Jonathan Llovet
COSC 417
2026-XX-XX

Template Library: Theory of Computation

Contents

0.1 Sets and Set Operations

Membership and common sets:

$$x \in A, \quad y \notin B, \quad \mathbb{N}, \quad \mathbb{Z}, \quad \mathbb{R}, \quad \emptyset$$

Operations:

$$A \cup B, \quad A \cap B, \quad A \setminus B, \quad \overline{A}, \quad A \times B$$

Subset and proper subset:

$$A \subseteq B, \quad A \subset B, \quad A \supseteq B$$

Power set:

$$\mathcal{P}(\{a,b\}) = \{\emptyset, \{a\}, \{b\}, \{a,b\}\}$$

Cardinality:

$$|A| = n, \quad |\mathcal{P}(A)| = 2^{|A|}$$

Set-builder notation:

$$A = \{x \in \mathbb{N} \mid x > 5\}$$

0.2 Logic and Quantifiers

Quantifiers:

$$\forall x \in A, \quad \exists y \in B, \quad \nexists z \in C$$

Logical connectives:

$$P \wedge Q, \quad P \vee Q, \quad \neg P, \quad P \implies Q, \quad P \iff Q$$

Contrapositive pattern (useful for proofs):

$$(P \implies Q) \iff (\neg Q \implies \neg P)$$

0.3 Functions

Function signature:

$$f : A \to B$$

Injective (one-to-one):

$$\forall a, b \in A,\ a \neq b \implies f(a) \neq f(b)$$

Surjective (onto):

$$\forall b \in B,\ \exists a \in A : f(a) = b$$

Bijective:

$$f \text{ is injective and surjective}$$

Composition:

$$(g \circ f)(x) = g(f(x))$$

0.4 Strings and Languages

Alphabet, string, empty string:

$$\Sigma = \{0, 1\}, \quad w \in \Sigma^*, \quad \varepsilon$$

String length and concatenation:

$$|w|, \quad w_1 \cdot w_2, \quad w^R \text{ (reversal)}$$

Language operations:

$$L_1 \cup L_2, \quad L_1 \cap L_2, \quad L_1 \cdot L_2, \quad L^*, \quad L^+, \quad \overline{L} = \Sigma^* \setminus L$$

Language defined by set-builder:

$$L = \{w \in \{0, 1\}^* \mid w \text{ contains an even number of 0s}\}$$

0.5 Direct Proof

Claim 1. *For all $n \in \mathbb{N}$, if n is even, then n^2 is even.*

Proof. Let n be an even natural number. Then $n = 2k$ for some $k \in \mathbb{N}$. Therefore $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$, which is even. ■

0.6 Proof by Contraposition

Claim 2. $\forall a, b \in \mathbb{N},\ a \neq b \implies f(a) \neq f(b)$.

Proof. We prove the contrapositive: $f(a) = f(b) \implies a = b$.

Suppose $f(a) = f(b)$. Then

$$\begin{aligned} f(a) &= f(b) && \text{(assumption)} \\ a - 2 &= b - 2 && \text{(by definition of } f\text{)} \\ a &= b && \text{(adding 2 to both sides)} \end{aligned}$$

Since $(P \implies Q) \iff (\neg Q \implies \neg P)$, the original statement holds. ∎

0.7 Proof by Contradiction

Claim 3. *$\sqrt{2}$ is irrational.*

Proof. Assume for contradiction that $\sqrt{2}$ is rational. Then $\sqrt{2} = \frac{p}{q}$ where $p, q \in \mathbb{Z}$, $q \neq 0$, and $\gcd(p, q) = 1$.

Squaring both sides: $2 = \frac{p^2}{q^2}$, so $p^2 = 2q^2$. Thus p^2 is even, so p is even. Write $p = 2k$. Then $4k^2 = 2q^2$, so $q^2 = 2k^2$, meaning q is also even.

But this contradicts $\gcd(p, q) = 1$. $\Rightarrow\!\Leftarrow$ ∎

(Note: define `\contradiction` in preamble as `\newcommand{\contradiction}{\Rightarrow\!` or use `\lightning` from `stmaryrd`.)

0.8 Proof by Induction

Claim 4. $\forall n \geq 1$, $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$.

Proof. **Base case** $(n = 1)$: $\sum_{i=1}^{1} i = 1 = \frac{1 \cdot 2}{2}$. ✓

Inductive hypothesis: Assume the claim holds for some $k \geq 1$:

$$\sum_{i=1}^{k} i = \frac{k(k+1)}{2} \tag{I.H.}$$

Inductive step: We show it holds for $k+1$.

$$\begin{aligned}
\sum_{i=1}^{k+1} i &= \left(\sum_{i=1}^{k} i\right) + (k+1) \\
&= \frac{k(k+1)}{2} + (k+1) \quad \text{by } (\textbf{??}) \\
&= \frac{k(k+1) + 2(k+1)}{2} \\
&= \frac{(k+1)(k+2)}{2}
\end{aligned}$$

By the principle of mathematical induction, the claim holds for all $n \geq 1$. ■

0.9 Pumping Lemma Proof (Regular Languages)

Claim 5. *$L = \{0^n 1^n \mid n \geq 0\}$ is not regular.*

Proof. Assume for contradiction that L is regular. Let p be the pumping length given by the Pumping Lemma.

Choose $s = 0^p 1^p \in L$. Since $|s| = 2p \geq p$, the Pumping Lemma guarantees $s = xyz$ where:

1. $|y| > 0$,
2. $|xy| \leq p$,
3. $\forall i \geq 0,\ xy^i z \in L$.

Since $|xy| \leq p$, y consists entirely of 0s. Write $y = 0^k$ for some $k \geq 1$.

Consider $xy^2z = 0^{p+k}1^p$. Since $k \geq 1$, we have $p + k > p$, so $xy^2z \notin L$. This contradicts condition 3.

Therefore L is not regular. ■

0.10 Pumping Lemma Proof (Context-Free Languages)

Claim 6. *$L = \{a^n b^n c^n \mid n \geq 0\}$ is not context-free.*

Proof. Assume for contradiction that L is context-free. Let p be the pumping length given by the Pumping Lemma for CFLs.

Choose $s = a^p b^p c^p \in L$. Since $|s| = 3p \geq p$, we can write $s = uvxyz$ where:

1. $|vy| > 0$,
2. $|vxy| \leq p$,
3. $\forall i \geq 0,\ uv^i xy^i z \in L$.

Since $|vxy| \leq p$, the substring vxy cannot span all three symbols a, b, c. Therefore pumping ($i = 2$) will increase the count of at most two of the three symbols, making $uv^2xy^2z \notin L$. Contradiction. ■

0.11 Equation Environments

Unnumbered display math (no label/ref possible):

$$a + b = c$$

Numbered equation (can label and reference):

$$E = mc^2 \tag{1}$$

Reference: Equation (**??**).

Multi-line aligned at equals, selectively numbered:

$$\begin{aligned} f(x) &= (x+1)^2 \\ &= x^2 + 2x + 1 \end{aligned} \tag{2}$$

Fully unnumbered multi-line:

$$\begin{aligned} a &= b + c \\ d &= e + f \end{aligned}$$

Annotated derivation (double alignment):

$$\begin{aligned} f(c) &= f(d) && \text{(given)} \\ c - 2 &= d - 2 && \text{(by definition)} \\ \therefore c &= d && \text{(by algebra)} \end{aligned}$$

Custom-tagged equation:

$$a = k - 2 \tag{Inductive Hypothesis}$$

Cases (piecewise functions):

$$f(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ f(n-1) + f(n-2) & \text{if } n \geq 2 \end{cases}$$

Inline condition after display math:

$$\begin{aligned} a &= c \\ b &= d \end{aligned} \quad \text{where } c, d \in \mathbb{N}.$$

0.12 Matrices

Parenthesized matrix:

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Bracketed matrix:

$$B = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Determinant (vertical bars):

$$\det(A) = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

Augmented matrix (useful for Gaussian elimination):

$$\left[\begin{array}{cc|c} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array} \right]$$

0.13 Transition Table (DFA)

δ	0	1
$\rightarrow q_0$	q_1	q_2
q_1	q_1	q_3
q_2	q_3	q_2
$*q_3$	q_3	q_3

Convention: $\rightarrow$ marks the start state, $*$ marks accepting states.

0.14 Transition Table (NFA)

δ	0	1	ε
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$	$\emptyset$
q_1	$\emptyset$	$\{q_2\}$	$\{q_2\}$
$*q_2$	$\emptyset$	$\emptyset$	$\emptyset$

0.15 Truth Table

P	Q	$P \wedge Q$	$P \vee Q$	$P \implies Q$
T	T	T	T	T
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

0.16 Diagonalization Table

Used in Cantor's proof that the set of infinite binary sequences is uncountable. Suppose for contradiction that $f : \mathbb{N} \to \{0, 1\}^\omega$ is a bijection. List the supposed enumeration:

$$\begin{array}{c|cccccc} & b_1 & b_2 & b_3 & b_4 & b_5 & \cdots \\ \hline f(1) & \bar{\mathbf{d}} & 0 & 1 & 0 & 1 & \cdots \\ f(2) & 1 & \bar{\mathbf{d}} & 0 & 1 & 1 & \cdots \\ f(3) & 0 & 1 & \bar{\mathbf{d}} & 0 & 0 & \cdots \\ f(4) & 1 & 1 & 0 & \bar{\mathbf{d}} & 1 & \cdots \\ f(5) & 0 & 0 & 1 & 1 & \bar{\mathbf{d}} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

Construct $d = d_1 d_2 d_3 \ldots$ where $d_i \neq f(i)_i$ (i.e., flip the i-th bit of the i-th sequence). The diagonal entries $\bar{d}$ are the bits we flip. Then d differs from every $f(n)$ in position n, so $d \notin \{f(1), f(2), \ldots\}$, contradicting surjectivity.

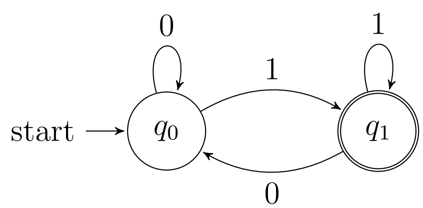
The same structure applies to proving A_{TM} is undecidable. Suppose H decides A_{TM}. Build a table of TMs vs. their own descriptions:

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\cdots$
M_1	**accept**	reject	accept	reject	$\cdots$
M_2	accept	**accept**	accept	accept	$\cdots$
M_3	reject	reject	**reject**	accept	$\cdots$
M_4	accept	accept	reject	**reject**	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$
D	**reject**	**reject**	**accept**	**accept**	$\cdots$

Entry $(M_i, \langle M_j \rangle)$ shows whether M_i accepts $\langle M_j \rangle$. The diagonal is $H(M_i, \langle M_i \rangle)$—does M_i accept its own description? The machine D flips the diagonal: $D(\langle M_i \rangle)$ does the opposite of M_i on $\langle M_i \rangle$. Then $D(\langle D \rangle)$ must both accept and reject—contradiction.

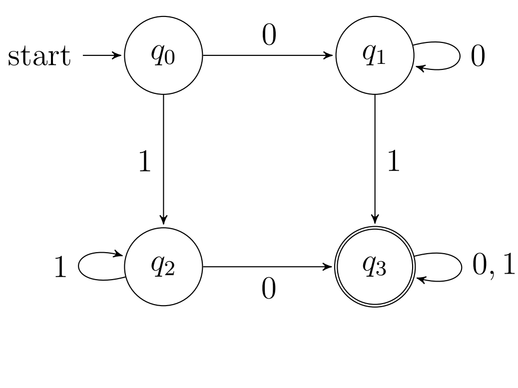
0.17 DFA: Simple Linear

A DFA that accepts strings ending in 1 over $\Sigma = \{0, 1\}$.



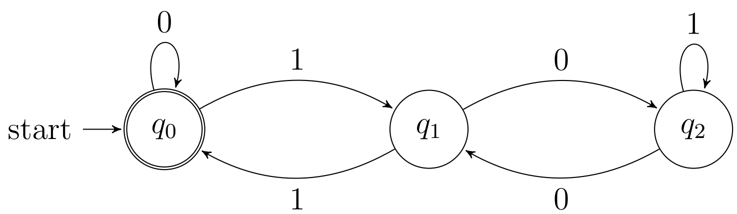
0.18 DFA: Grid Layout

A DFA with states in a 2x2 grid (e.g., tracking two properties).



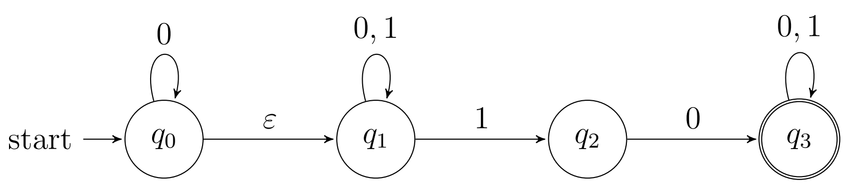
0.19 DFA: Modular Arithmetic (mod 3)

Accepts binary strings whose numeric value is divisible by 3.



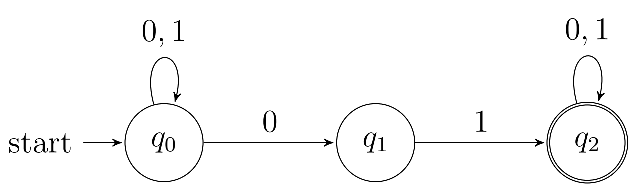
0.20 NFA: With Epsilon Transitions

An NFA with ε-transitions.



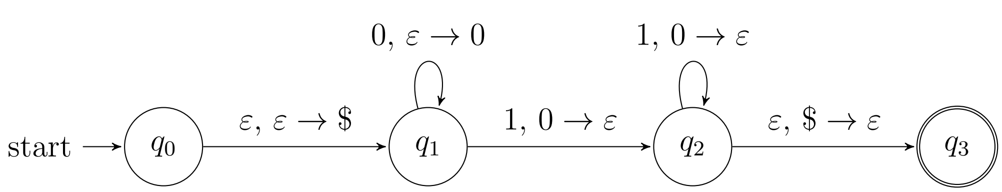
0.21 NFA: Nondeterministic Branching

An NFA that accepts strings containing the substring "01".



0.22 PDA: Pushdown Automaton

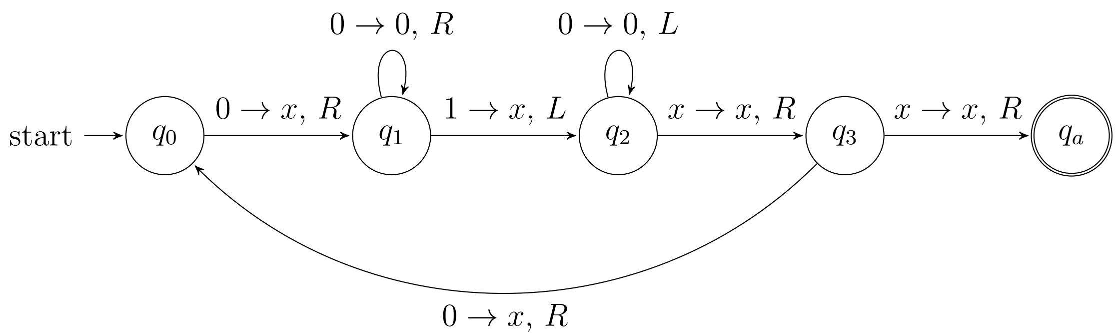
A PDA for $\{0^n1^n \mid n \geq 0\}$.



PDA transitions use the notation: input, pop $\to$ push.

0.23 Turing Machine

A TM that accepts $\{0^n1^n \mid n \geq 1\}$.



TM transitions use the notation: read $\to$ write, direction.

0.24 Context-Free Grammar

A CFG for $\{0^n1^n \mid n \geq 0\}$:

$$S \to 0S1 \mid \varepsilon$$

A CFG for balanced parentheses:

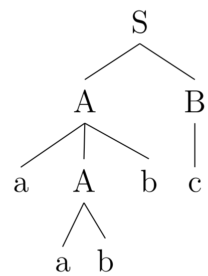
$$S \to SS \mid (S) \mid \varepsilon$$

A more elaborate grammar with multiple productions:

$$\begin{aligned} S &\to AB \mid \varepsilon \\ A &\to aAb \mid ab \\ B &\to cB \mid c \end{aligned}$$

0.25 Parse Tree

Using `tikz-qtree` for parse trees:



0.26 Derivation Steps

A leftmost derivation:

$$\begin{aligned} S &\Rightarrow AB && (\text{rule } S \to AB) \\ &\Rightarrow aAbB && (\text{rule } A \to aAb) \\ &\Rightarrow aabbB && (\text{rule } A \to ab) \\ &\Rightarrow aabbc && (\text{rule } B \to c) \end{aligned}$$

Using $\Rightarrow^*$ for multi-step derivation:

$$S \Rightarrow^* aabbc$$

0.27 Regular Expressions

Basic notation:

$$\begin{aligned} \Sigma &= \{0,1\} \\ L_1 &= 0^*10^* && (\text{strings with exactly one 1}) \\ L_2 &= (0 \cup 1)^* && (\text{all strings over } \Sigma) \\ L_3 &= \Sigma^*01\Sigma^* && (\text{contains substring 01}) \end{aligned}$$

Common patterns:

$$\begin{array}{ll} \text{At least one } a: & a\Sigma^* \cup \Sigma^* a \\ \text{Even length:} & (\Sigma\Sigma)^* \\ \text{Starts and ends same:} & 0\Sigma^*0 \cup 1\Sigma^*1 \cup 0 \cup 1 \cup \varepsilon \end{array}$$

0.28 Complexity Classes

Big-O and related notation:

$$f(n) = O(g(n)), \quad f(n) = \Omega(g(n)), \quad f(n) = \Theta(g(n))$$

Class definitions:

$$\mathbf{P} = \bigcup_{k \geq 0} \mathrm{TIME}(n^k)$$

$$\mathbf{NP} = \bigcup_{k \geq 0} \mathrm{NTIME}(n^k)$$

0.29 Reduction Notation

Mapping reduction:

$$A \leq_m B$$

If $A \leq_m B$ and B is decidable, then A is decidable.
If $A \leq_m B$ and A is undecidable, then B is undecidable.

0.30 Labeled Equation References

Define and reference equations:

$$\forall a, b \in A,\ f(a) = f(b) \implies a = b \tag{3}$$

As shown in (**??**), the function is injective.

0.31 Enumerate with Custom Labels

Roman numerals:

(i) First condition

(ii) Second condition

(iii) Third condition

Alphabetical:

(a) Case a

(b) Case b

0.32 Verbatim / Pseudocode

```
DECIDE(w):
  Simulate M on w for |w|^2 steps
  If M accepts, ACCEPT
  If M rejects or loops, REJECT
```

0.33 Including External Files

Include a full PDF:

```
\includepdf[pages=-,pagecommand={},width=\textwidth]{solution.pdf}
```

Include an image:

```
\begin{center}
  \includegraphics[width=0.8\linewidth]{screenshot.png}
\end{center}
```

0.34 Spacing Reference

`\quad`	medium space (1em)
`\qquad`	large space (2em)
`\,`	thin space
`\;`	medium-thin space
`\medskip`	vertical medium skip
`\bigskip`	vertical big skip
`\pagebreak`	force new page

0.35 Useful Symbols Quick Reference

Symbol	Command	Usage
ε	`\varepsilon`	empty string
$\emptyset$	`\emptyset`	empty set
Σ	`\Sigma`	alphabet
δ	`\delta`	transition function
$\vdash$	`\vdash`	yields / proves
$\therefore$	`\therefore`	therefore
$\implies$	`\implies`	implies (long arrow)
$\iff$	`\iff`	if and only if
$\le_m$	`\le_m`	mapping reducible
$\Rightarrow$	`\Rightarrow`	derives (grammar)
$\langle M \rangle$	`\langle M \rangle`	encoding of M
$\overline{L}$	`\overline{L}`	complement
$\mathcal{P}$	`\mathcal{P}`	power set
$\mathbb{N}$	`\mathbb{N}`	natural numbers
∞	`\infty`	infinity