BU CS 332 – Theory of Computation

https://forms.gle/EmwazuipdvDh21yLA

Lecture 5:

- Closure Properties
- Regular Expressions

Reading:

Sipser Ch 1.2-1.3

Mark Bun

February 5, 2024

Last Time

- Nondeterministic Finite Automata
- NFAs vs. DFAs
  - Subset construction: NFA → DFA

Closure Properties

An Analogy

In algebra, we try to identify operations which are common to many different mathematical structures

Example: The integers $\mathbb{Z} = \{\ldots -2, -1, 0, 1, 2, \ldots\}$ are **closed** under

- Addition: $x + y$ — $7 + (-8) = -1 \in \mathbb{Z}$
- Multiplication: $x \times y$ — $24 \times (-2) = -48 \in \mathbb{Z}$
- Negation: $-x$ — $-(-7) = 7 \in \mathbb{Z}$
- …but NOT Division: $x\ /\ y$ — $2/7 \notin \mathbb{Z}$

We'd like to investigate similar closure properties of the class of regular languages

Regular operations on languages

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

$A = \{a, b\}$

$A^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}$

Union: $A \cup B = \{w \mid w \in A \textbf{ or } w \in B\}$

$B = \{aa, b\}$

$B^* = \{\varepsilon, aa, b, aaaa, aab, baa, bb, aaaaaa, \ldots\}$

Concatenation: $A \circ B = \{xy \mid x \in A, y \in B\}$

$A \circ A = \{w_1 w_2 \mid w_1, w_2 \in A\}$

Star: $A^* = \{w_1 w_2 \ldots w_n \mid n \geq 0,\ w_i \in A \text{ for each } i = 1, \ldots, n\}$

$= \{\varepsilon\} \cup A \cup A \circ A \cup A \circ A \circ A \cup \ldots$

$= \bigcup_{n=0}^{\infty} A^n$ where $A^n = \underbrace{A \circ A \ldots \circ A}_{n}$

Other operations

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

Complement: $\bar{A} = \{w \mid w \notin A\}$

Intersection: $A \cap B = \{w \mid w \in A \textbf{ and } w \in B\}$

Reverse: $A^R = \{w \mid w^R \in A\}$

Operations on languages

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

Regular Operations

Union: $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$

Concatenation: $A \circ B = \{xy \mid x \in A, y \in B\}$

Star: $A^* = \{ w_1 w_2 ... w_n \mid n \geq 0 \text{ and } w_i \in A\}$

Complement: $\bar{A} = \{x \mid x \notin A\}$

Intersection: $A \cap B = \{x \mid x \in A \text{ and } x \in B\}$

Reverse: $A^R = \{ a_1 a_2 ... a_n \mid a_n ... a_1 \in A\}$

Theorem: The class of regular languages is closed under all six of these operations, i.e., if A and B are regular, applying any of these operations yields a regular language

Proving Closure Properties

Complement

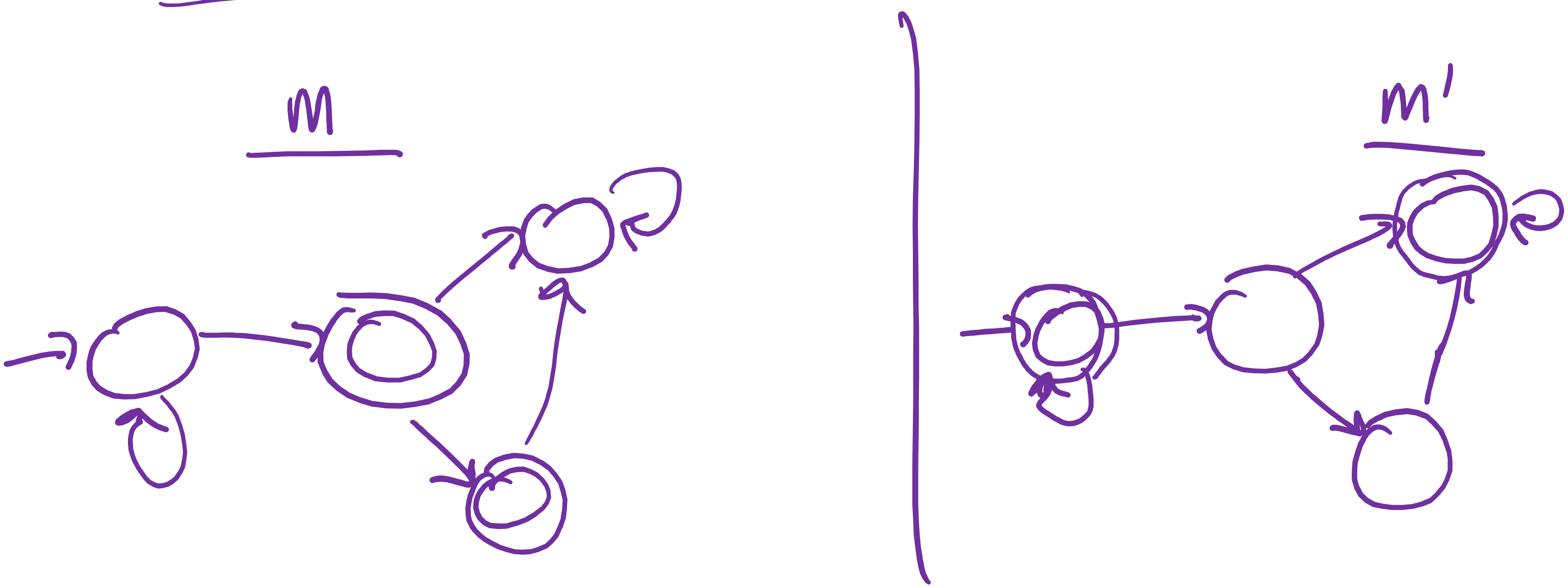
Complement: $\bar{A} = \{ w \mid w \notin A \}$

Theorem: If A is regular, then $\bar{A}$ is also regular

Proof idea: A regular $\Rightarrow$ A recognized by some DFA M

Use M to construct a new DFA M' recognizing $\bar{A}$

Construction of M': Exchange roles of accept & reject states of M



Complement, Formally

Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA recognizing a language A. Which of the following represents a DFA recognizing $\bar{A}$?

a) $(F, \Sigma, \delta, q_0, Q)$

b) $(Q, \Sigma, \delta, q_0, Q \setminus F)$, where $\underbrace{Q \setminus F}_{\text{set of reject states in original DFA}}$ is the set of states in Q that are not in F

c) $(Q, \Sigma, \delta', q_0, F)$ where $\delta'(q, s) = p$ such that $\delta(p, s) = q$

$p \xrightarrow{s} q$ in M $\qquad$ $p \xleftarrow{s} q$ in M'

d) None of the above

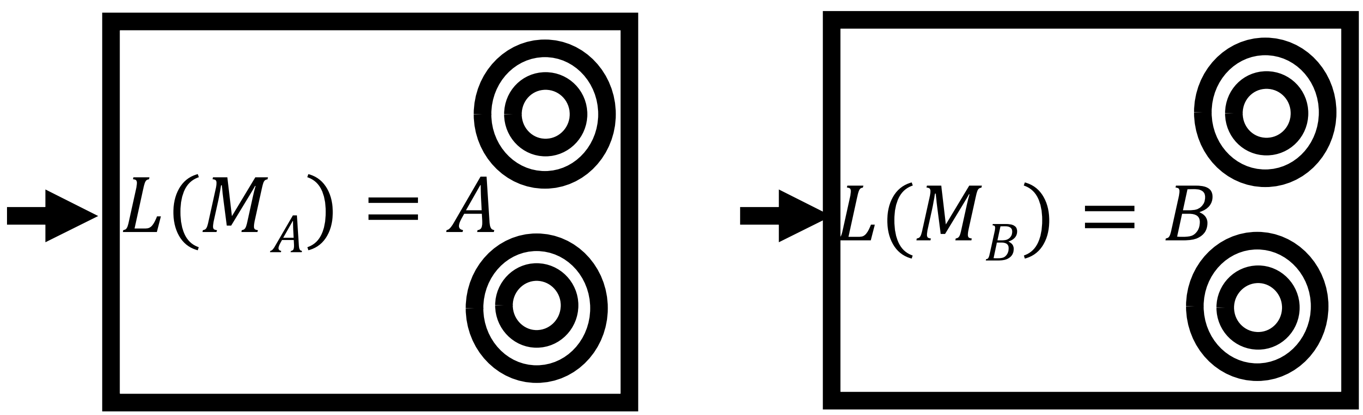
Closure under Concatenation

Concatenation: $A \circ B \; = \; \{ \, xy \; | x \in A, y \in B \, \}$

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

Proof idea: 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.



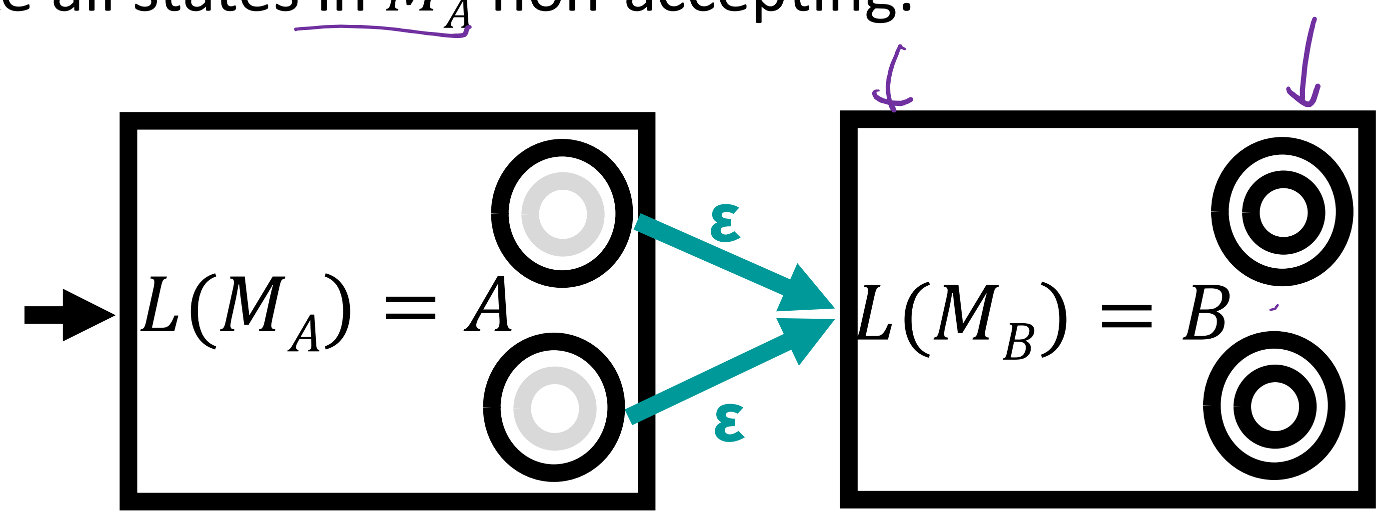
Closure under Concatenation

Concatenation: $A \circ B = \{ xy \mid x \in A, y \in B \}$

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

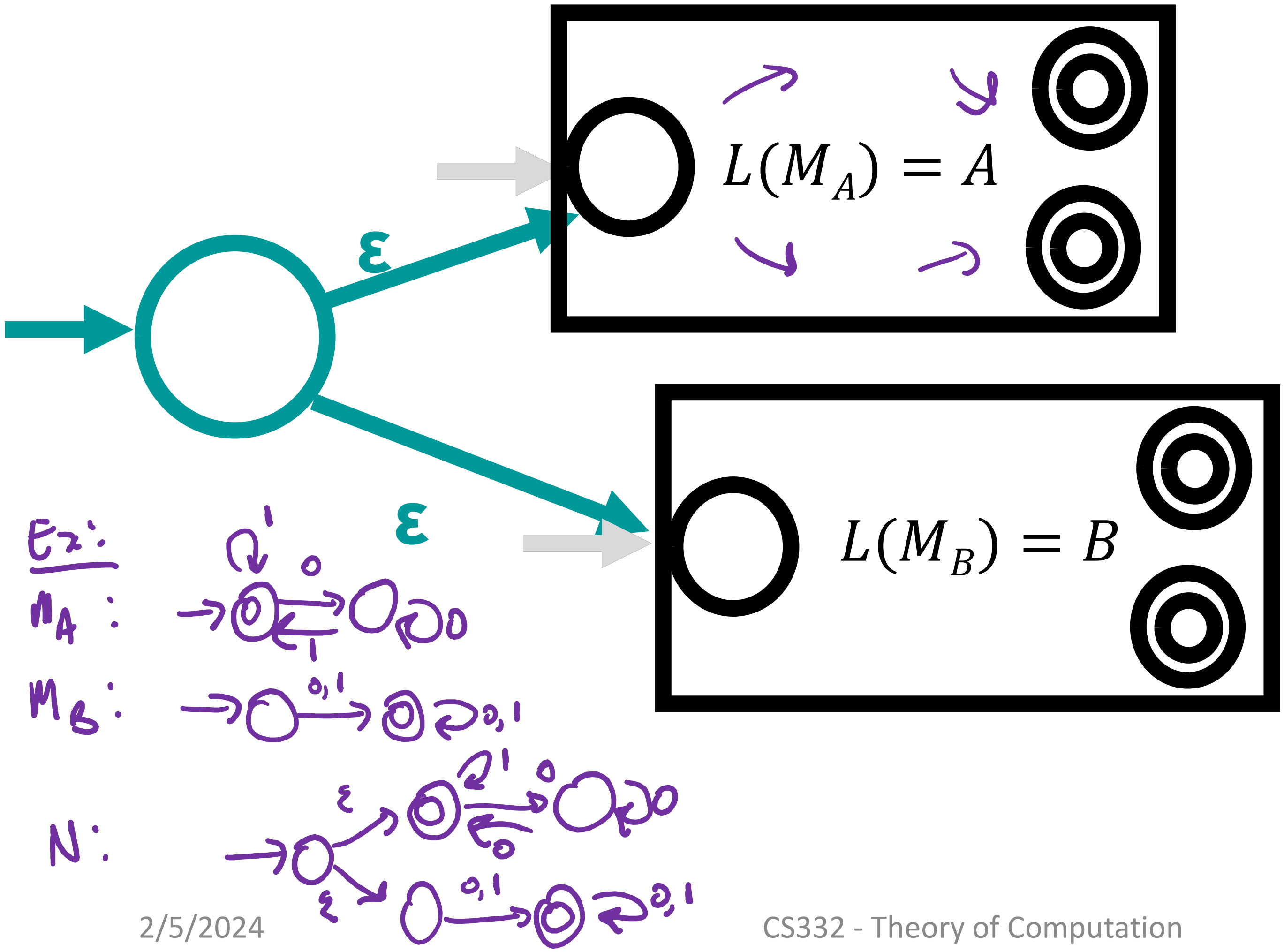
Proof idea: Given DFAs M_A and M_B, construct NFA by

- 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?



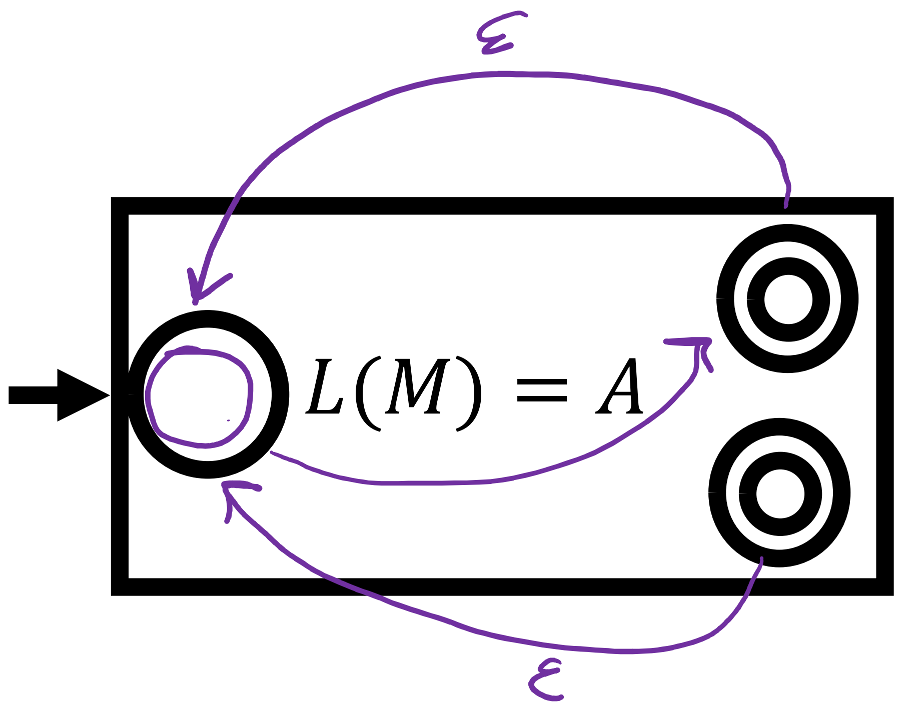
a) $A \cup B$
b) $A \circ B$
c) $A \cap B$
d) $\{\varepsilon\} \cup A \cup B$

Closure under Star

Star: $A^* = \{ a_1 a_2 \ldots a_n \,|\, n \geq 0 \text{ and } a_i \in A\}$

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

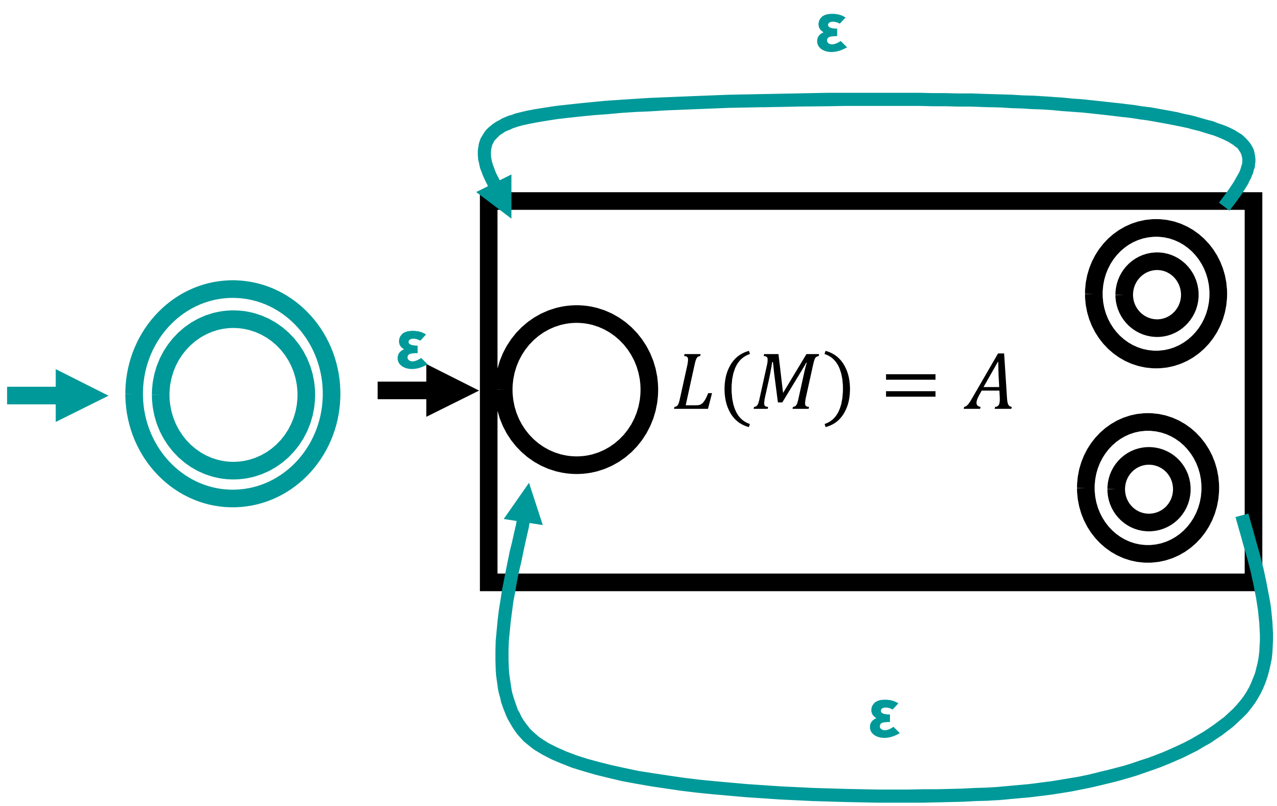
Subtle issue: might accidentally accept strings not in A^*



Closure under Star

Star: $A^* = \{ a_1 a_2 \ldots a_n \mid n \geq 0 \text{ and } a_i \in A\}$

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



On proving your own closure properties

You'll have homework/test problems of the form "show that the regular languages are closed under some operation"

Given op(A, B) on languages, show: If A, B are arbitrary regular langs, then op(A, B) is also regular

What would Sipser do?

- Give the "proof idea": Explain how to take machine(s) recognizing regular language(s) and create a new machine
- Explain in a few sentences why the construction works
- Give a formal description of the construction
- No need to formally prove that the construction works

Sometimes possible to bypass if op can be written as a composition of other operations

Regular Expressions

Regular Expressions

- A different way of describing regular languages
- A regular expression expresses a (possibly complex) language by combining simple languages using the regular operations

"Simple" languages: $\emptyset, \{\varepsilon\}, \{a\}$ for some $a \in \Sigma$

Regular operations:

Union: $A \cup B$

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

Star: $A^* = \{ a_1 a_2 \ldots a_n \mid n \geq 0$ and $a_i \in A\}$

Regular Expressions – Syntax

A regular expression R is defined recursively using the following rules:

1. ε, $\emptyset$, and a are regular expressions for every $a \in \Sigma$

2. If R_1 and R_2 are regular expressions, then so are
   $(R_1 \cup R_2)$, $(R_1 \circ R_2)$, and (R_1^*)

Examples: (over $\Sigma = \{a, b, c\}$)

$(a \circ b)$ $\quad$ $((((a \circ (b^*)) \circ c) \cup (((a^*) \circ b))^*))$ $\quad$ $(\emptyset^*)$

Regular Expressions – Semantics

$L(R)$ = the language a regular expression describes

1. $L(\emptyset) = \emptyset$
2. $L(\varepsilon) = \{\varepsilon\}$
3. $L(a) = \{a\}$ for every $a \in \Sigma$
4. $L((R_1 \cup R_2)) = L(R_1) \cup L(R_2)$
5. $L((R_1 \circ R_2)) = L(R_1) \circ L(R_2)$
6. $L\big((R_1^*)\big) = (L(R_1))^*$

Regular Expressions – Example

$L(((a^*) \circ (b^*))) =$

a) $\{a^n b^n \mid n \geq 0\}$

b) $\{a^m b^n \mid m, n \geq 0\}$

c) $\{(ab)^n \mid n \geq 0\}$

d) $\{a, b\}^*$

$L(a \cup b) = \{a, b\}$

1) $L(a) = \{a\}$

$L(b) = \{b\}$

2) $L(a^*) = (L(a))^* = \{a\}^* = \{a^n \mid n \geq 0\}$

$L(b^*) = \{b^n \mid n \geq 0\}$

3) $L((a^*) \circ (b^*)) = L(a^*) \circ L(b^*)$

$= \{a^n \mid n \geq 0\} \circ \{b^n \mid n \geq 0\}$

$= \{a^m b^n \mid m, n \geq 0\}$

Simplifying Notation

- Omit $\circ$ symbol: $(ab) = (a \circ b)$

- Omit many parentheses, since union and concatenation are associative:

$$(a \cup b \cup c) = (a \cup (b \cup c)) = ((a \cup b) \cup c)$$

- Order of operations: Evaluate star, then concatenation, then union

$$ab^* \cup c = (a(b^*)) \cup c$$

Examples

Let $\Sigma = \{0, 1\}$

01 01000

1. $\{w \mid w$ contains exactly one $1\}$

Attempt 1

$L(1) = \{1\}$

Attempt 2:

$L(0^* 1 0^*)$ ✓

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

2. $\{w \mid w$ has length at least 3 and its third symbol is $0\}$

$L((0 \cup 1)(0 \cup 1) 0 (0 \cup 1)^*)$

3. $\{w \mid$ every odd position of w is $1\}$

Attempt 1 $L((1(0 \cup 1))^*)$

10

11

1010

1011 ...

111

Attempt 2

$L(((1(0 \cup 1))^* (1 \cup \varepsilon)))$