INTRODUCTION TO AUTOMATA THEORY

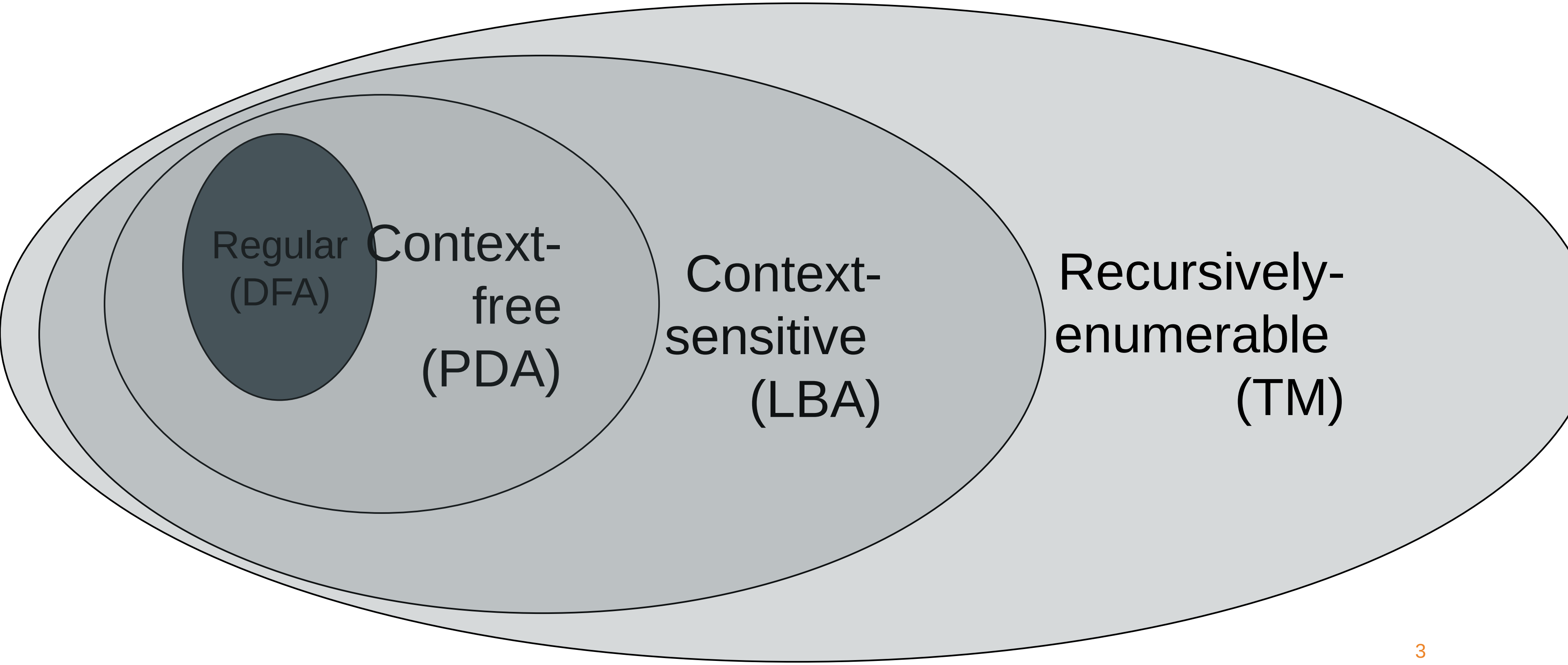
READING: CHAPTER 1

WHAT IS AUTOMATA THEORY?

- *Study of abstract computing devices, or "machines"*
- Automaton = an abstract computing device
  - Note: A "device" need not even be a physical hardware!
- A fundamental question in computer science:
  - Find out what different models of machines can do and cannot do
  - The *theory of computation*
- Computability vs. Complexity

THE CHOMSKY HIERACHY

- A containment hierarchy of classes of formal languages



THE CENTRAL CONCEPTS OF AUTOMATA THEORY

ALPHABET

An alphabet is a finite, non-empty set of symbols

- We use the symbol ∑ (sigma) to denote an alphabet
- Examples:
  - Binary: ∑ = {0,1}
  - All lower case letters: ∑ = {a,b,c,..z}
  - Alphanumeric: ∑ = {a-z, A-Z, 0-9}
  - DNA molecule letters: ∑ = {a,c,g,t}

STRINGS

- *A string or word is a finite sequence of symbols chosen from ∑*
- ***Empty string is ε (or "epsilon")***

- Length of a string *w*, denoted by |*w*|, is equal to the *number of (non- ε) characters in the string*
    - *E.g., x = 010100* |*x*| = 6
    - *x = 01 ε 0 ε 1 ε 00 ε* |*x*| = ?

- *xy = concatenation* of two strings *x* and *y*

POWERS OF AN ALPHABET

- Let Σ be an alphabet.
- Σ^k = the set of all strings of length k
- $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots$
- $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots$

Strings

- The set of all possible strings over Σ is denoted by Σ^*.
- We define $\Sigma^0 = \{\epsilon\}$ and $\Sigma^n = \Sigma^{n-1} \cdot \Sigma$
  - with some abuse of the concatenation notation applying to sets of strings now
- So $\Sigma^n = \{\omega | \omega = xy \text{ and } x \in \Sigma^{n-1} \text{ and } y \in \Sigma\}$
- $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \cdots \Sigma^n \cup \cdots = \bigcup_0^\infty \Sigma^i$
  - Alternatively, $\Sigma^* = \{x_1 x_2 \ldots x_n | n \geq 0 \text{ and } x_i \in \Sigma \text{ for all } i\}$
- Φ denotes the empty set of strings $\Phi = \{\}$,
  - but $\Phi^* = \{\epsilon\}$

$\Sigma^0 = \varepsilon$ for any Σ

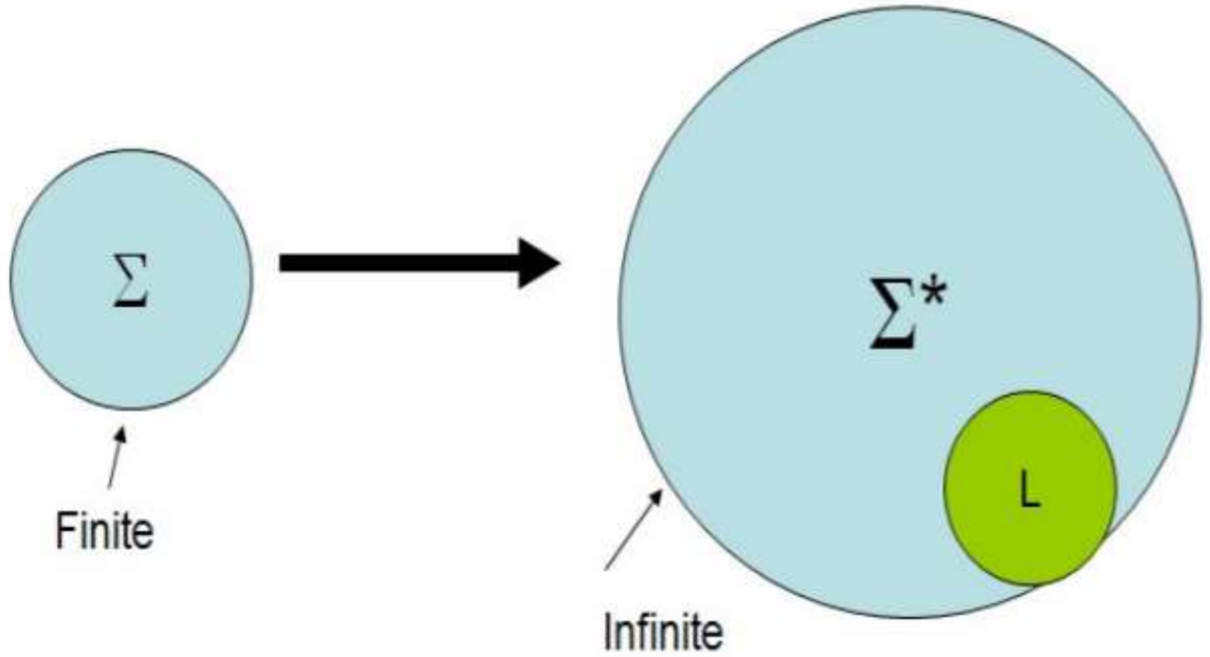
- $\Sigma^1 = \{ a, b, c\}$
- $\Sigma^2 = \{ aa, ab, ac, ba, bb, bc, ca, cb, cc\}$

Strings

- Σ^* is a countably infinite set of finite length strings
- If x is a string, we write x^n for the string obtained by concatenating n copies of x.
  - $(aab)^3 = aabaabaab$
  - $(aab)^0 = \epsilon$

Languages

- A language L over Σ is any subset of Σ^*



- L can be finite or (countably) infinite

LANGUAGES

L is a said to be a language over alphabet Σ, only if $L \subseteq \Sigma^*$

➔ this is because Σ^* is the set of all strings (of all possible length including 0) over the given alphabet Σ

Examples:

1. Let L be *the* language of all strings consisting of n 0's followed by n 1's:

   L = {ε, 01, 0011, 000111,…}

2. Let L be *the* language of all strings of with equal number of 0's and 1's:

   L = {ε, 01, 10, 0011, 1100, 0101, 1010, 1001,…}

Definition: Ø denotes the Empty language Let L = {ε}; Is L=Ø?

Some Languages

- $L = \Sigma^*$ – The mother of all languages!
- $L = \{a, ab, aab\}$ – A fine finite language.
  - Description by enumeration
- $L = \{a^n b^n : n \geq 0\} = \{\epsilon,\ \mathrm{ab},\ \mathrm{aabb},\ \mathrm{aaabbb}, \ldots\}$
- $L = \{\omega | n_a(\omega) \text{ is even}\}$
  - $n_x(\omega)$ denotes the number of occurrences of x in ω
  - all strings with even number of a's.
- $L = \{\omega | \omega = \omega^R\}$
  - All strings which are the same as their reverses – palindromes.
- $L = \{\omega | \omega = xx\}$
  - All strings formed by duplicating some string once.
- $L = \{\omega | \omega$ is a syntactically correct Java program $\}$

Languages

- Since languages are sets, all usual set operations such as intersection and union, etc. are defined.
- Complementation is defined with respect to the universe Σ^* : $\overline{L} = \Sigma^* - L$

Languages

- If L, L_1 and L_2 are languages:
  - $L_1 \cdot L_2 = \{xy | x \in L_1 \text{ and } y \in L_2\}$
  - $L^0 = \{\epsilon\}$ and $L^n = L^{n-1} \cdot L$
  - $L^* = \bigcup_0^\infty L^i$
  - $L^+ = \bigcup_1^\infty L^i$

THE MEMBERSHIP PROBLEM

Given a string $w \in \Sigma^*$ and a language L over Σ, decide whether or not $w \in L$.

<u>Example:</u>

Let w = 100011

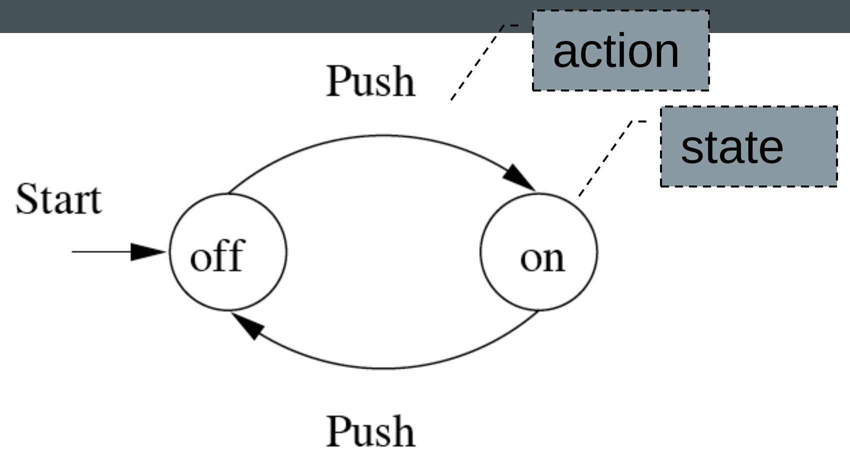
Q) Is $w \in$ the language of strings with equal number of 0s and 1s?

FINITE AUTOMATA

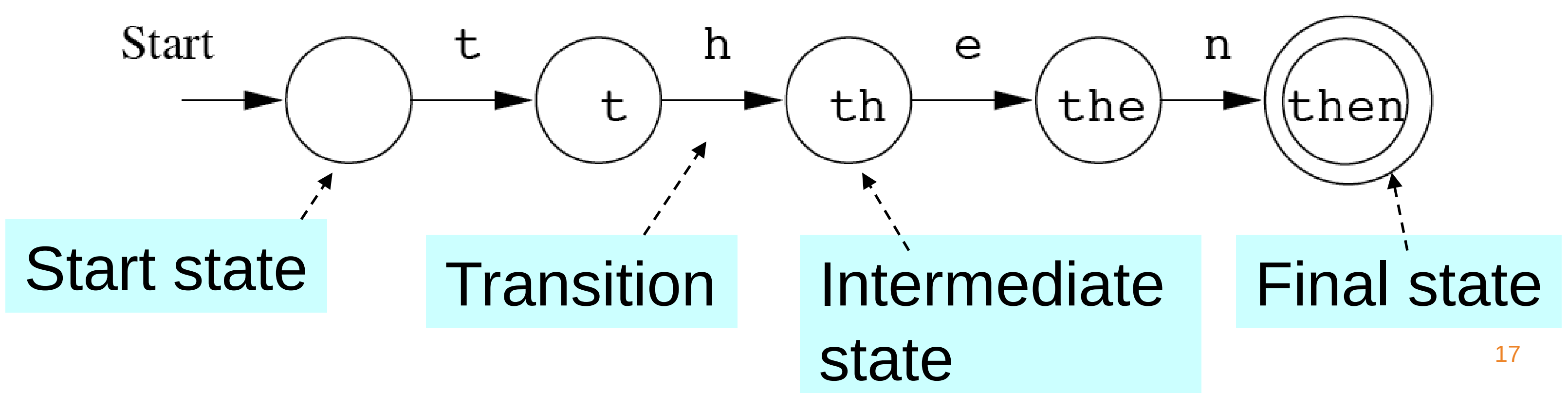
- Some Applications
  - Software for designing and checking the behavior of digital circuits
  - Lexical analyzer of a typical compiler
  - Software for scanning large bodies of text (e.g., web pages) for pattern finding
  - Software for verifying systems of all types that have a finite number of states (e.g., stock market transaction, communication/network protocol)

FINITE AUTOMATA : EXAMPLES

- On/Off switch



- Modeling recognition of the word "*then*"



STRUCTURAL EXPRESSIONS

- Regular expressions
  - E.g., unix style to capture city names such as “Palo Alto CA”:
    - [A-Z][a-z]*([ ][A-Z][a-z]*)*[ ][A-Z][A-Z]

Start with a letter

A string of other letters (possibly empty)

Other space delimited words (part of city name)

Should end w/ 2-letter state code

SUMMARY

- Automata theory & a historical perspective
- Chomsky hierarchy
- Finite automata
- Alphabets, strings/words/sentences, languages
- Membership problem
- Read chapter 1 for more examples and exercises