形式语言与自动机理论

课程简介与基础知识

王春宇

chunyu@hit.edu.cn

计算机科学与技术学院
哈尔滨工业大学

综合楼412

课堂名称：形式语言与自动机
课堂编号：CV676

1、扫码关注公众号：微助教服务号。

2、点击系统通知："点击此处加入【形式语言与自动机】课堂"，填写学生资料加入课堂。

*如未成功收到系统通知，请点击公众号下方"学生" --- "加入课堂" --- "输入课堂编号"手动加入课堂

二维码有效期至：2019-04-24
到期后系统会生成新的二维码

课程简介与基础知识

- 课程简介
- 基础知识

计算理论

核心问题

计算机的基本能力和限制是什么?

❶ 究竟哪些问题, 可通过计算解决? — 可计算性理论

❷ 解决可计算的问题, 究竟需要多少资源? — 计算复杂性理论

❸ 为了研究计算, 要使用哪些计算模型? — 形式语言与自动机理论

什么是自动机理论?

自动机理论: 研究抽象机器及其所能解决问题的理论.

- 图灵机
- 有限状态机
- 文法, 下推自动机

什么是形式语言?

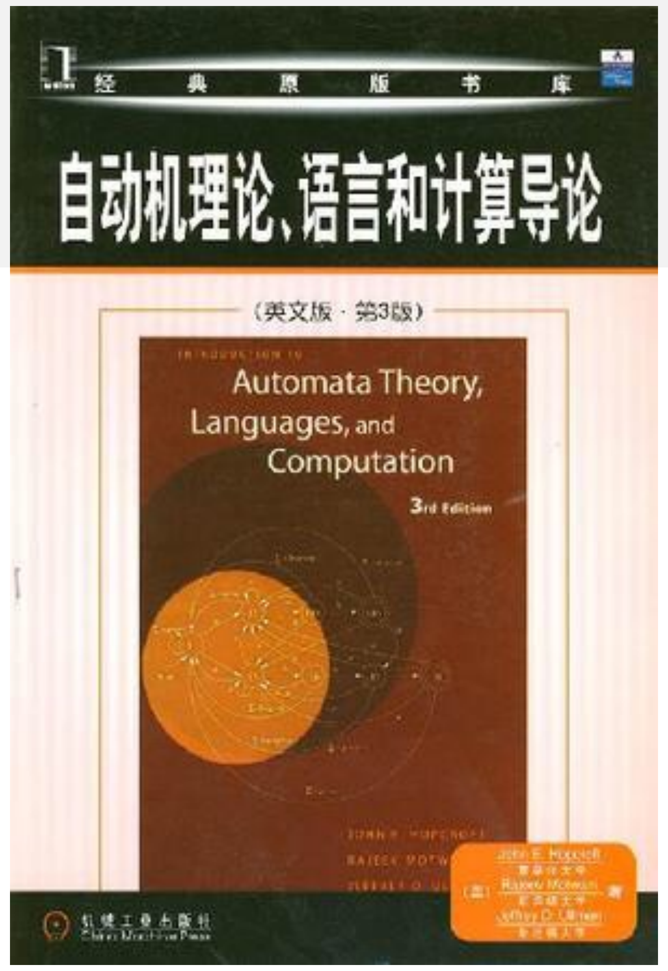
形式语言: 经数学定义的语言.

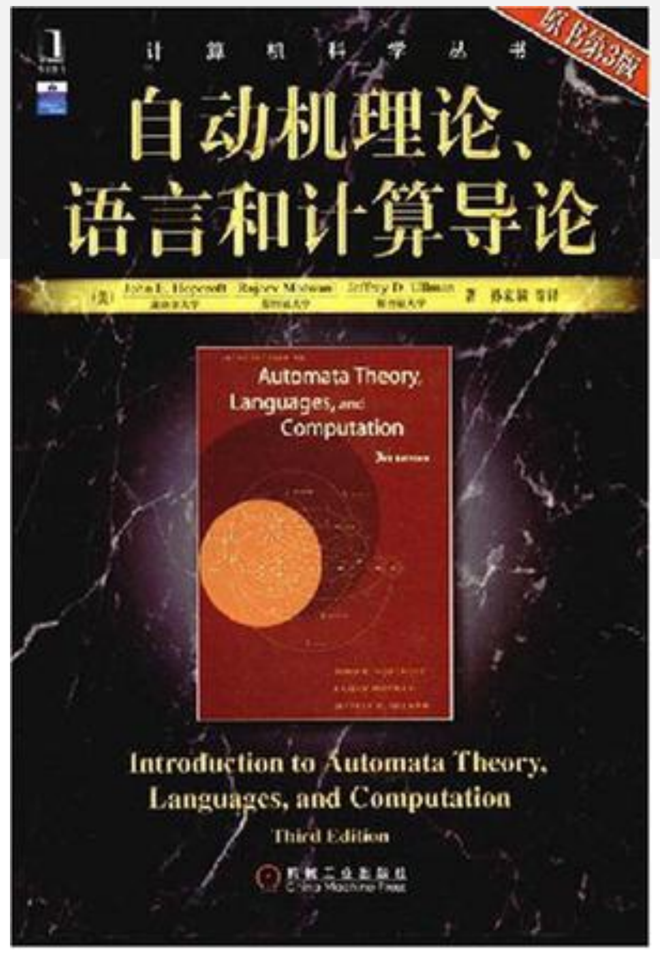
		自然语言		形式语言	
		English	中文	化学分子式	C 语言
语言	字符	A,a,B,b,...	天, 地,...	A-Z,a-z,0-9 ...	`A-Z,a-z,0-9...`
	单词	apple	苹果	H_2O	`char`
	句子	How're you?	早上好!	$2H_2+O_2=2H_2O$	`char a = 10;`
	语法	Grammar	语法规则	精确定义的规则	

课程内容

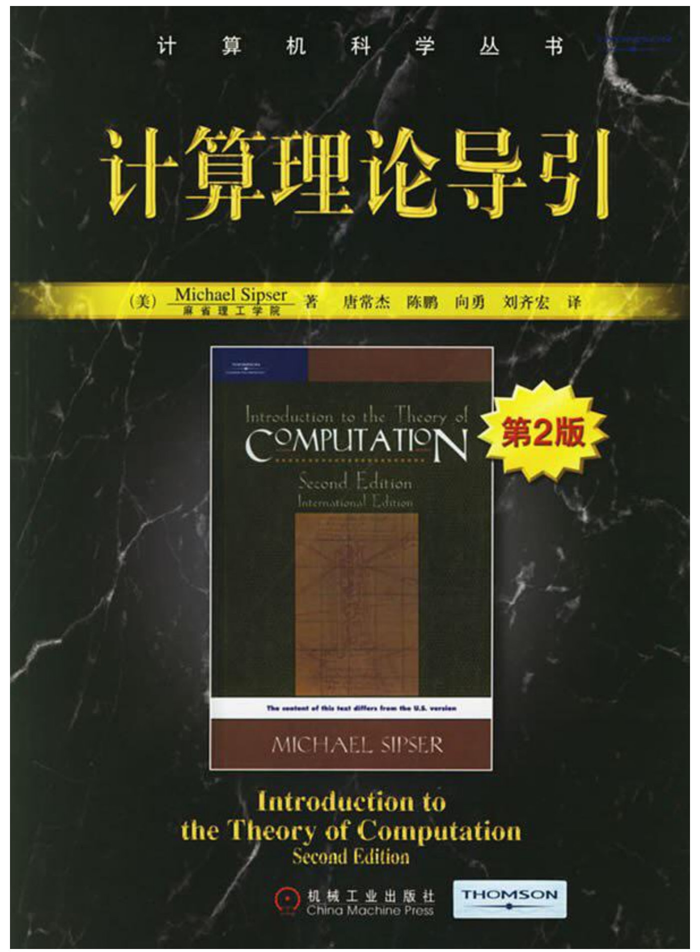
- 正则语言
  - 有穷自动机
  - 正则表达式
  - 正则语言的性质
- 上下文无关语言
  - 上下文无关文法
  - 下推自动机
  - 上下文无关语言的性质
- 计算导论
  - 图灵机及其扩展
  - 不可判定性

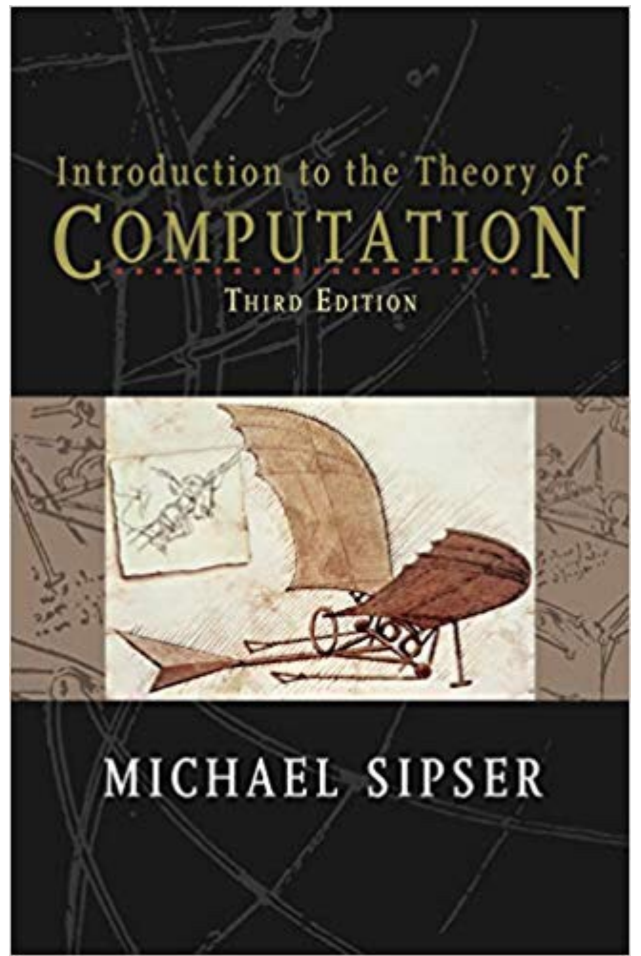
参考书





- John E. Hopcroft. *Introduction to Automata Theory, Languages, and Computation.*
  《自动机理论、语言和计算导论》机械工业出版社
- Michael Sipser *Introduction to the Theory of Computation.*
  《计算理论导引》机械工业出版社





课程简介与基础知识

- 课程简介
- 基础知识
  - 基本概念
  - 语言和问题
  - 形式化证明

基本概念

1. 字母表: 符号 (字符) 的非空有穷集.

$$
\begin{aligned}
\Sigma_1 &= \{0, 1\}, \\
\Sigma_2 &= \{a, b, \ldots, z\}, \\
\Sigma_3 &= \{x \mid x \text{ 是一个汉字}\}.
\end{aligned}
$$

2. 字符串: 由某字母表中符号组成的有穷序列.

若 $\Sigma_1 = \{0, 1\}$, 那么 $0, 1, 00, 111001$ 为 Σ_1 上的字符串;
若 $\Sigma_2 = \{a, b, \ldots, z\}$, 那么 $ab, xkcd$ 为 Σ_2 上的字符串.

3. 空串: 记为ε, 有 0 个字符的串.

字母表 Σ 可以是任意的, 但都有 $\varepsilon \notin \Sigma$.

符号使用的一般约定:

- 字母表: $\Sigma, \Gamma, \ldots$
- 字符: $a, b, c, \ldots$
- 字符串: $\ldots, w, x, y, z$
- 集合: $A, B, C, \ldots$

4. 字符串的长度：字符串中符号所占位置的个数，记为$|\sqcup|$.
若字母表为 Σ，可递归定义为：

$$|w| = \begin{cases} 0 & w = \varepsilon \\ |x| + 1 & w = xa \end{cases},$$

其中 $a \in \Sigma$，w 和 x 是 Σ 中字符组成的字符串.

$$|aab| = |aa| + 1$$
$$= |a| + 1 + 1$$
$$= |\varepsilon| + 1 + 1 + 1 = 3$$

5. 字符串 x 和 y 的连接: 将首尾相接得到新字符串的运算, 记为 $x \cdot y$ 或 xy. 同样, 可递归定义为

$$x \cdot y = \begin{cases} x & y = \varepsilon \\ (x \cdot z)a & y = za \end{cases},$$

其中 $a \in \Sigma$, 且 x, y, z 都是字符串.

$x = 01$ $y = abc$ $x \cdot y = 01abc$

$y \cdot x = abc01$

对任何字符串 x, 有 $\varepsilon \cdot x = x \cdot \varepsilon = x$.

连接运算的符号 "$\cdot$" 一般省略.

6. 字符串 x 的 n 次幂$(n \geq 0)$, 递归定义为

$$x^n = \begin{cases} \varepsilon & n = 0 \\ x^{n-1}x & n > 0 \end{cases}.$$

$x^0 = \varepsilon$.

例如, 若 $\Sigma = \{a, b\}$, 那么

$$(ba)^2 = (ba)^1 ba = (ba)^0 baba = \varepsilon baba = baba$$

$$ba^2 = ba^1 a = b\varepsilon aa = baa$$

7. 集合 A 和 B 的连接, 记为 $A \cdot B$ 或 AB, 定义为

$$A \cdot B = \{w \mid w = x \cdot y,\ x \in A \text{ 且 } y \in B\}.$$

$A = \{0, 11\}$ $B = \{a, b\}$

$A \cdot B = \{0a, 0b, 11a, 11b\}$

$B \cdot A = \{a0, a11, b0, b11\}$

$\emptyset \cdot A = \emptyset$

8. 集合 A 的 n 次幂$(n \geq 0)$, 递归定义为

$$A^n = \begin{cases} \{\varepsilon\} & n = 0 \\ A^{n-1}A & n \geq 1 \end{cases}.$$

$\emptyset^0 = \{\varepsilon\}$ $\quad$ $\{\varepsilon\}^0 = \{\varepsilon\}$

$\emptyset^5 = \emptyset$ $\quad$ $\{\varepsilon\}^{10} = \{\varepsilon\}$

那么, 若 Σ 为字母表, 则 Σ^n 为 Σ 上长度为 n 的字符串集合.
如果 $\Sigma = \{0, 1\}$, 有

$$\Sigma^0 = \{\varepsilon\}, \ \Sigma^1 = \{0, 1\}, \ \Sigma^2 = \{00, 01, 10, 11\},$$

$$\Sigma^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}, \ldots$$

9. 克林闭包(*Kleene Closure*):

$$A^* = \bigcup_{i=0}^{\infty} A^i$$

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i. = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cdots$$

10. 正闭包(*Positive Closure*):

$$A^* = A^+ \cup \{\varepsilon\}$$

$$\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i.$$

$\varepsilon \in A$

显然,

$$\Sigma^* = \Sigma^+ \cup \{\varepsilon\}.$$

其他的概念如有向图, 树, 字符串的前缀, 后缀等定义这里省略.

语言

定义

若 Σ 为字母表且 $\forall L \subseteq \Sigma^*$, 则 L 称为字母表 Σ 上的语言.

- 自然语言, 程序设计语言等
- $\{0^n1^n \mid n \geq 0\}$ $= \{\varepsilon, 01, 0011, 000111, \cdots\}$
- The set of strings of 0's and 1's with an equal number of each:

$$\{\varepsilon, 01, 10, 0011, 0101, 1100, \ldots\}$$

- $\emptyset$, $\{\varepsilon\}$ 和 Σ^* 分别都是任意字母表 Σ 上的语言, 但注意 $\emptyset \neq \{\varepsilon\}$

关于语言

唯一重要的约束就是所有字母表都是有穷的.

问题

自动机理论中的典型问题

判断给定的字符串 w 是否属于某个具体的语言 L,

$$w \in L?$$

- 任何所谓问题,都可以转为语言成员性的问题
- 语言和问题其实是相同的东西

形式化证明：演绎法，归纳法和反证法

例 1. 若 x 和 y 是 Σ 上的字符串，请证明 $|xy| = |x| + |y|$.

证明：通过对 $|y|$ 的归纳来证明

❶ 基础：当 $|y| = 0$，即 $y = \varepsilon$

$$
\begin{aligned}
|x\varepsilon| &= |x| && \text{连接的定义} \\
&= |x| + |\varepsilon| && \text{长度的定义}
\end{aligned}
$$

❷ 递推：假设 $|y| = n\ (n \geq 0)$ 时命题成立，
那么当 $|y| = n + 1$，即 $y = wa$

$$
\begin{aligned}
|x(wa)| &= |(xw)a| && \text{连接的定义} \\
&= |xw| + 1 && \text{长度的定义} \\
&= |x| + |w| + 1 && \text{归纳假设} \\
&= |x| + |wa| && \text{长度的定义} \quad \square
\end{aligned}
$$

形式化证明：演绎法，归纳法和反证法

例 1. 若 x 和 y 是 Σ 上的字符串，请证明 $|xy| = |x| + |y|$.

√ 结构

证明：通过对 y 的归纳来证明

❶ 基础：$y = \varepsilon$ 时

$$\begin{aligned} |x\varepsilon| &= |x| && \text{连接的定义} \\ &= |x| + |\varepsilon| && \text{长度的定义} \end{aligned}$$

❷ 递推：假设 $y = w \ (w \in \Sigma^*)$ 时命题成立，
那么当 $y = wa$ 时

$$\begin{aligned} |x(wa)| &= |(xw)a| && \text{连接的定义} \\ &= |xw| + 1 && \text{长度的定义} \\ &= |x| + |w| + 1 && \text{归纳假设} \\ &= |x| + |wa| && \text{长度的定义} \quad \square \end{aligned}$$