma:** จะพิสูจน์ L ไม่ reg
prove

สมมติว่า L = reg

- Given a infinite regular language L

- there exists an integer m จำนวน state ใน DFA ของ L

we

เลือก str w ที่อยู่ใน L

เพื่อให้มันพิมพ์ซ้ำได้

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

ตามกฎ PH pri.

opponent ฝั่งตรงข้าม

- we can write $w = x\,y\,z$

ความยาว x y รวมกัน

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

เราคาดหวังว่า $x\,y^i z \notin L$ เพราะต้องพิสูจน์ว่า L isn't reg.

- 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

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

(ตรงข้าม) สมมติ

Assume for contradiction

that L is a regular language

ก่อน พิสูจน์โดยใช้ pum "ให้มั่นใจก่อนว่า เขียน DFA ไม่ได้จริง"
NFA

Since L is infinite

we can apply the Pumping Lemma

ใช้เฉพาะ: No reg เท่านั้น
ซึ่งต้องมั่นใจว่า $L \neq$ reg

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

Let m be the integer in the Pumping Lemma

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

length $|w| \geq m$

$xy^iz \notin L$ pum แล้ว "ต้องไม่อยู่ใน L"

- ถ้าอยู่ใน L = สรุปไม่ได้

$n = m$ เพื่อให้สอดคล้องกับ m

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

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

opponent

ความยาว = 0
pum ไม่ได้

From the Pumping Lemma

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

x กับ y รวมกัน

$$xyz = a^m b^m \quad = \quad \overbrace{\underbrace{a\ldots a}_{x}\underbrace{a\ldots a}_{y}\underbrace{a\ldots a}_{}}^{m}\!\underbrace{\overbrace{b\ldots b}^{m}}_{z}$$

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

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

We

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

xyyz

a มี k ตัว

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

with the final $z = a...ab...b$

k=0 ไม่ได้ งั้นจำนวน a ≠ จำนวน 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 $\{a^n b^n : \quad n \geq 0\}$