

Non-regular languages

(Pumping Lemma)

Non-regular languages

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

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

Regular languages

$$a^*b$$

$$b^*c + a$$

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

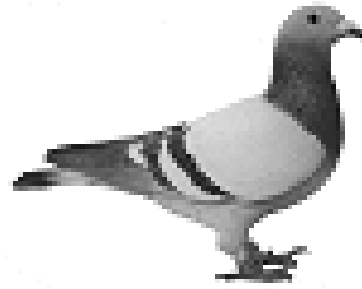
etc...

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

Prove that there is no DFA or NFA or RE that accepts L

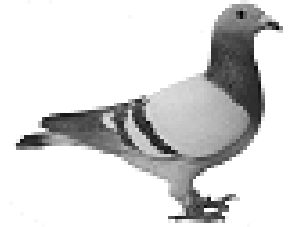
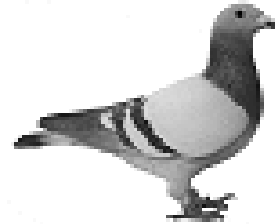
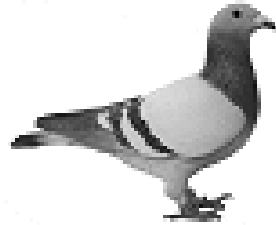
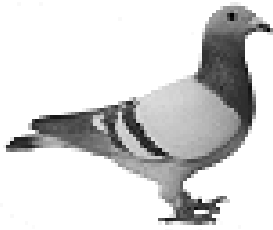
Difficulty: this is not easy to prove
(since there is an infinite number of them)

Solution: use the Pumping Lemma !!

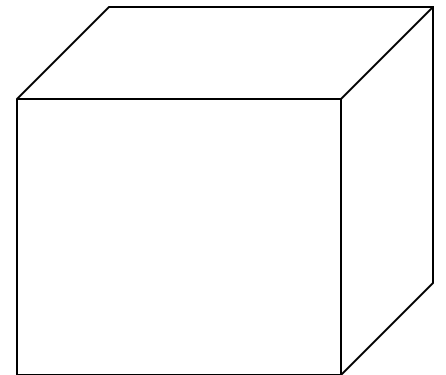
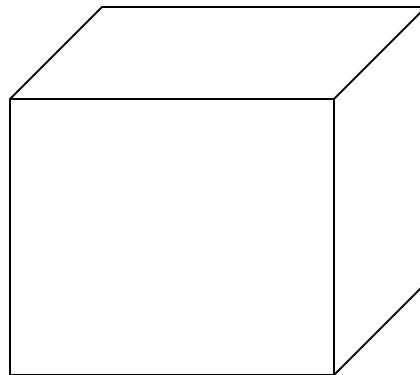
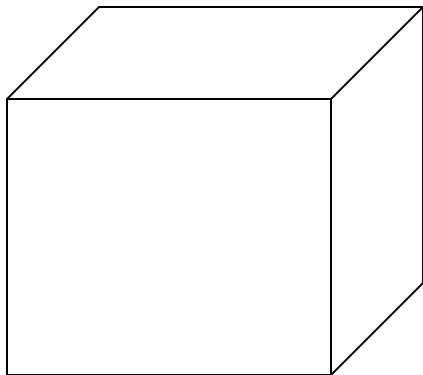


The Pigeonhole Principle

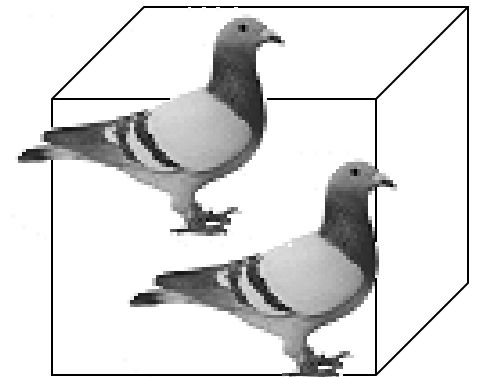
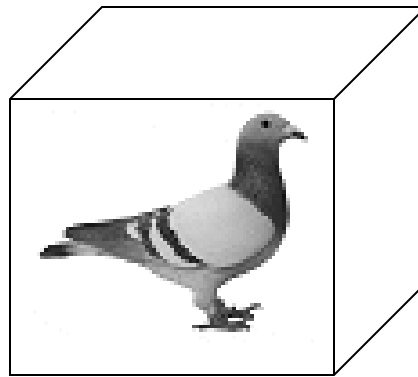
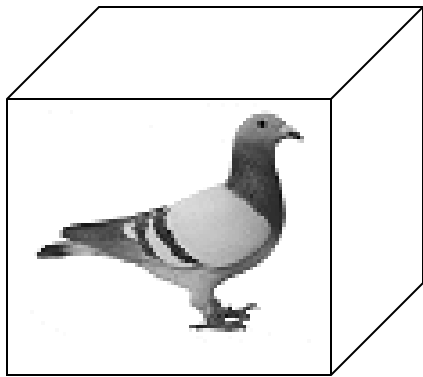
4 pigeons



3 pigeonholes



A pigeonhole must
contain at least two pigeons



n pigeons

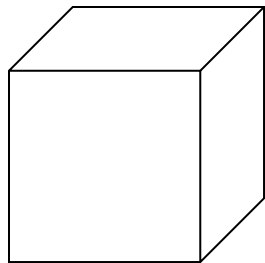
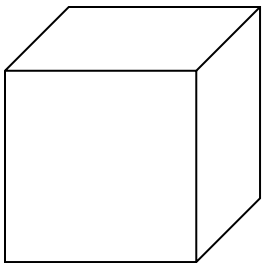


.....

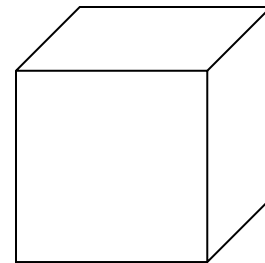


m pigeonholes

$n > m$



.....



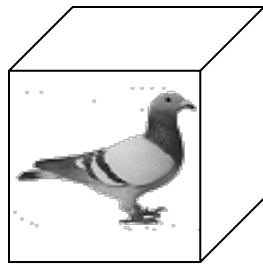
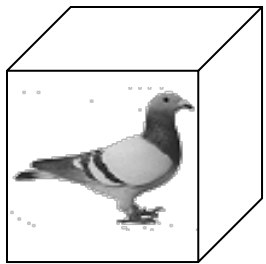
The Pigeonhole Principle

n pigeons

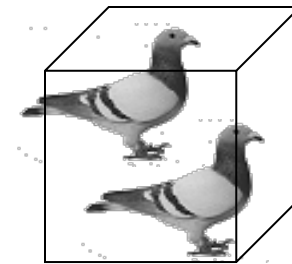
m pigeonholes

$$n > m$$

There is a pigeonhole
with at least 2 pigeons



.....

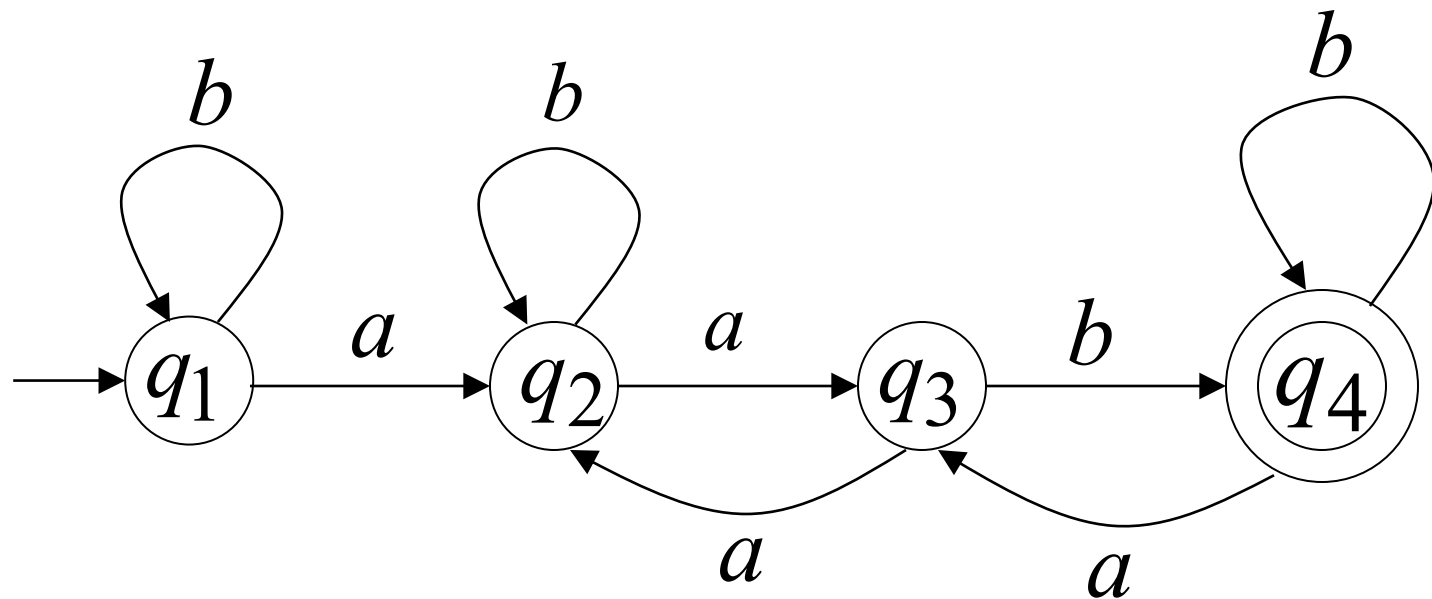


The Pigeonhole Principle

and

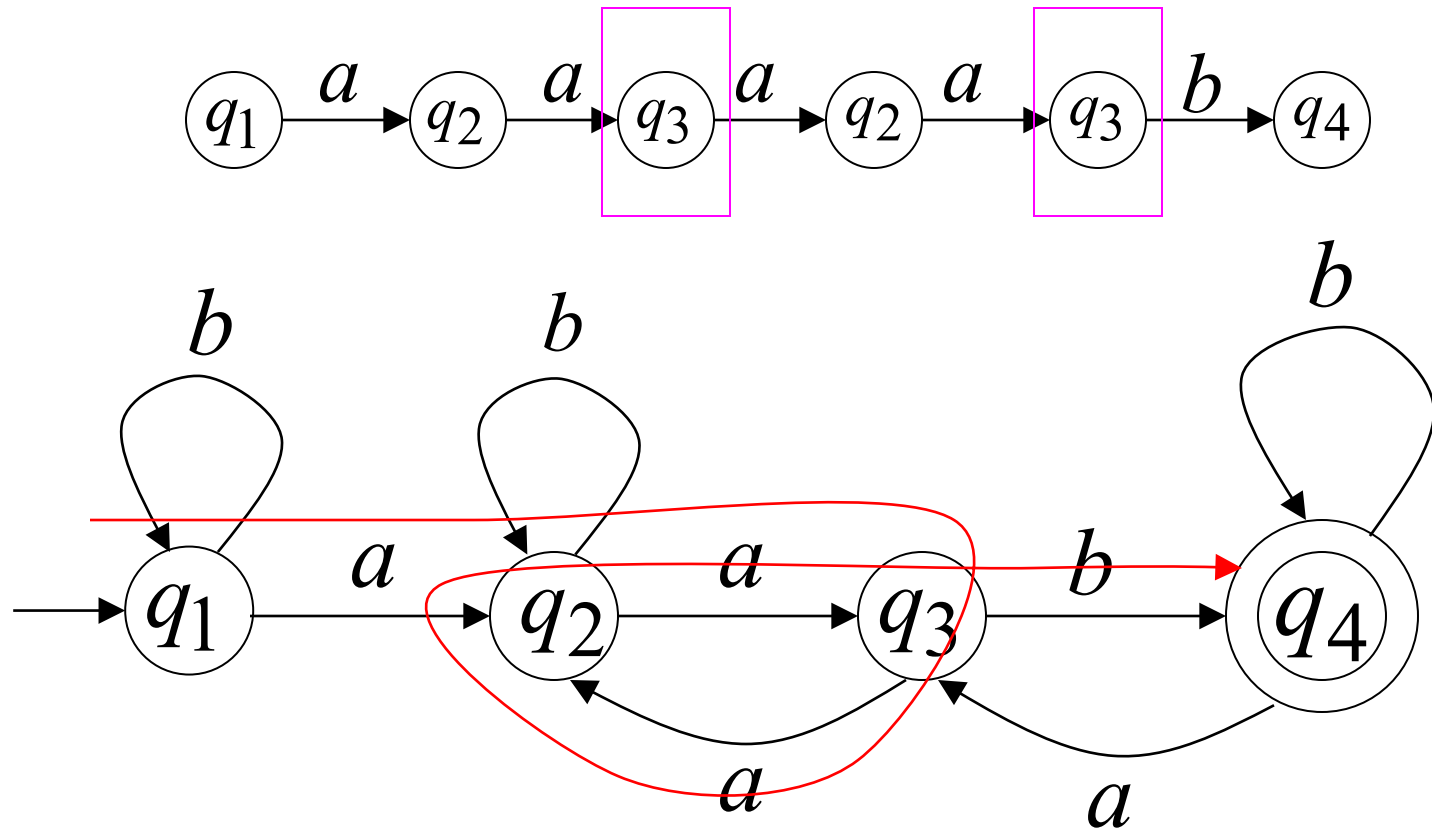
DFAs

Consider a DFA with 4 states



Consider the walk of a “long” string: $aaaaab$
(length at least 4)

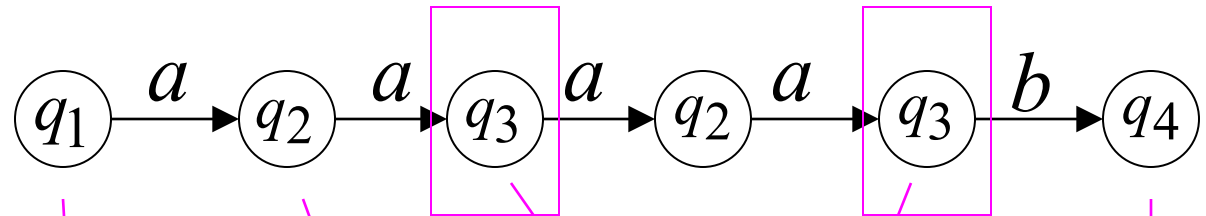
A state is repeated in the walk of $aaaaab$



The state is repeated as a result of the pigeonhole principle

Walk of $aaaaab$

Pigeons:
(walk states)



Are more than

Nests:
(Automaton states)

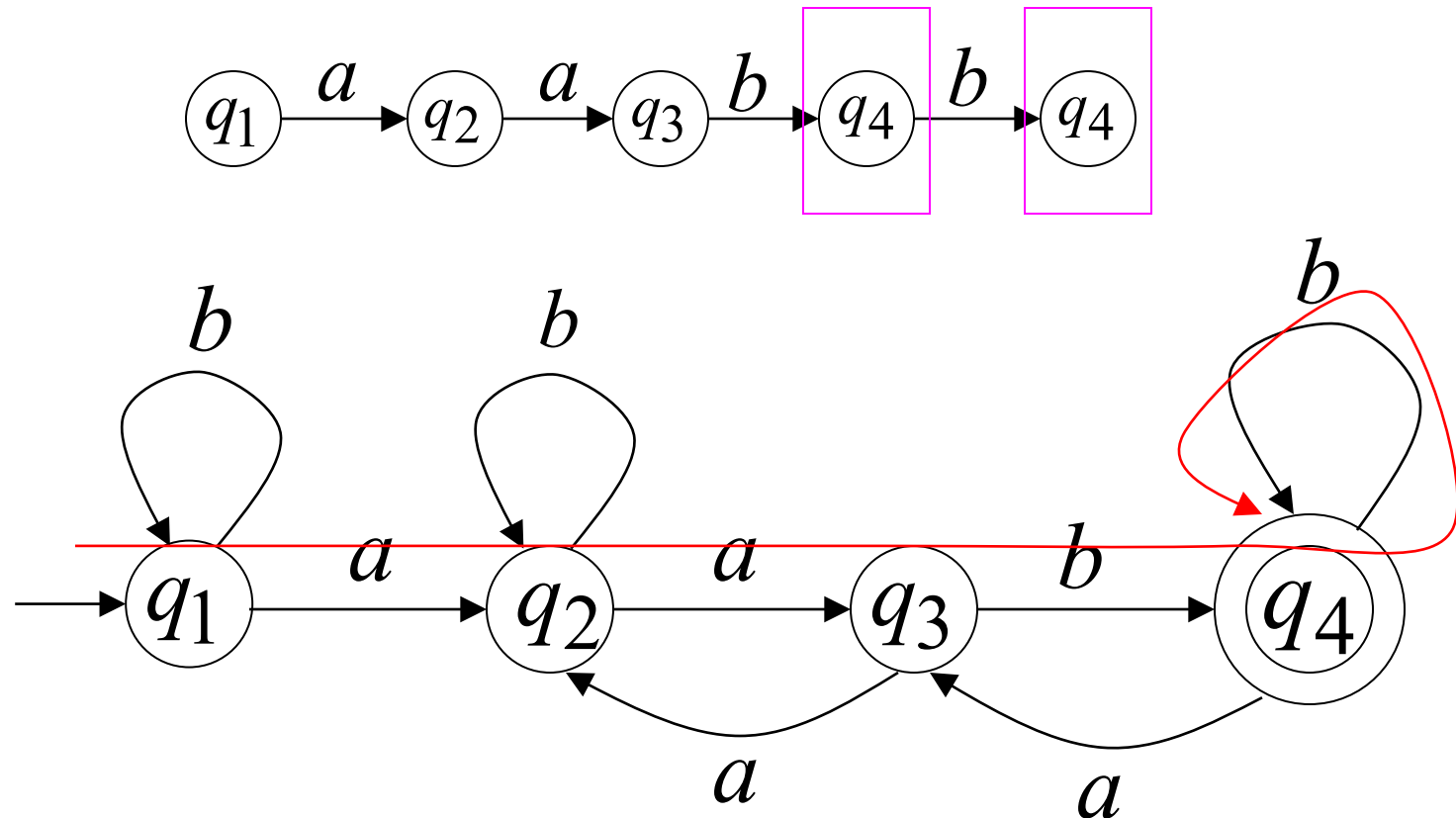


Repeated
state

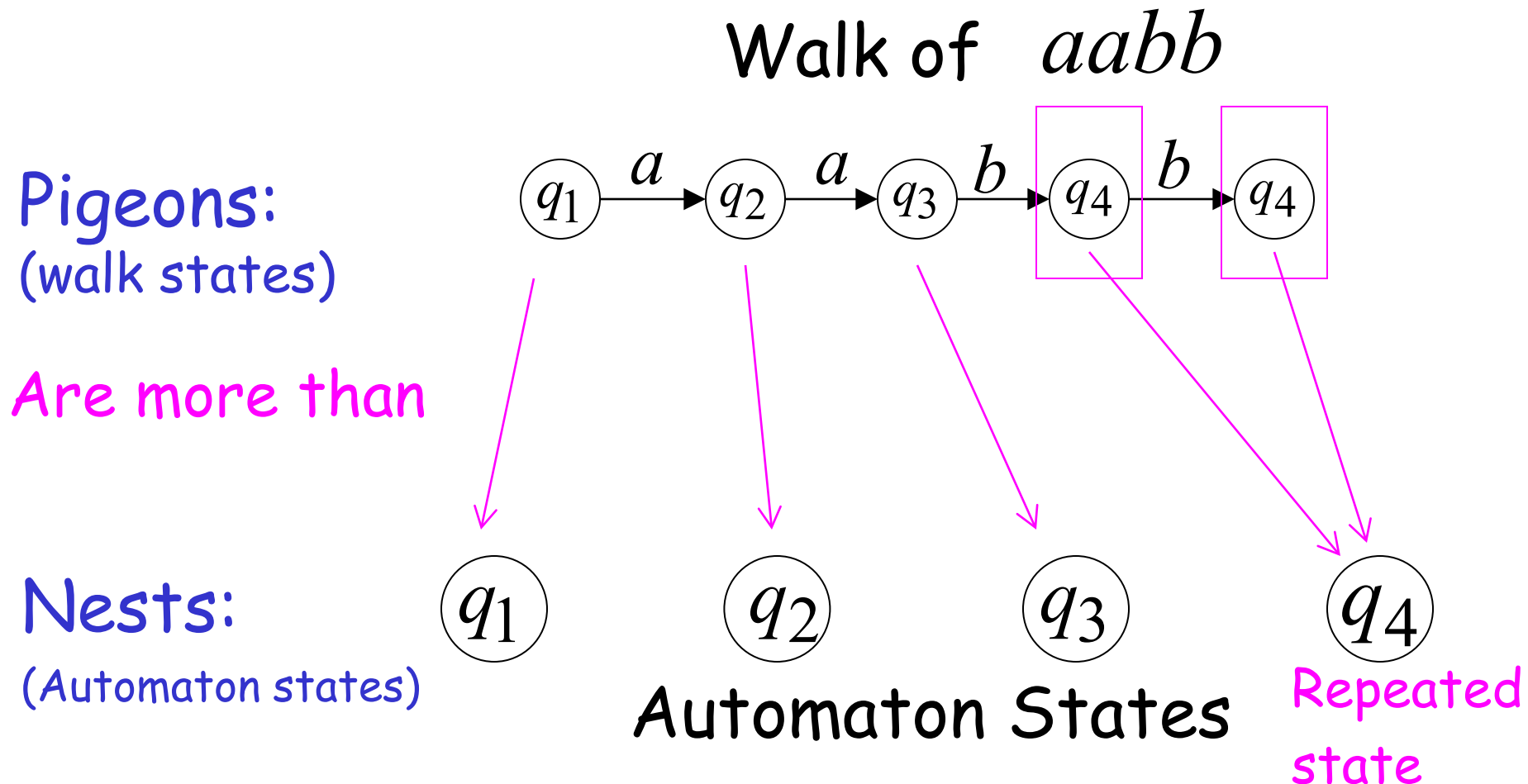
Consider the walk of a “long” string: $aabb$
(length at least 4)

Due to the pigeonhole principle:

A state is repeated in the walk of $aabb$

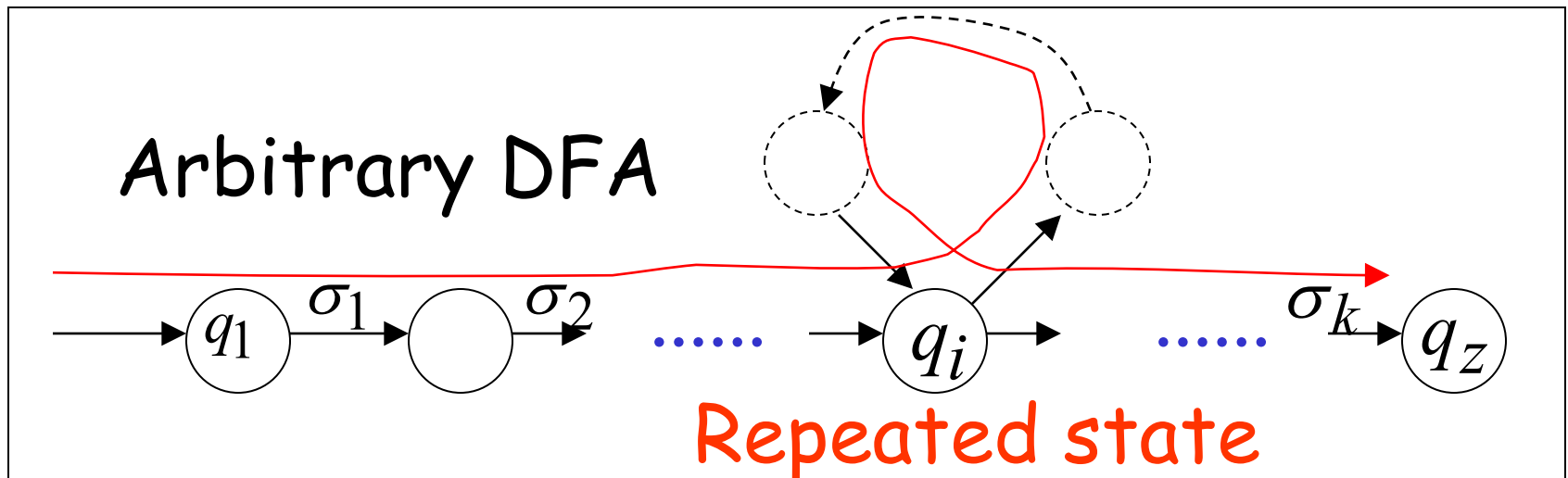
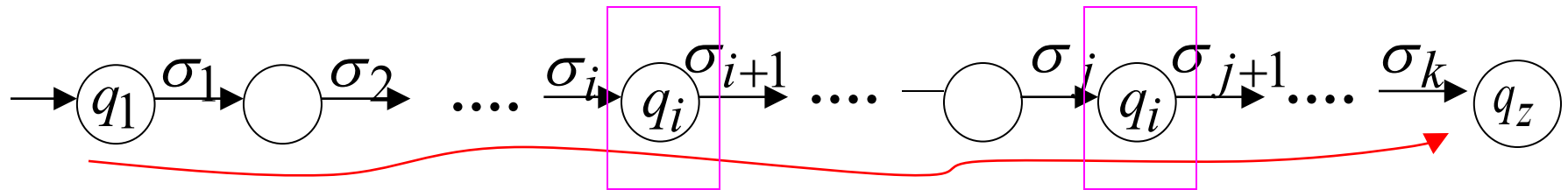


The state is repeated as a result of the pigeonhole principle

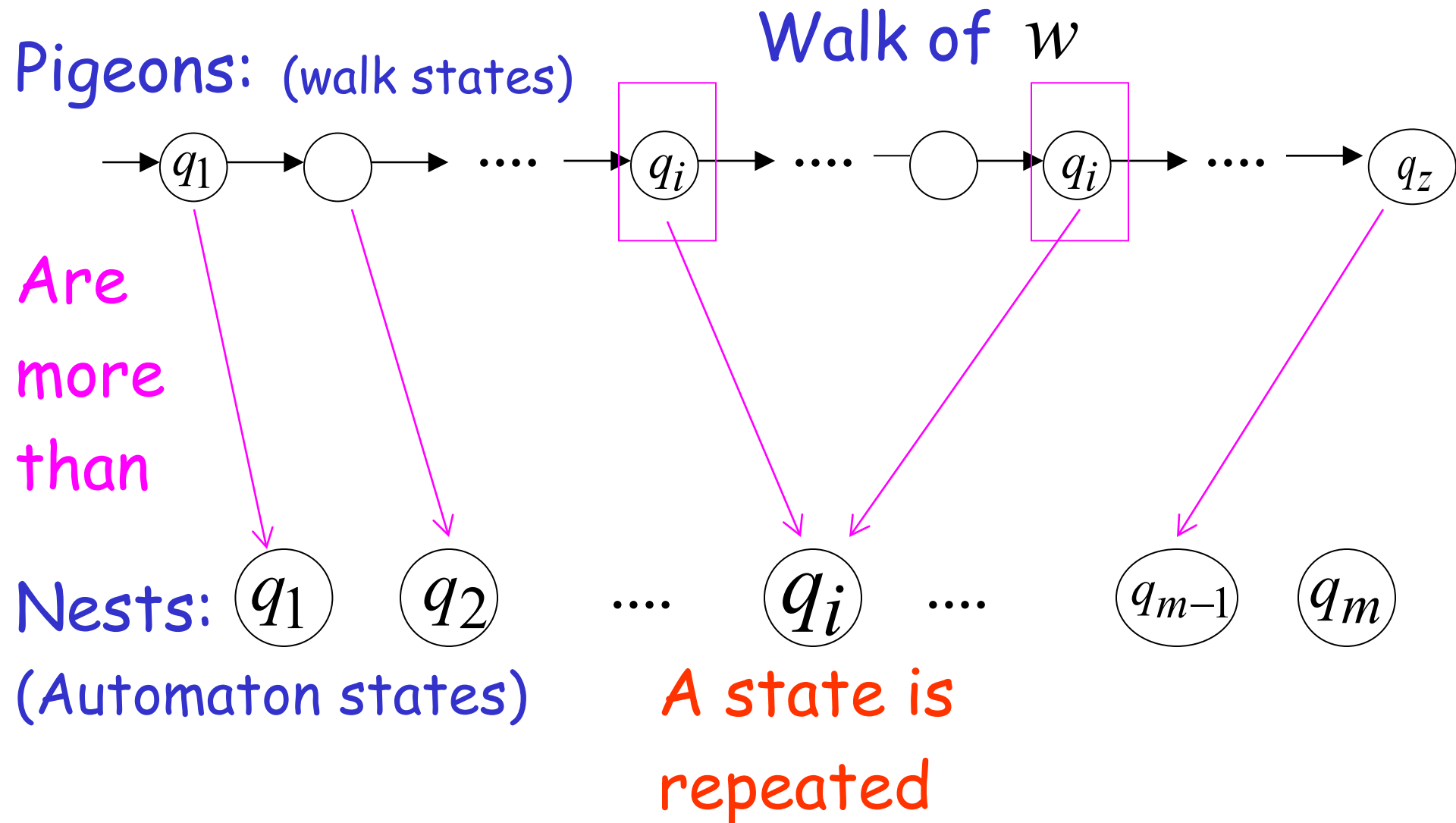


In General: If $|w| \geq \# \text{states of DFA}$,
by the pigeonhole principle,
a state is repeated in the walk w

Walk of $w = \sigma_1 \sigma_2 \cdots \sigma_k$



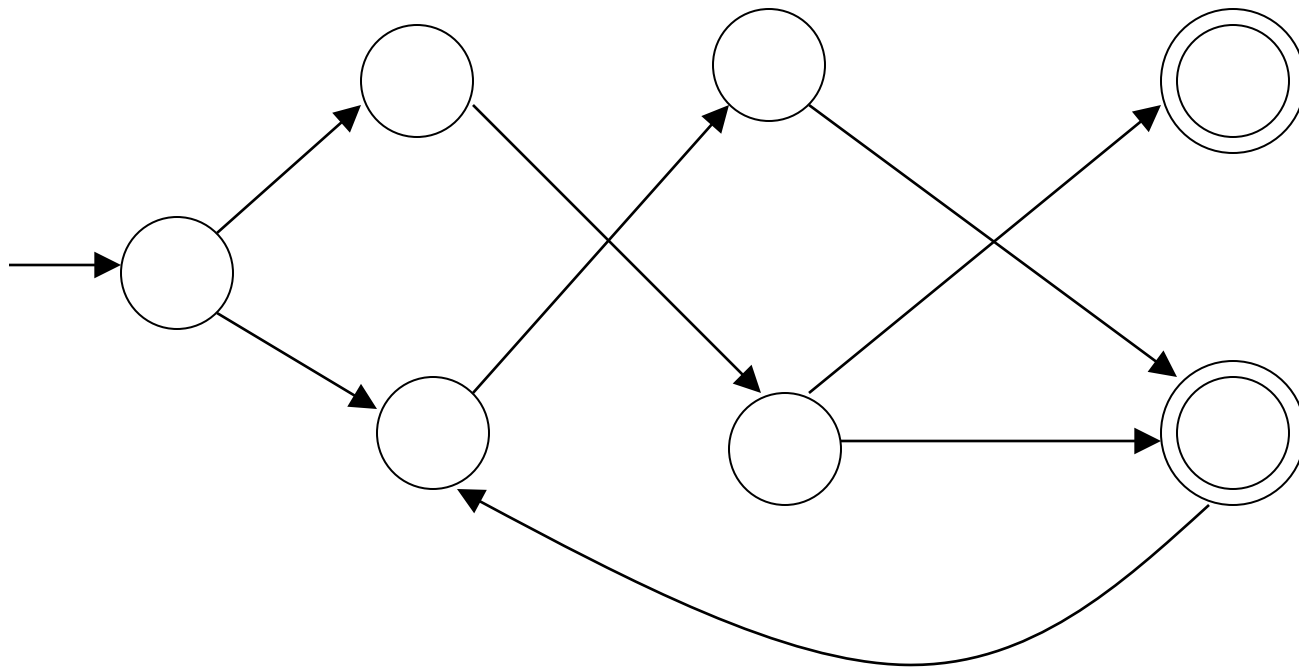
$$|w| \geq \# \text{states of DFA} = m$$



The Pumping Lemma

Take an **infinite** regular language L
(contains an infinite number of strings)

There exists a DFA that accepts L

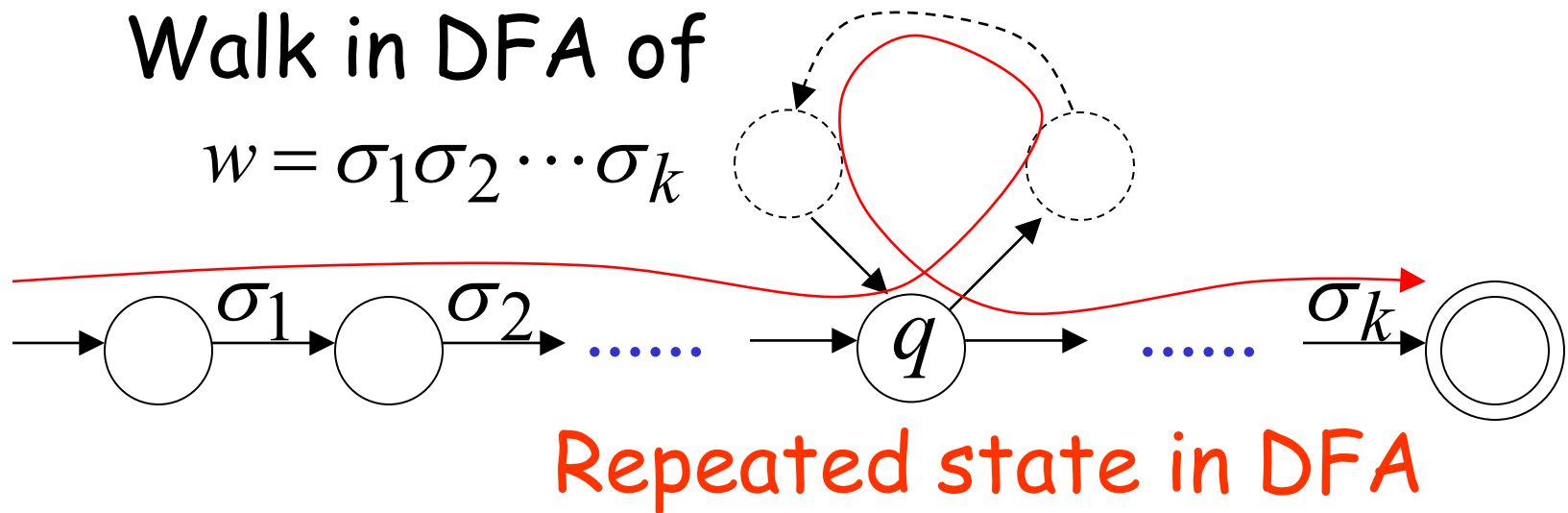


m
states

Take string $w \in L$ with $|w| \geq m$

(number of
states of DFA)

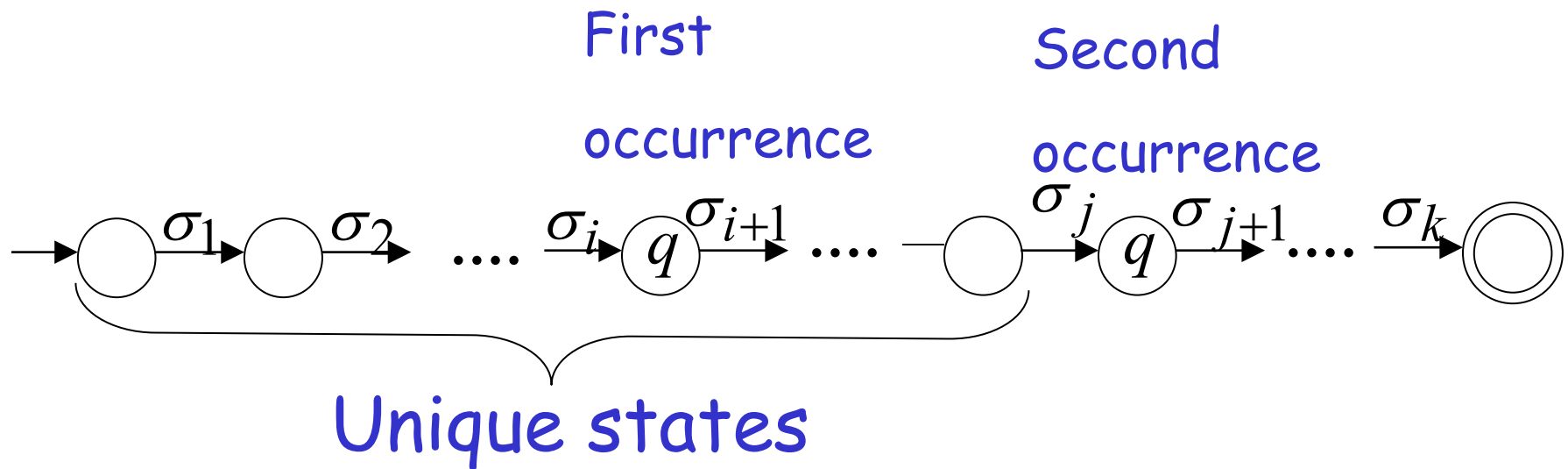
then, at least one state is repeated
in the walk of w



There could be many states repeated

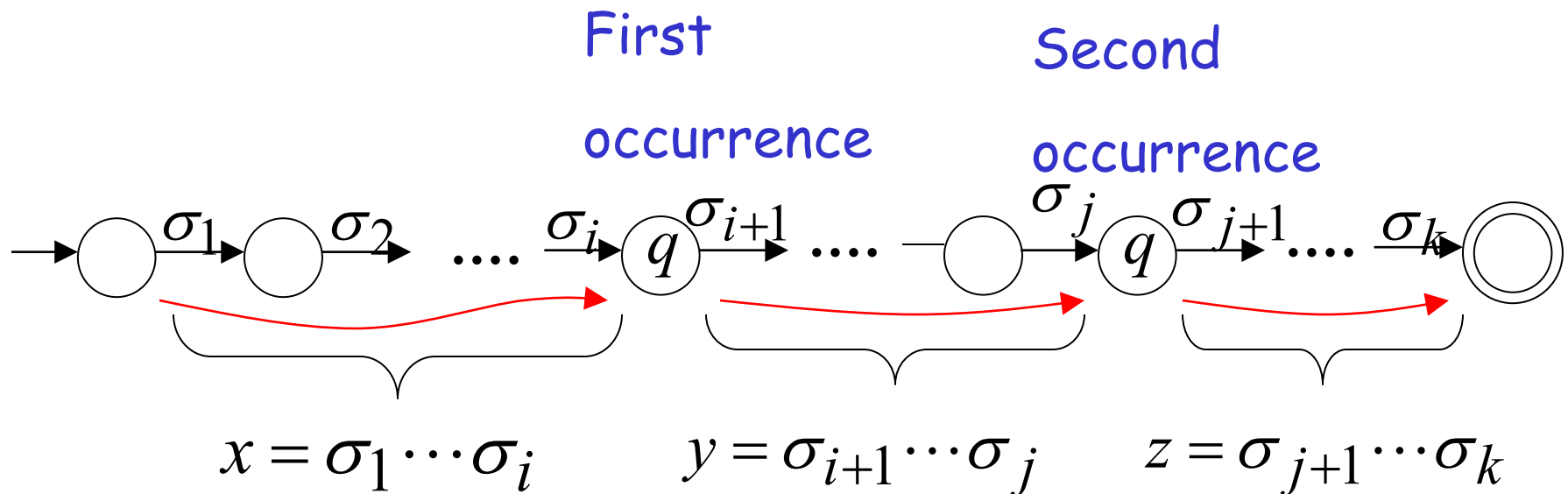
Take q to be the first state repeated

One dimensional projection of walk w :



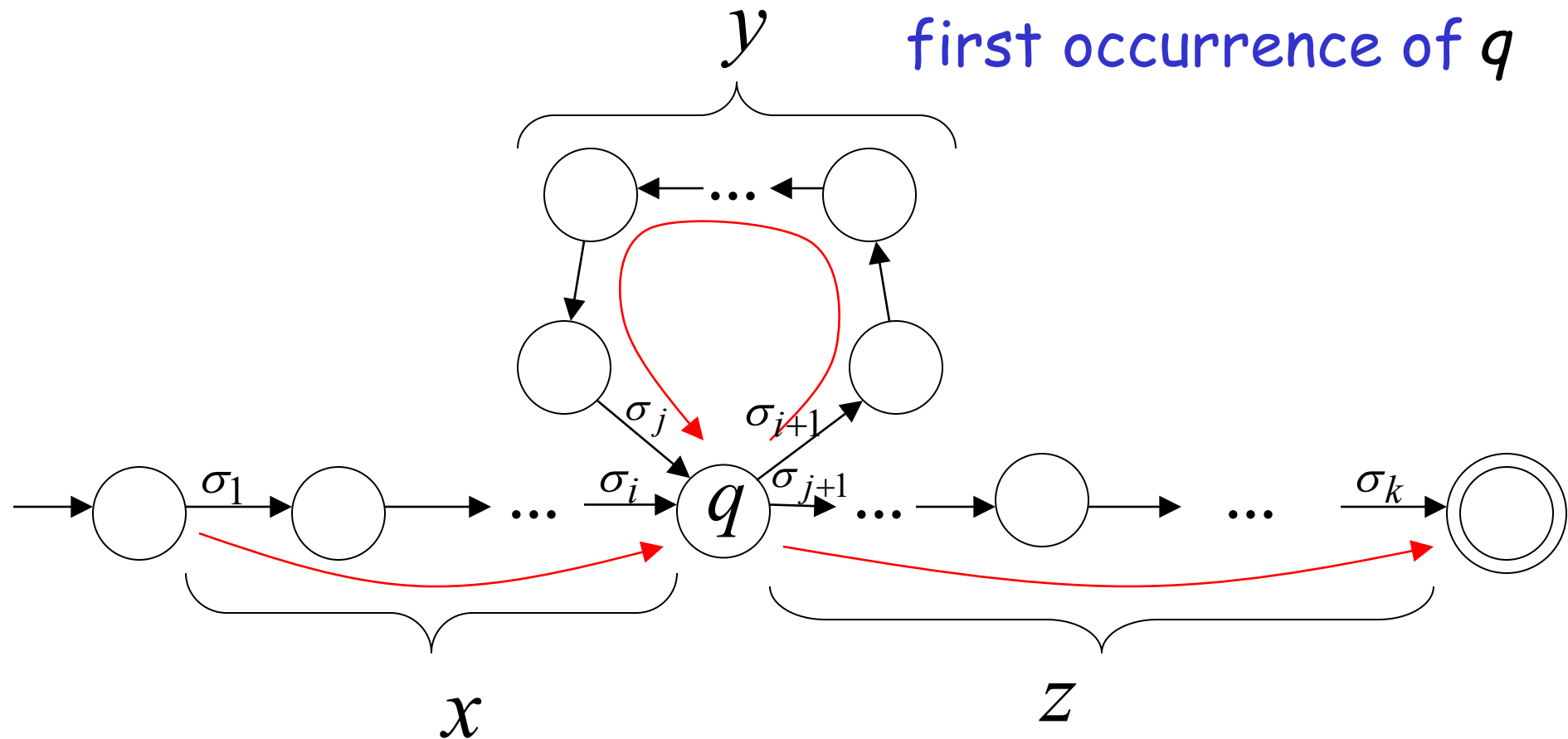
We can write $w = xyz$

One dimensional projection of walk w :

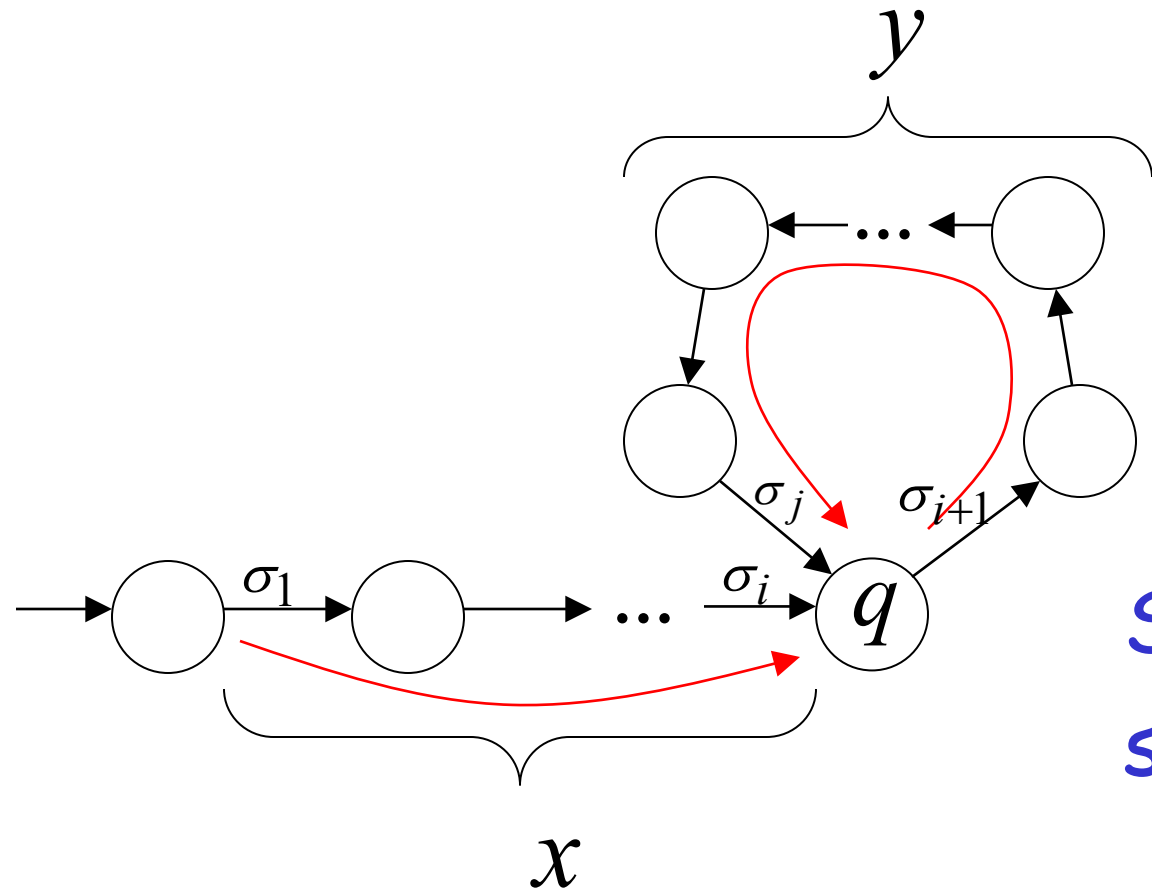


In DFA: $w = x y z$

contains only
first occurrence of q



Observation: $\text{length } |x y| \leq m$ number of states of DFA

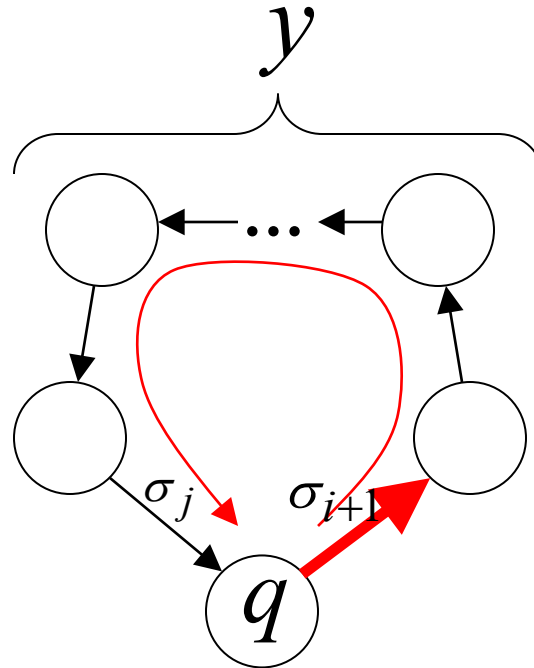


Unique States

Since, in xy no state is repeated (except q)

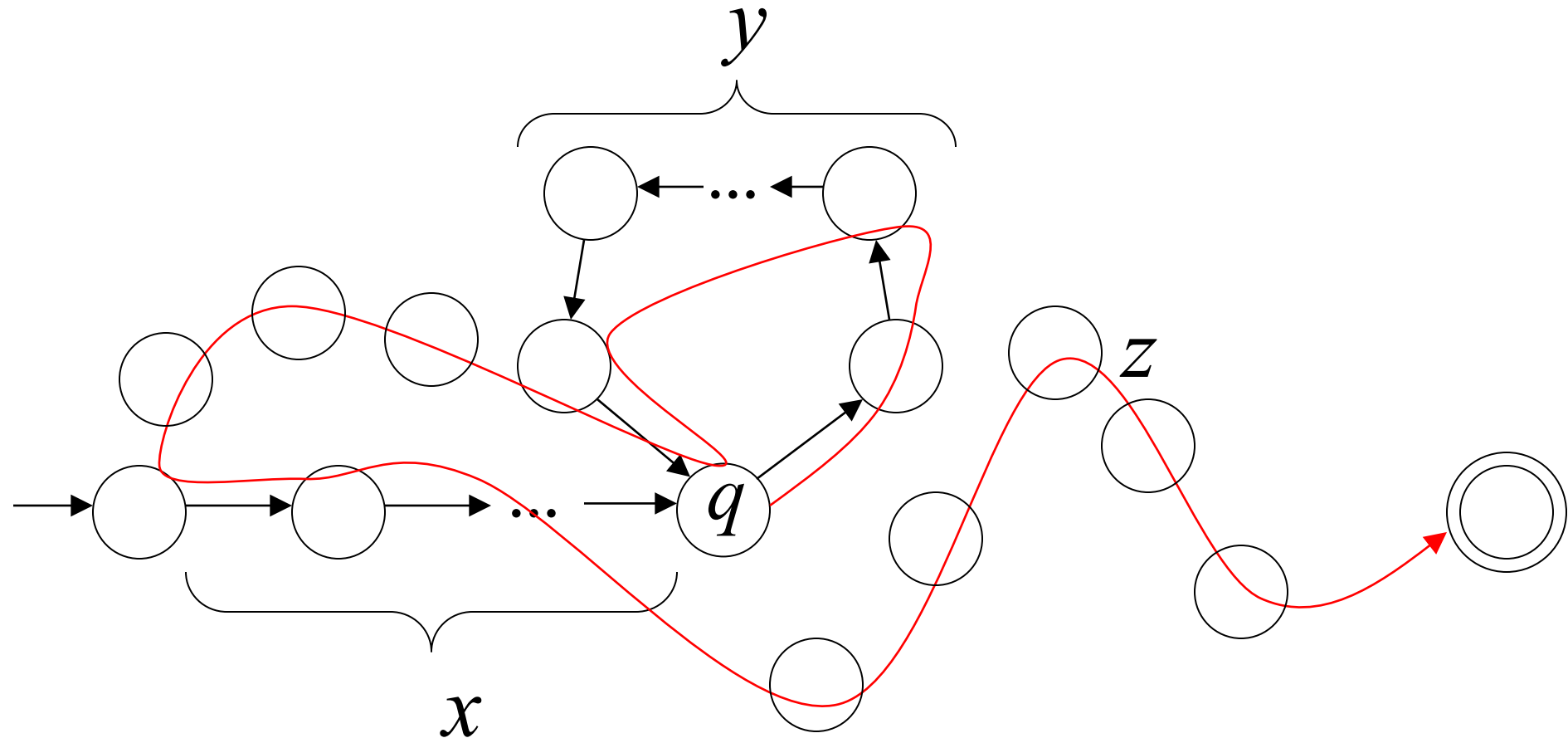
Observation: $\text{length } |y| \geq 1$

Since there is at least one transition in loop



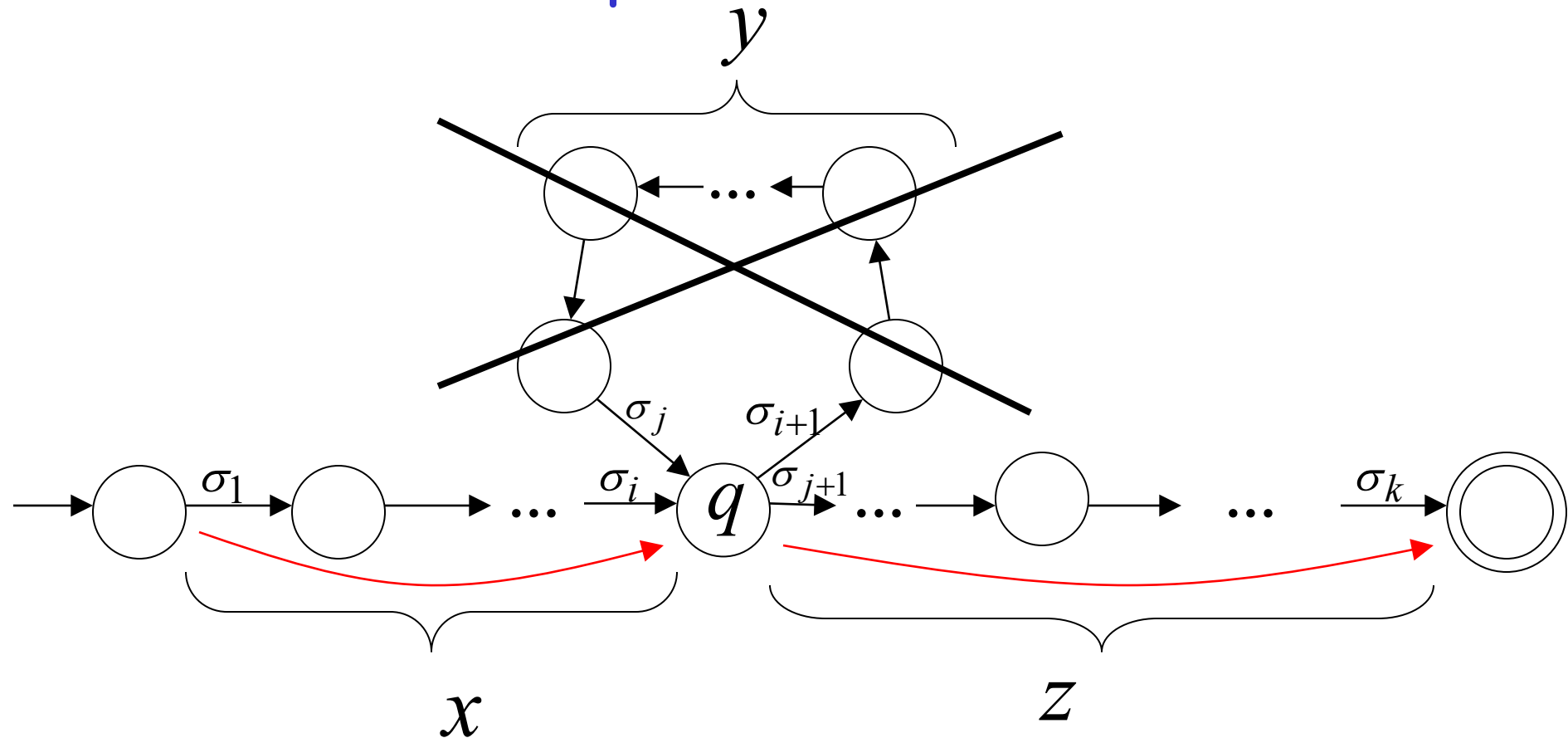
We do not care about the form of string z

z may actually overlap with the paths of x and y



Additional string: The string xz
is accepted

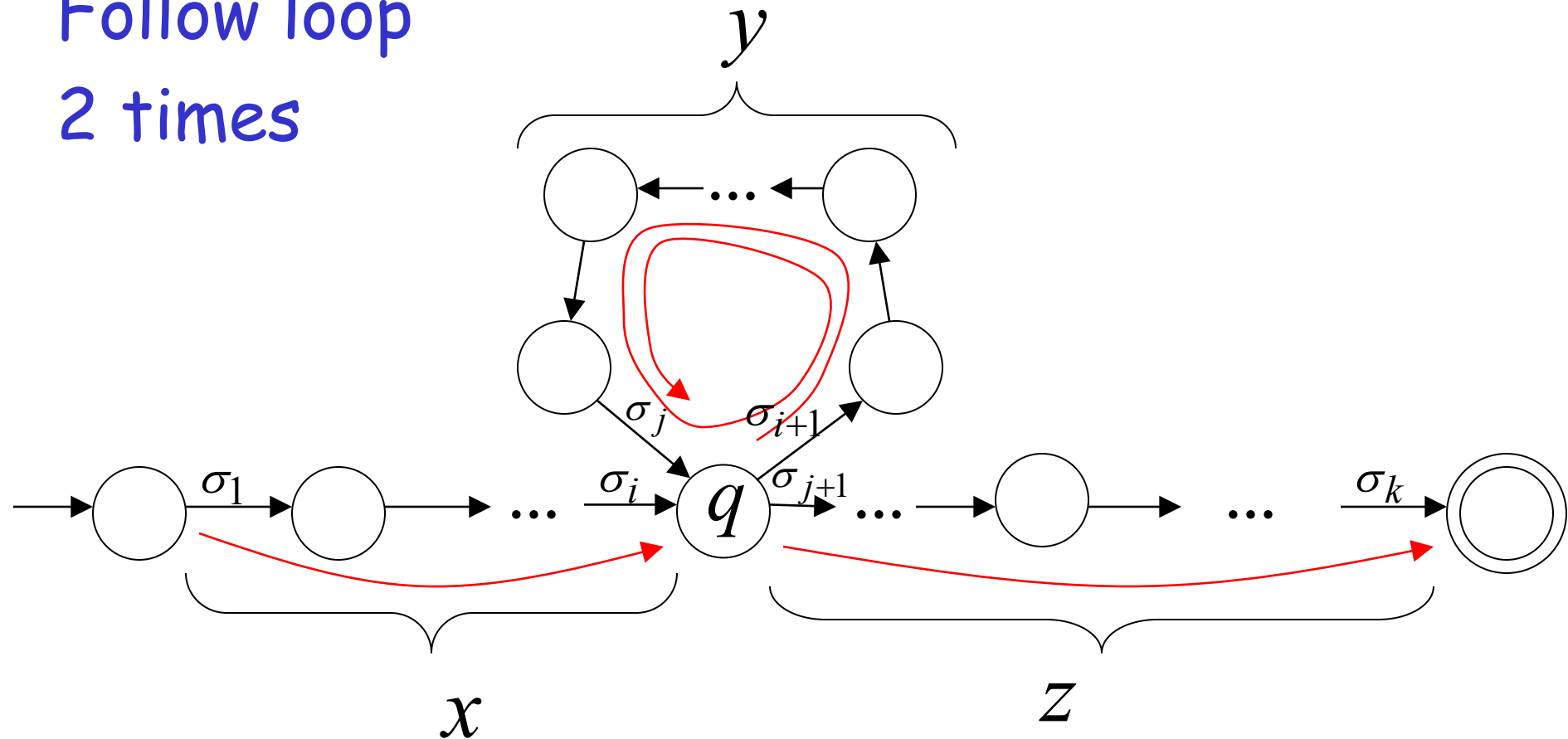
Do not follow loop



Additional string:

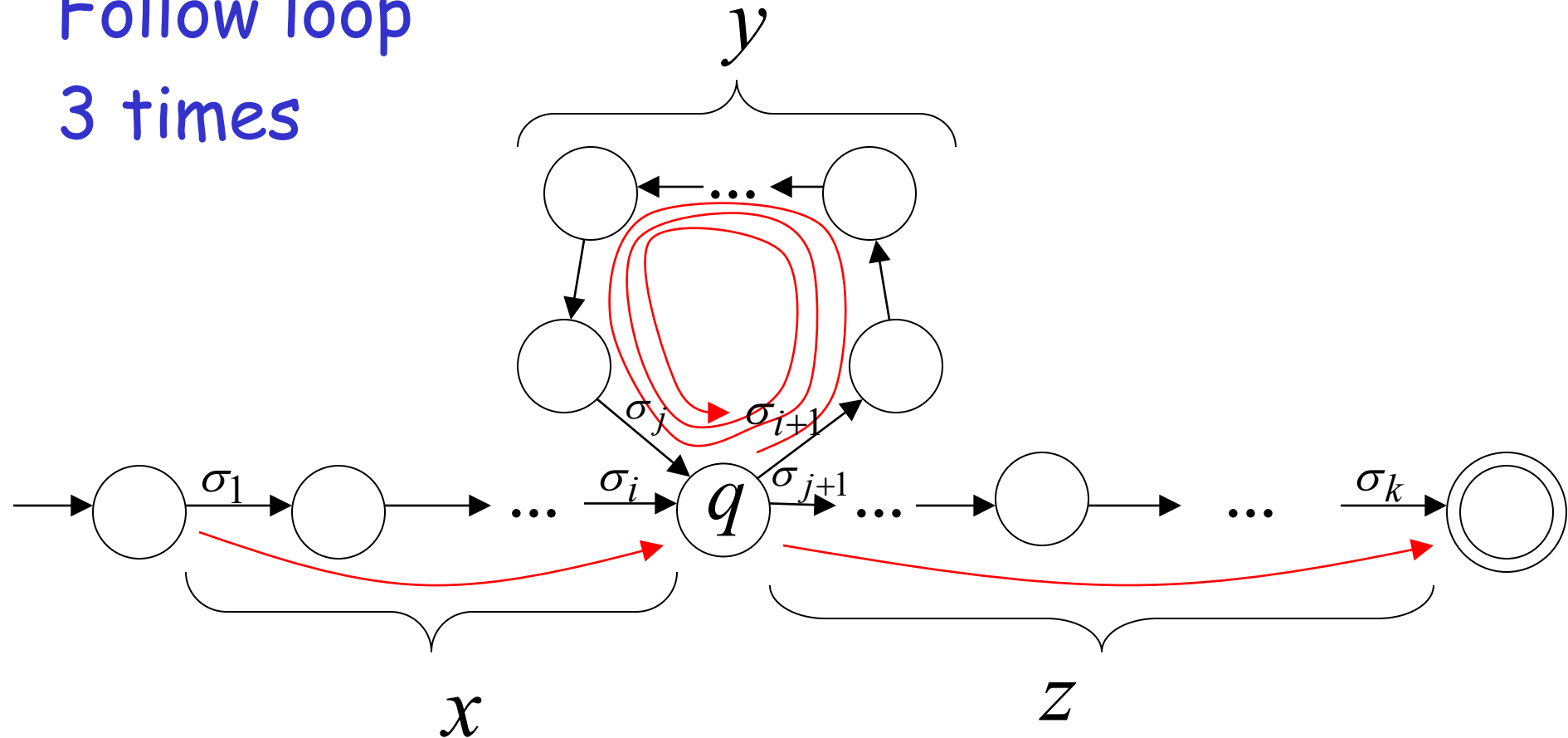
The string $x y y z$
is accepted

Follow loop
2 times



Additional string: The string $x y y y z$
is accepted

Follow loop
3 times



In General:

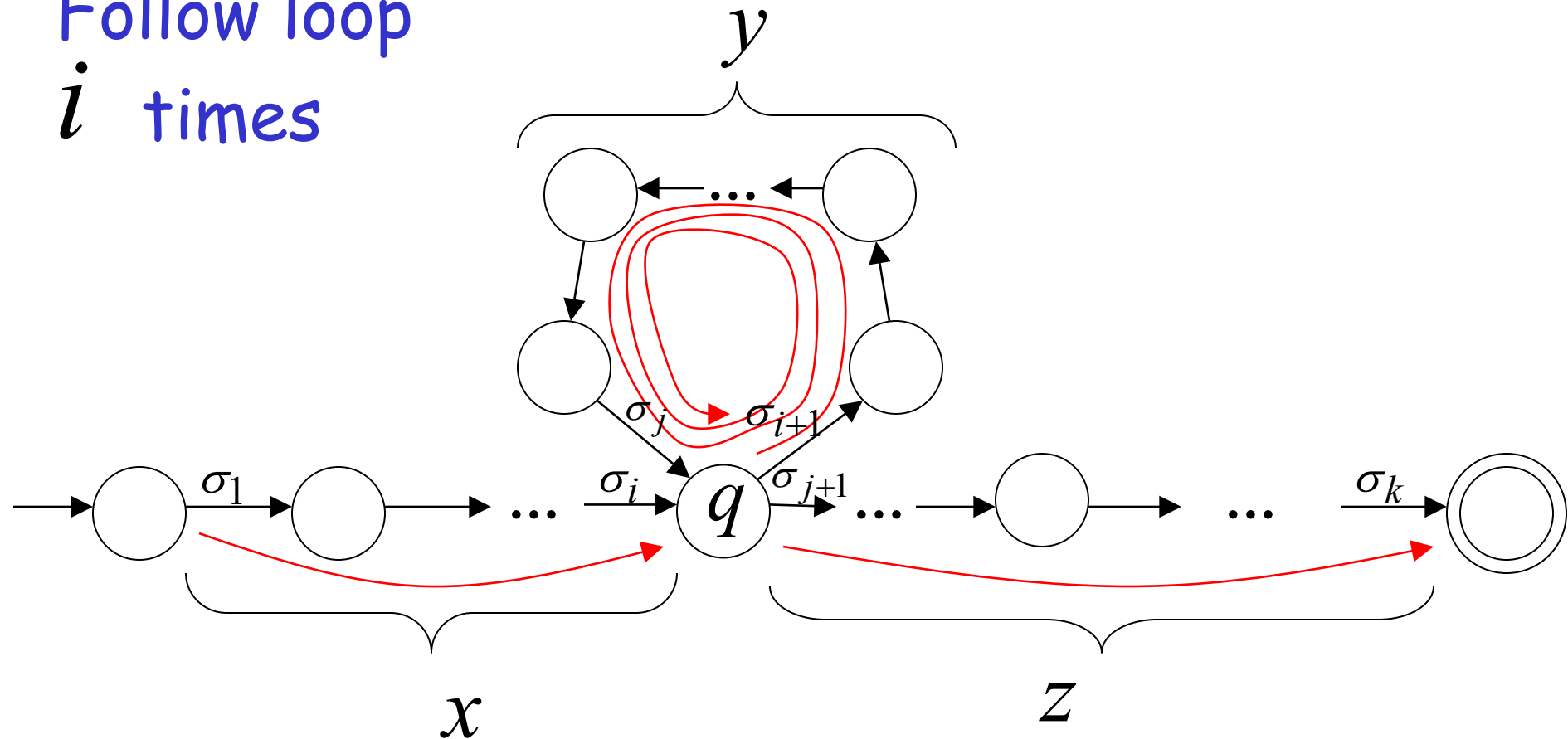
The string

$x y^i z$

is accepted

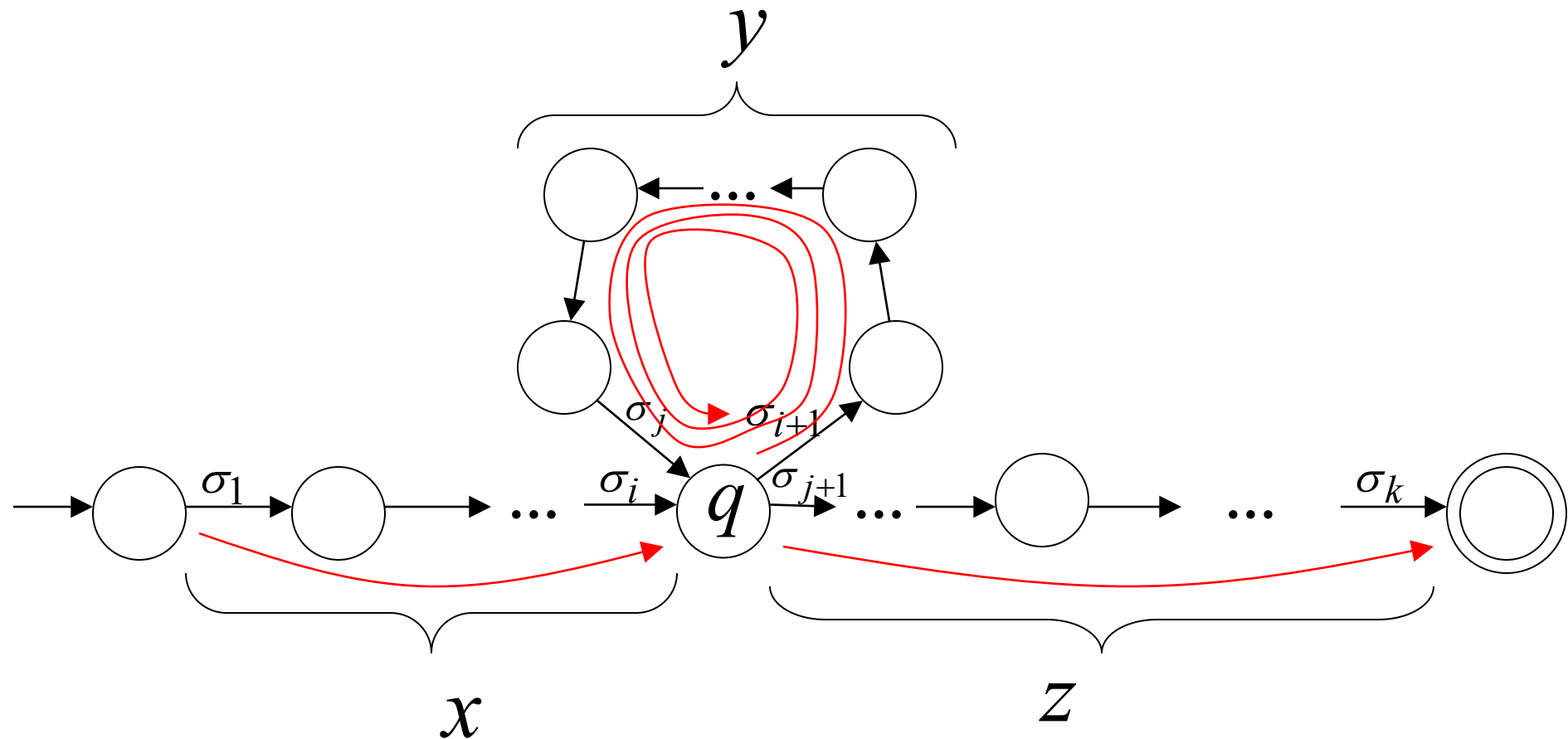
$i = 0, 1, 2, \dots$

Follow loop
 i times

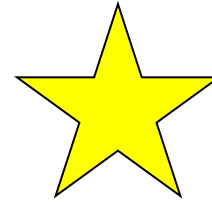
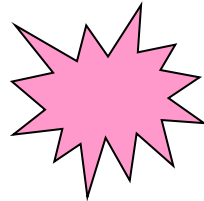


Therefore: $x y^i z \in L \quad i = 0, 1, 2, \dots$

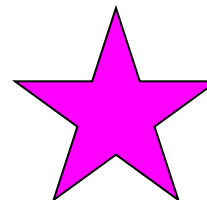
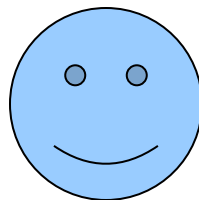
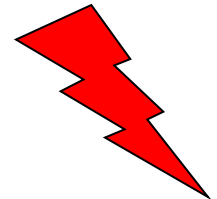
Language accepted by the DFA



In other words, we described:



The Pumping Lemma !!!



The Pumping Lemma:

- Given a infinite regular language L
- there exists an integer m (critical length)
- 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: $x y^i z \in L \quad i = 0, 1, 2, \dots$

In the book:

Critical length m = Pumping length p

Applications of the Pumping Lemma

Observation:

Every language of finite size has to be regular

(we can easily construct an NFA
that accepts every string in the language)

Therefore, every non-regular language
has to be of infinite size

(contains an infinite number of strings)

Suppose you want to prove that
An infinite language L is not regular

1. Assume the opposite: L is regular
2. The pumping lemma should hold for L
3. Use the pumping lemma to obtain a contradiction
4. Therefore, L is not regular

Explanation of Step 3: How to get a contradiction

1. Let m be the critical length for L
2. Choose a particular string $w \in L$ which satisfies the length condition $|w| \geq m$
3. Write $w = xyz$
4. Show that $w' = xy^i z \notin L$ for some $i \neq 1$
5. This gives a contradiction, since from pumping lemma $w' = xy^i z \in L$

Note: It suffices to show that
only one string $w \in L$
gives a contradiction

You don't need to obtain
contradiction for every $w \in L$

Example of Pumping Lemma application

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 critical length for L

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

and length $|w| \geq m$

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

From the Pumping Lemma:

we can write $w = a^m b^m = x y z$

with lengths $|x y| \leq m, |y| \geq 1$

$$w = xyz = a^m b^m = \underbrace{a \dots a}_{x} \underbrace{a \dots a}_{y} \underbrace{a \dots a b \dots b}_{z}$$

The diagram illustrates the decomposition of the string $w = a^m b^m$ into xyz . The string is represented as $a \dots a a \dots a a \dots a b \dots b$. A green bracket above the string indicates the total length m for the a segment and another green bracket indicates the total length m for the b segment. Red brackets below the string indicate the decomposition into x , y , and z . x is a substring of a 's, y is a substring of a 's, and z contains the remaining a 's and all b 's.

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

$$x y z = a^m b^m$$

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

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

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

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

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

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

$$xy^2z = \overbrace{a \dots a a \dots a a \dots a a \dots a}^{m+k} \overbrace{b \dots b}^m \in L$$

$\underbrace{\hspace{1.5cm}}_x \quad \underbrace{\hspace{1.5cm}}_y \quad \underbrace{\hspace{1.5cm}}_y \quad \underbrace{\hspace{3.5cm}}_z$

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

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

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



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

CONTRADICTION!!!

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

Conclusion: L is not a regular language

END OF PROOF

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

Regular languages

$$L(a^* b^*)$$