BU CS 332 – Theory of Computation

Lecture 4:

- Non-regular languages
- Pumping Lemma

Reading:

Sipser Ch 1.4

Ran Canetti

September 15, 2020

Recall: Operations on languages

Let $A, B \subseteq \Sigma^*$ be languages. Define

Regular Operations

- Union: $A \cup B$
- Concatenation: $A \circ B = \{ab \mid a \in A, b \in B\}$
- Star: $A^* = \{ a_1 a_2 \ldots a_n \mid n \geq 0 \text{ and } a_i \in A\}$

- Complement: $\bar{A}$
- Intersection: $A \cap B$
- Reverse: $A^R = \{ a_1 a_2 \ldots a_n \mid a_n \ldots a_1 \in A\}$

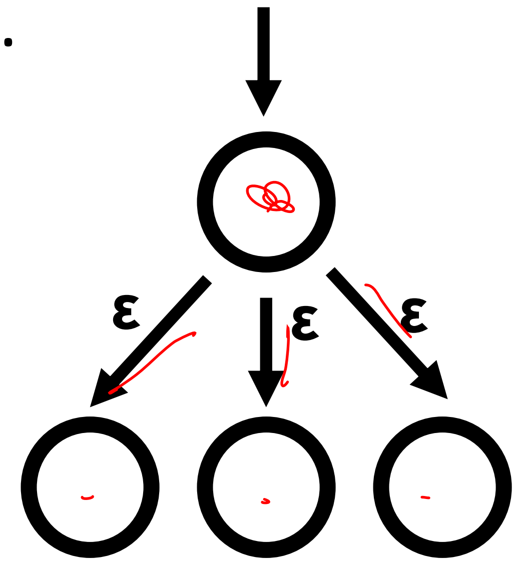
Theorem: The class of regular languages is closed under all six of these operations

Closure under Reverse

Theorem. The reverse of a regular language is also regular

Proof: Let L be a regular language and M be a DFA recognizing it. Construct an NFA M' recognizing L^R:

- Define M' as M with the arrows reversed.
- Make the start state of M be the accept state in M'.
- Make a new start state that goes to all accept states of M by ε-transitions.



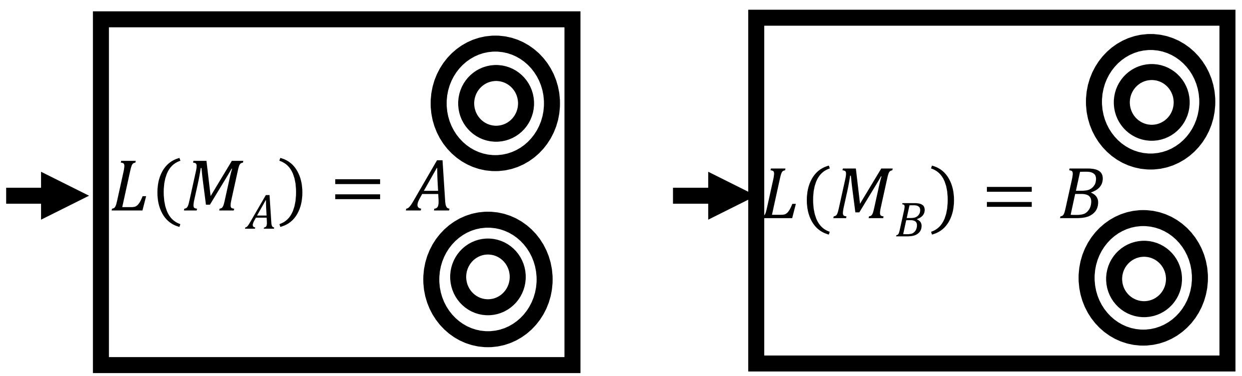
Closure under Concatenation

Concatenation: $A \circ B = \{ ab \mid a \in A \text{ and } b \in B \}$

Theorem. If A and B are regular, $A \circ B$ is also regular.

Proof: Given DFAs M_A and M_B, construct NFA by

- Connecting all accept states in M_A to the start state in M_B.
- Make all states in M_A non-accepting.



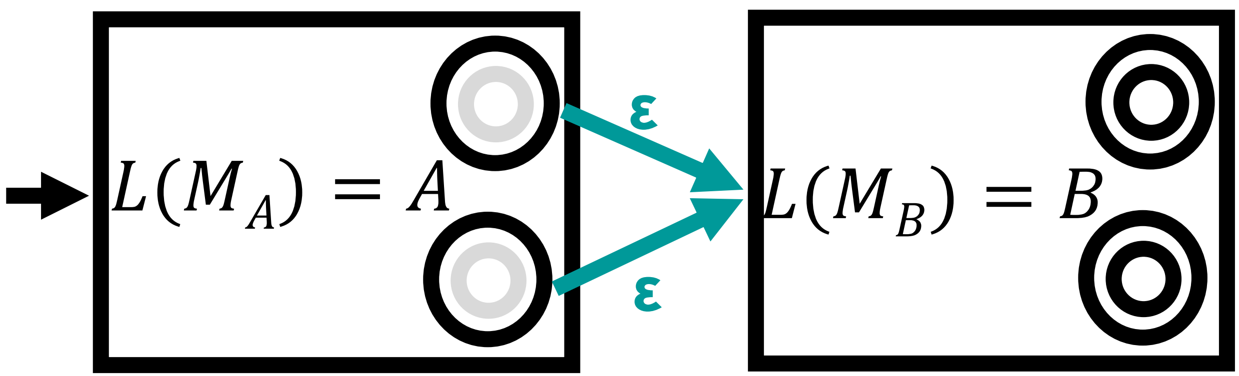
Closure under Concatenation

Concatenation: $A \circ B \;=\; \{\, ab \mid a \in A \text{ and } b \in B \,\}$

Theorem. If A and B are regular, $A \circ B$ is also regular.

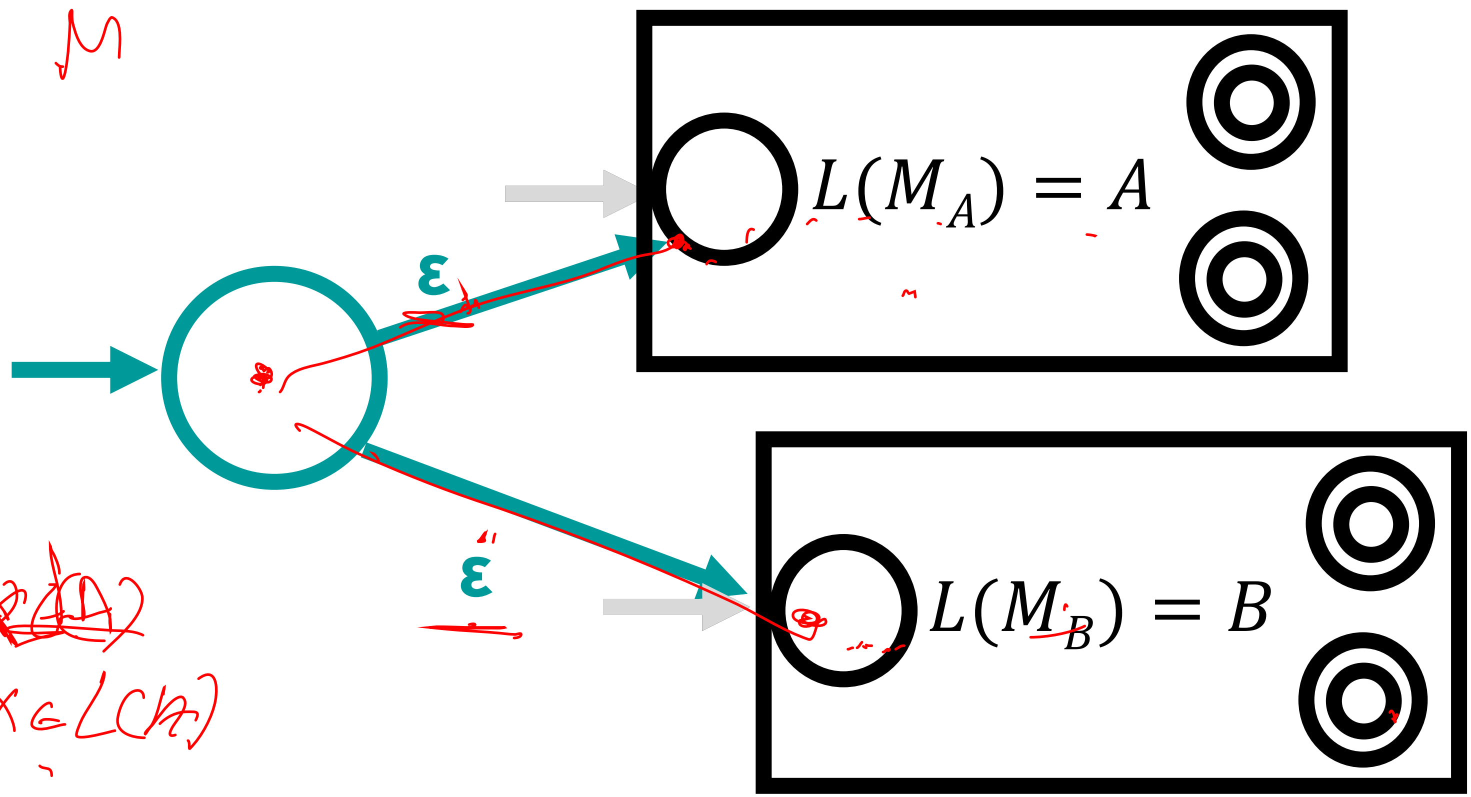
Proof: Given DFAs M_A and M_B, construct NFA by

- Connecting all accept states in M_A to the start state in M_B.
- Make all states in M_A non-accepting.



A Mystery Construction

Given DFAs M_A recognizing A and M_B recognizing B, what does the following NFA recognize?



~~$x \in L(A)$~~

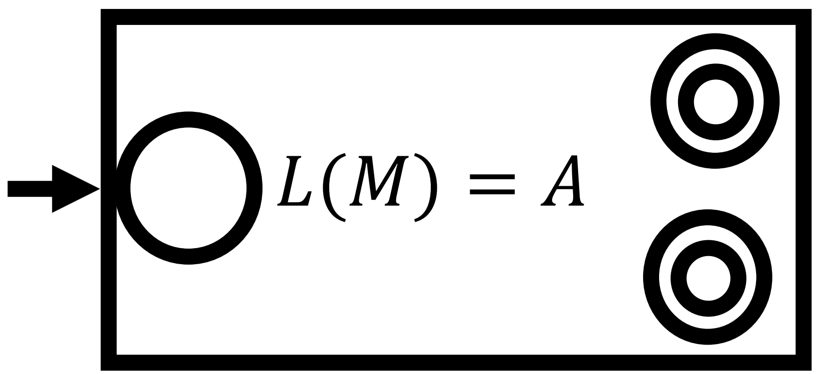
$x \in L(A)$

$x \in L(M) \Rightarrow x \in L(B)$

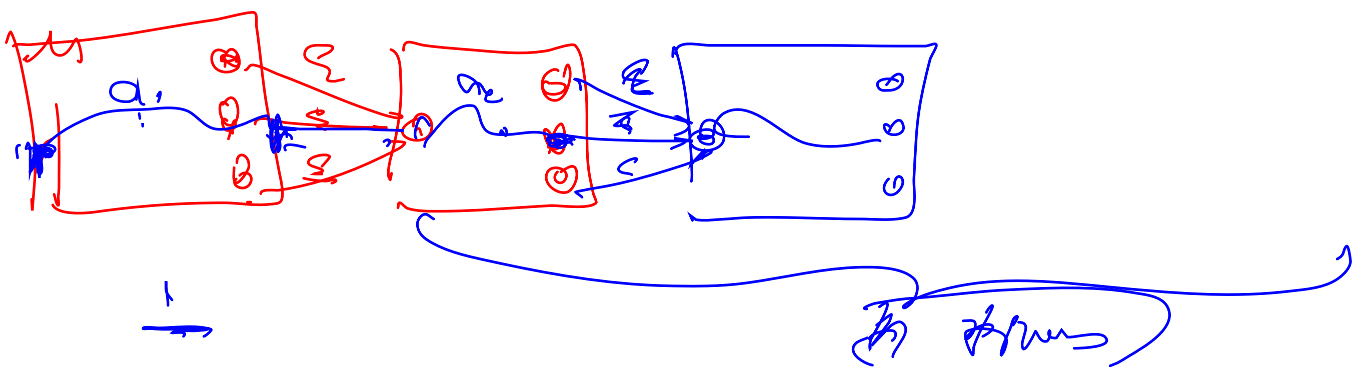
Closure under n-times concatenation

n-times: $A^n = \{\, a_1 a_2 \ldots a_n \mid \text{and } a_i \in A\}$

Theorem. If A is regular, A^n is also regular.



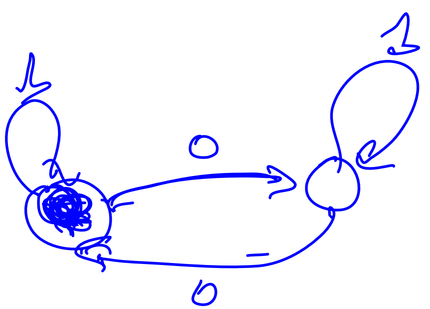
M^3

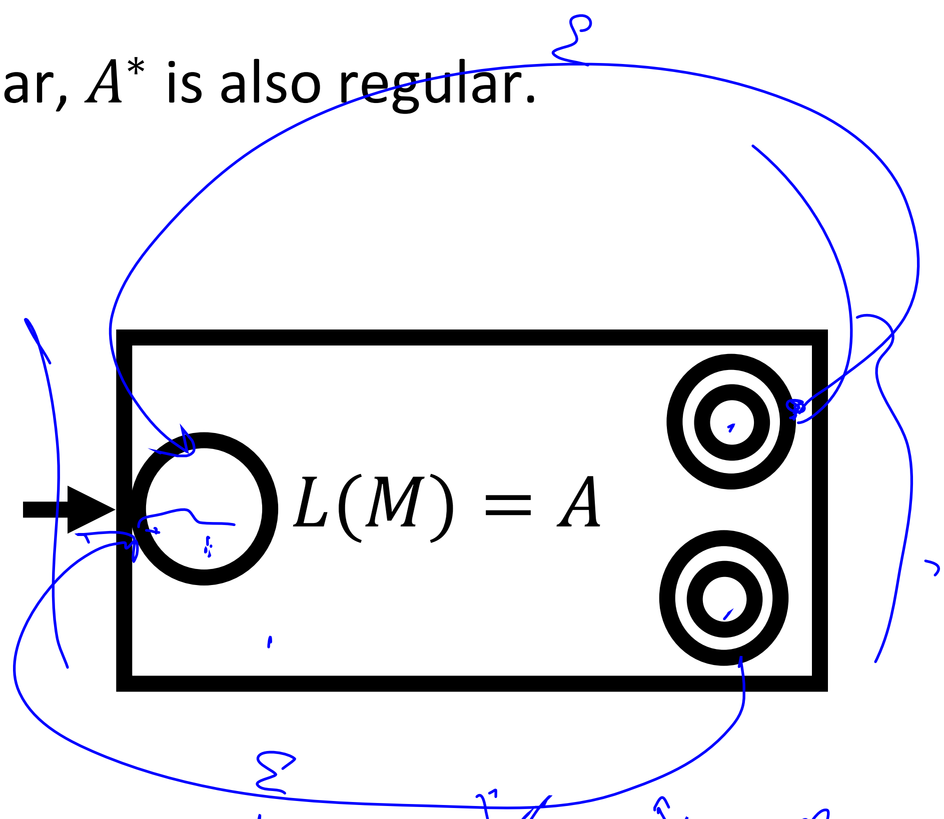


Closure under Star

Star: $A^* = \cup_{n>0} \{A^n\}$

Theorem. If A is regular, A^* is also regular.

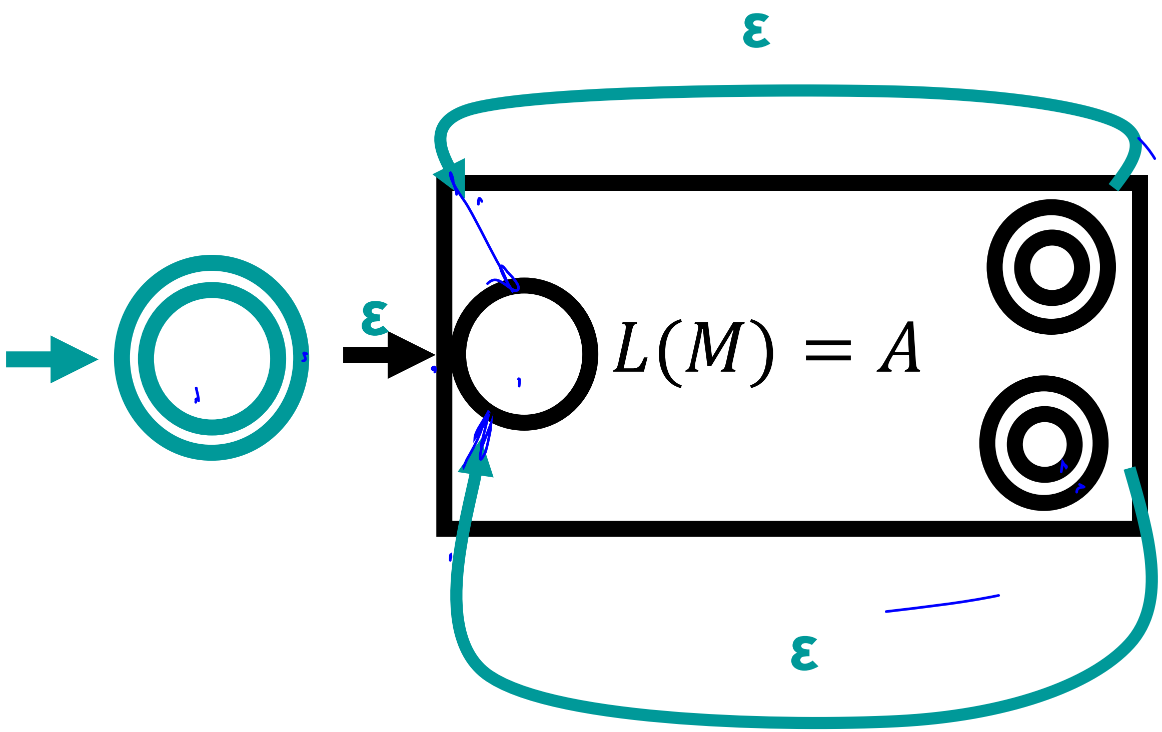




$x = x_1 \ldots x_n, \ \forall i \ x_i \in A$

Closure under Star

Theorem. If A is regular, A^* is also regular.



Closure under strict n-times concatenation

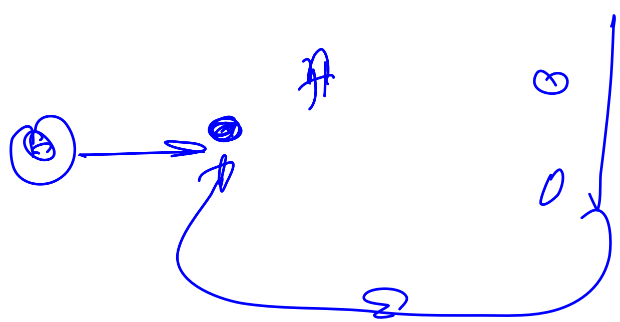
n-times:{ $\tilde{A}^n = \{ \overbrace{a \quad a \quad \ldots a}^{n \text{ times}} \mid$ and $a \quad \in A\}$

Is $\tilde{A}^*$ regular?

$\tilde{A}^* = \bigcup_{n \geq} \tilde{A}^n$

$\{\underbrace{a \ldots c}_{n} \mid n > 0 \quad a \in \tilde{A}\}$

ε



a_1, a_2

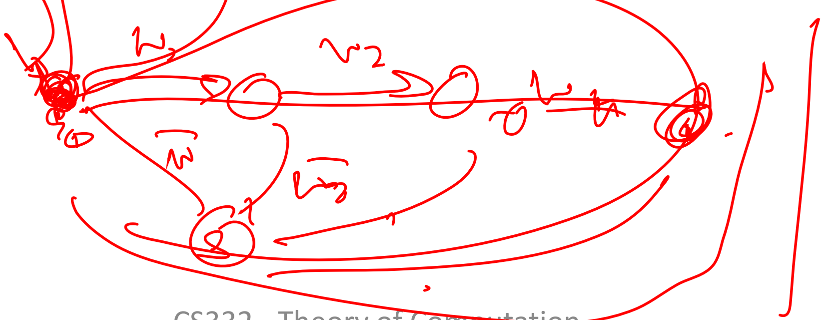
Continuing the exploration

- We’ve seen techniques for showing that languages are regular

- Could it be the case that every language is regular?

$w = w_1 \dots w_n$

M_w :



$L(M_w) = \{w\}$

Regular?

Construct an NFA for the following languages

- $\{0^n1^n \mid 0 < n \leq 2\}$

  ε, 01, 0011,

- $\{0^n1^n \mid 0 < n \leq k\}$

  ε, 01, 0011, 000111, … 0^k1^k,

- $\{0^n1^n \mid n > 0\}$

Proving a language is not regular

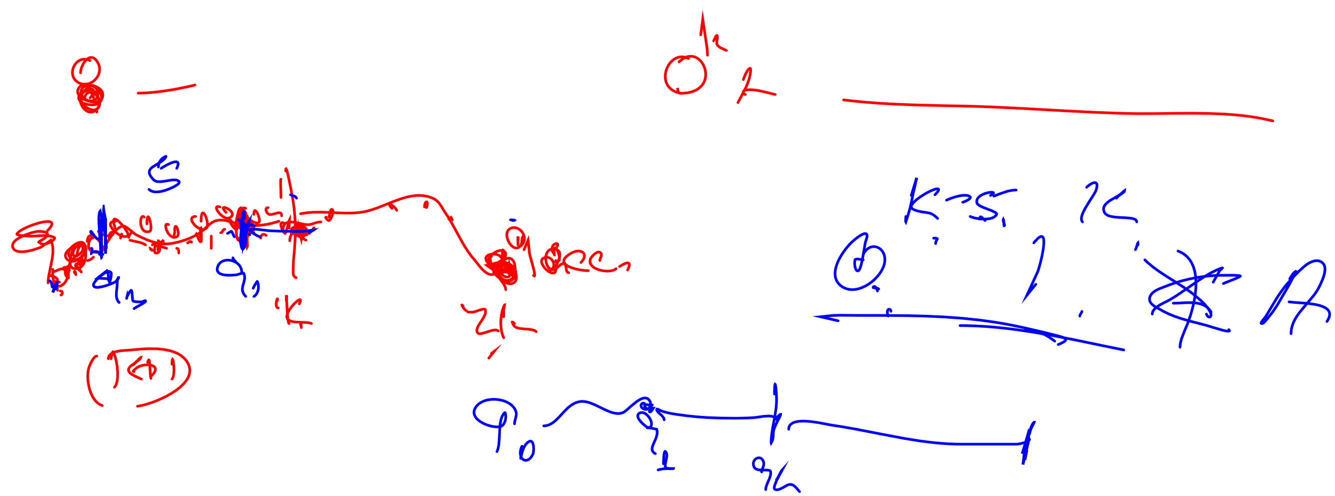
Theorem: $A = \{0^n1^n \mid n > 0\}$ is not regular

Proof: (by contradiction)

Let M be a DFA with k states recognizing A

Consider running M on input 0^k1^k

assume M accepts x.



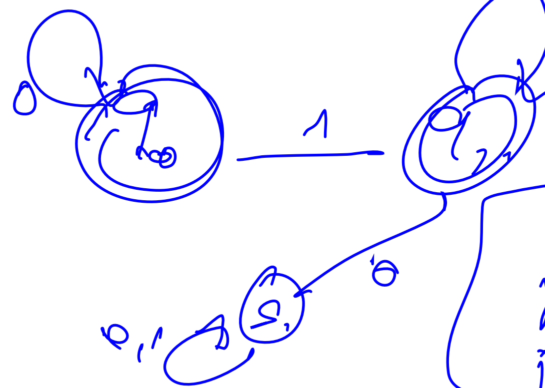
Regular or not?

A is not regular.
If E is reg. then A is reg.
$\Rightarrow$ C is not regular.

$C \;=\; \{\, w \mid w$ has equal number of 1s and 0s$\}$

$E = \{0^n 1^{n'} \mid n, n' \geq 0\}$

E is regular



$A = C \cap E$

regular

regular

regular

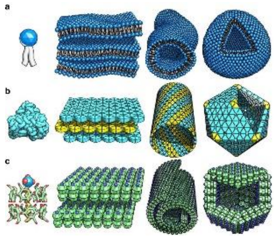
$D \;=\; \{\, w \mid w$ has equal number of 10s and 01s$\}$

The Pumping Lemma

A **systematic** way to prove that a language is not regular

Why do we teach this?

- Demonstrates how can prove negative results on the power of computers (computational models)
- Proof illuminates essential structure of finite automata
- Generalizes to other models of computation / classes of languages (CFLs, self-assembly)
- Applying it can be fun!



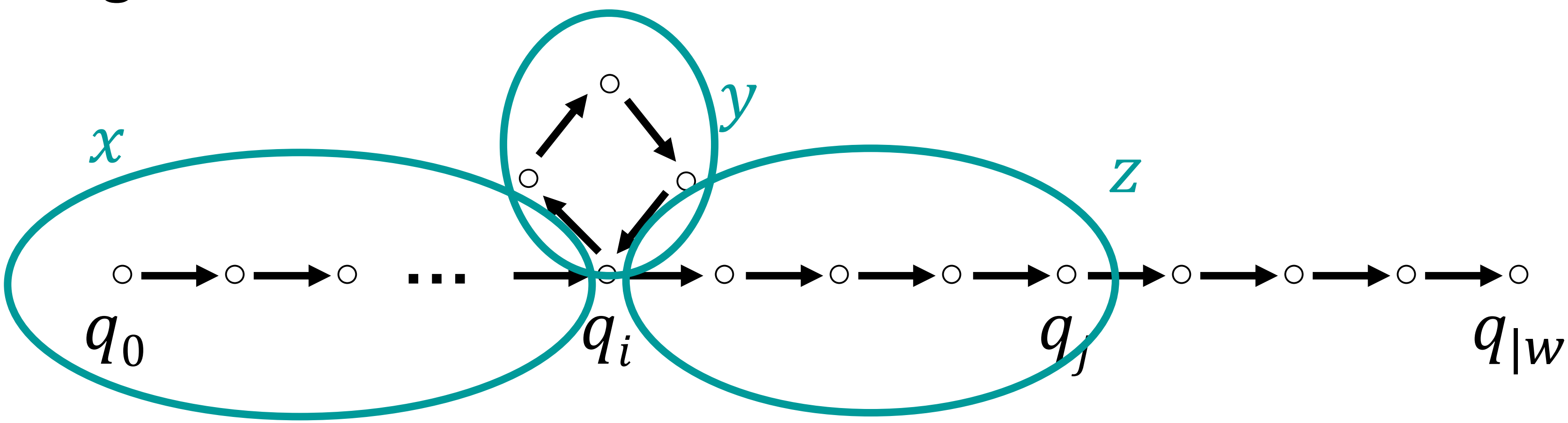
Intuition for the Pumping Lemma

Imagine a **DFA** with p states that recognizes strings of length $> p$

Idea: If you can go around the cycle once, you can go around 0 or 2,3,4… times

Pumping Lemma (Informal)

Let L be a regular language. Let w be a “long enough” string in L.



Then we can write $w = xyz$ such that $xy^iz \in L$ for every $i \geq 0$.

$i = 0$:

$i = 1$:

$i = 2$:

$i = 3$:

Pumping Lemma (Formal)

Let L be a regular language.

Then there exists a "pumping length" p such that

For every $w \in L$ where $|w| \geq p$,

w can be split into three parts $w = xyz$ where:

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

Example:
Let $L = \{w \mid$ all a's in w appear before all b's$\}$; $p = 1$

Using the Pumping Lemma

Theorem: $A = \{0^n 1^n \mid n > 0\}$ is not regular

Proof: (by contradiction)

Assume instead that A is regular. Then A has a pumping length p.

What happens if we try to pump $0^p 1^p$?

If A is regular, w can be split into $w = xyz$, where

1. $|y| > 0$
2. $|xy| \leq p$
3. $xy^i z \in A$ for all $i \geq 0$

General Strategy for proving L is not regular

Proof by contradiction: assume L is regular.

Then there is a pumping length p.

Pumping Lemma as a game

1. YOU pick the language L to be proved nonregular.
2. ADVERSARY picks a possible pumping length p.
3. YOU pick w of length at least p.
4. ADVERSARY divides w into x, y, z, obeying rules of the Pumping Lemma: $|y| > 0$ and $|xy| \leq p$.
5. YOU win by finding $i \geq 0$, for which $xy^i z$ is not in L.

If *regardless* of how the ADVERSARY plays this game, you can always win, then L is nonregular

Example: Palindromes

<u>Claim:</u> $L = \{ww^R \mid w \in \{0,1\}^*\}$ is not regular

<u>Proof:</u> Assume L is regular with pumping length p

1. Find $w \in L$ with $|w| > p$

2. Show that w cannot be pumped
   <u>Intuitively</u>

Example: Palindromes

Claim: $L = \{ww^R \mid w \in \{0,1\}^*\}$ is not regular

Proof: Assume L is regular with pumping length p

1. Find $w \in L$ with $|w| > p$

2. Show that w cannot be pumped
   Formally   If $w = xyz$ with $|xy| \leq p$, then...

Now you try!

Claim: $L = \{0^i 1^j \mid i > j \geq 0\}$ is not regular

Proof: Assume L is regular with pumping length p

1. Find $w \in L$ with $|w| > p$

2. Show that w cannot be pumped
   Intuitively

Now you try!

Claim: $L = \{0^i 1^j \,|i > j \geq 0\}$ is not regular

Proof: Assume L is regular with pumping length p

1. Find $w \in L$ with $|w| > p$

2. Show that w cannot be pumped
   Formally If $w = xyz$ with $|xy| \leq p$, then...

Choosing wisely

<u>Claim:</u> $BALANCED = \{w \mid w \text{ has an equal \# of 0s and 1s}\}$ is not regular

<u>Proof:</u> Assume L is regular with pumping length p

1. Find $w \in L$ with $|w| > p$

2. Show that w cannot be pumped
   <u>Formally</u> If $w = xyz$ with $|xy| \leq p$, then…

Reusing a Proof

Pumping a language can be lots of work…

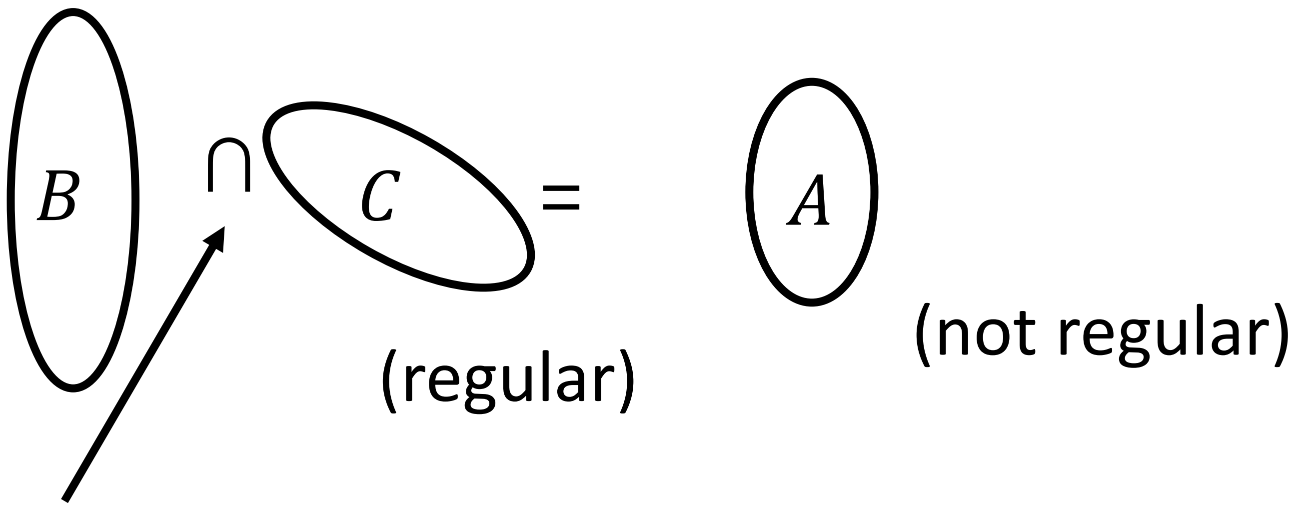
Let's try to reuse that work!

How else might we show that $BALANCED$ is regular?

$\{0^n 1^n \mid n \geq 0\}$ = $BALANCED \cap \{w$ | all 0s in w appear before all 1s$\}$

Using Closure Properties

If A is not regular, we can show a related language B is not regular



any of $\{\circ, \cup, \cap\}$ or, for one language, $\{\neg, {}^{R}, {}^{*}\}$

<u>By contradiction:</u> If B is regular, then $B \cap C$ $(= A)$ is regular.

But A is not regular so neither is B!

Example

Prove $B = \{0^i 1^j | i \neq j\}$ is not regular

using nonregular language $A = \{0^n 1^n | n \geq 0\}$

and regular language

$C = \{w \mid \text{all 0s in } w \text{ appear before all 1s}\}$