第 一 讲

✧ 课程概述及预备知识

形式语言与自动机

Formal Languages and Automata

课程概述及预备知识

- 有关信息
- 预备知识、记号
- 主要知识点预览
- 常用证明技术

有关信息

- 课程信息
- 课程性质
- 教学内容和目标
- 相关课程
- 教师信息
- 助教信息
- 教材
- 参考书目
- 课程网页
- 课程计划与进度
- 书面作业
- 考核计划
- 答疑与交流

课程信息

- 名称　　形式语言与自动机
- 类别　　必修
- 时间　　16-2-22 至 16-6-6
  每周一下午 1:30-3:05
- 教室　　六教 6C102
- 班级　　计 41-45
- 时数　　32-(2*?)

课程性质

- 计算机相关专业基础课
  - 上世纪 60 年代末、70年代初，研究的高峰
  - 之后，研究生的基础课程
  - 上世纪 90 年代后，本科阶段的专业基础课
- 专业工作者必须的理论素养
  - 形式化模型
    系统行为建模、模拟，模型检验
  - 计算理论基础
    可计算性，计算复杂性
  - 计算机系统及应用系统设计的基础
    形形色色 （编译系统，软硬件设计…）

教学内容和目标

- 主要讲授理论和应用中的常用语言类及其相应的计算模型，以及计算模型之间的联系
- 培养计算机科学理论方面的素养，提高逻辑思维和解决相关问题的能力，为后续专业课程的学习以及今后从事科学研究或技术开发工作打下扎实的基础

相 关 课 程

- 先修课程
  - 《离散数学》（数理逻辑，集合与代数，图论）
- 后续课程
  - 《编译原理》
- 其它相关课程
  - 《高级语言程序设计》
    《数字逻辑》
    《计算语言学》
    《可计算性与计算复杂性理论》
    …

教师信息

- 姓名　王生原
- 单位　计算机系软件技术研究所
- 电话　62794240（O） 13366102912
- 办公室　东主楼 10 区 209
- 电子信箱　wwssyy@tsinghua.edu.cn
- 研究领域
  - 程序设计语言理论与实现
  - 并发程序设计(方法与模型)
  - 程序验证

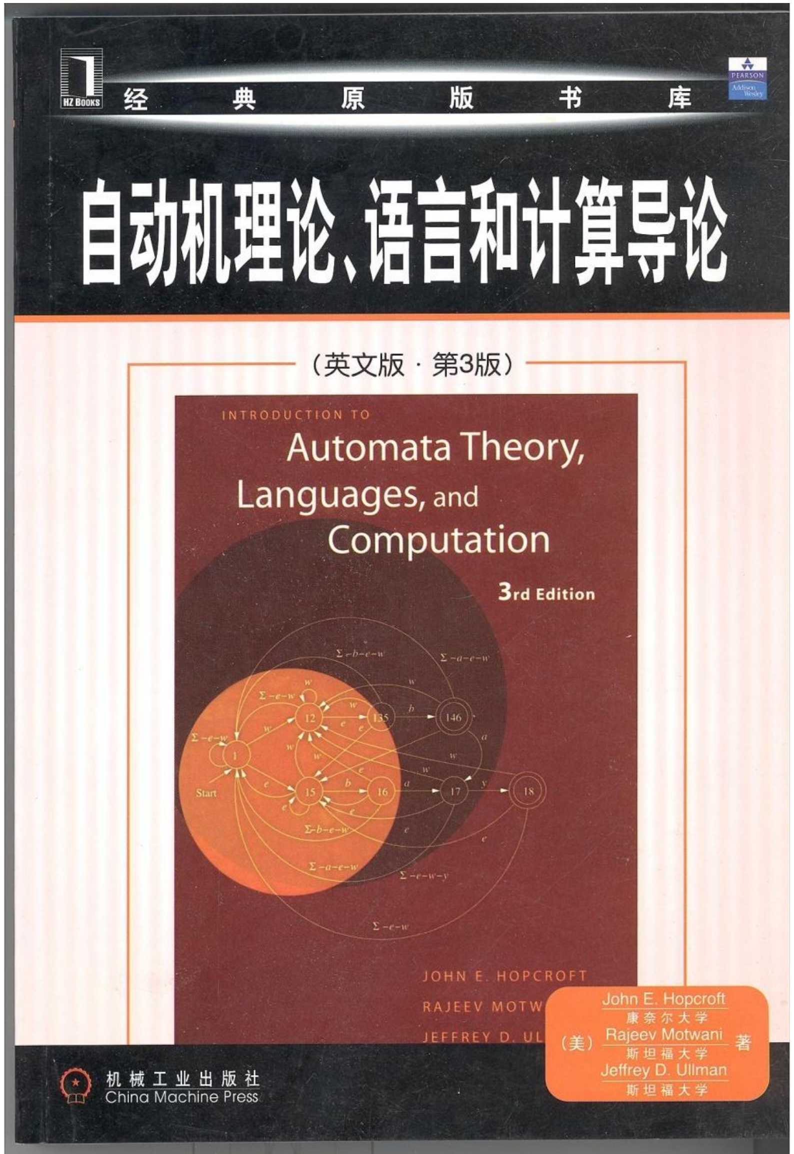
助教信息

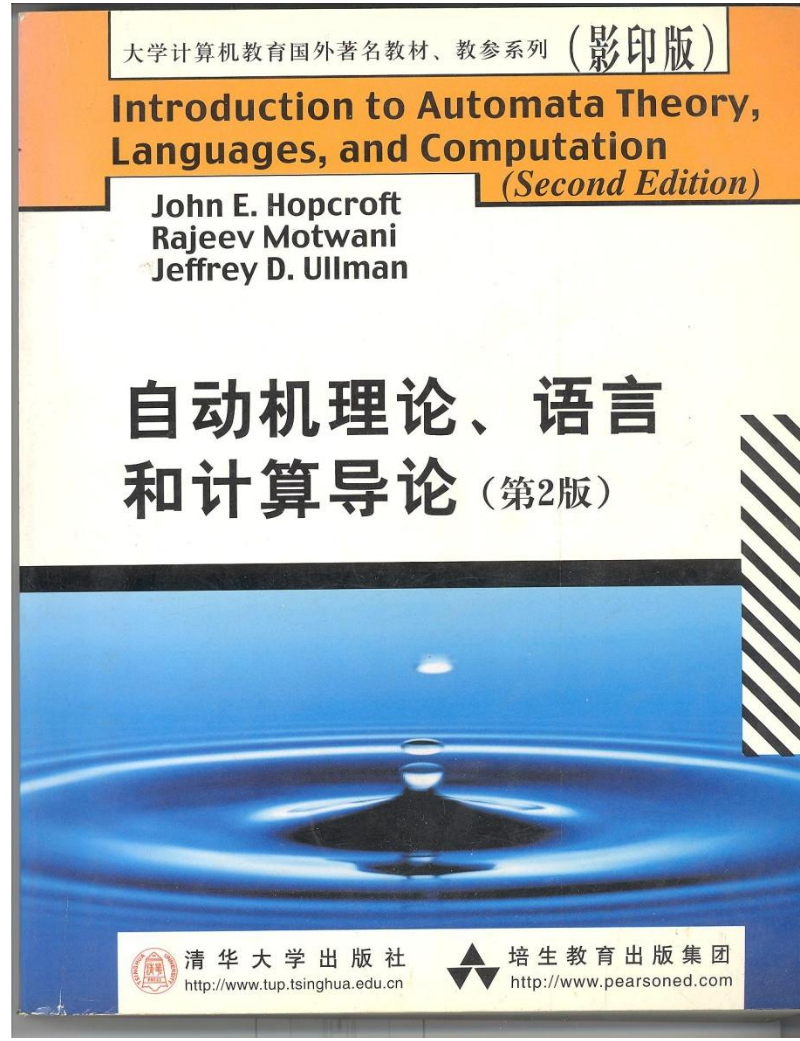
- 姓名　　崔文凯
- 单位　　计算机系高性能研究所
- 电话　　62785592
- 答疑时间　待定
- 答疑地点　待定
- 网上答疑　网络学堂
- 电子信箱　cuiwk14@mails.tsinghua.edu.cn

助教信息

- 姓名　吴波
- 单位　计算机系高性能研究所
- 电话　62785592
- 答疑时间　待定
- 答疑地点　待定
- 网上答疑　网络学堂
- 电子信箱　xiongx05@gmail.com

教 材


经 典 原 版 书 库
自动机理论、语言和计算导论
（英文版 · 第3版）
INTRODUCTION TO
Automata Theory,
Languages, and
Computation
3rd Edition
JOHN E. HOPCROFT
RAJEEV MOTW
JEFFREY D. UL
John E. Hopcroft
康奈尔大学
（美） Rajeev Motwani 著
斯坦福大学
Jeffrey D. Ullman
斯坦福大学
机械工业出版社
China Machine Press


大学计算机教育国外著名教材、教参系列（影印版）
Introduction to Automata Theory,
Languages, and Computation
(Second Edition)
John E. Hopcroft
Rajeev Motwani
Jeffrey D. Ullman
自动机理论、语言
和计算导论（第2版）
清 华 大 学 出 版 社
http://www.tup.tsinghua.edu.cn
培 生 教 育 出 版 集 团
http://www.pearsoned.com

教 材

- ✧ 书名 Introduction to Automata Theory, Languages, and Computation
- ✧ 作者 John E. Hopcroft （Cornell）
  Rajeev Motwani （Stanford）
  Jefferey D. Ullman （Stanford）
- ✧ 出版社 Addison Wesley

  清华大学出版社影印（第 2 版），2001

  机械工业出版社影印（第 3 版），2008

John.E.Hopcroft, the Turing Award winner in 1986.

参考书目

- 中译本 Second Edition 2004 Third Edition 2008

  机械工业出版社，北京

- 《An Introduction to Formal Lanhuages and Automata》

  Peter Linz Third Edition 2001（Jones&Bartlett）

  机械工业出版社影印，2004 中译本 2005

- 《形式语言与自动机》

  陈有祺 编著 机械工业出版社，2008

- 《形式语言与自动机理论》

  蒋宗礼，姜守旭 编著

  清华大学出版社，北京，2003

课 程 网 页

- 清华网络学堂

  http://learn.tsinghua.edu.cn

课程计划与进度

- 课时安排（粗略）
  - 课程概况及预备知识　2 学时
  - 有限状态自动机，正规语言，正规表达式

    第 2，3，4 章，约 10 学时
  - 上下文无关文法，上下文无关语言，下推自动机

    第 5，6，7 章，约 10 学时
  - 图灵机，计算理论初步

    第 8 章，约 3 学时

    第 9，10，11 章，约 2 学时

书面作业

- ✧ 随堂布置
  - – 以课本中的练习为主

    标记：*，!，!!
  - – 思考题
  - – 自测题
- ✧ 抽查完成情况
- ✧ 按时完成

考核计划

- 总评成绩　（100%）
  - 期中考试　（20%）
  - 平时成绩　（10%）

    书面作业（6%）与平时表现（4%）
  - 期末考试　（70%）

答疑与交流

- 通过网络
  - 清华网络学堂（课程讨论区）

    问题探讨，交流解题技巧
  - 电子邮件 wwssyy@tsinghua.edu.cn
- 面对面
  - 时间预约

    第 2 – 16 周上班时间（节假日除外）
  - 地点

    东主楼 10 区 209 室

预备知识、记号

- ✧ 字母表
- ✧ 字符串
- ✧ 关于字符串的运算
- ✧ 字母表上的运算
- ✧ 语言
- ✧ 关于语言的运算

字母表（Alphabet）

- 概念　形式符号的非空有限集合
- 记号　常用 Σ 表示
- 举例
  - 英文字母表 { *a*, *b*, …, *z*, *A*, *B* , …, *Z* }
  - 英文标点符号表 { , ; : . ? ! ’ ‘ “ ” ( ) [ ] – - … }
  - 汉字表 { …, 自, …, 动 , …, 机, … }
  - 化学元素表 { H, He, Li, …, Une }
  - Σ = { *a*, *n*, *y*, 任,意 }

字符串（string）

✧ **概念** 字母表 Σ 上的一个字符串（串），或称为字（*word*），为 Σ 中字符构成的一个有限序列。空串（*empty string*），常用 ε 表示，不包含任何字符。

举例 设 $\Sigma = \{ a, b \}$，则 $\varepsilon, a, aaa, baba$ 等都是串

✧ 字符串 w 的长度，记为 $|w|$，是包含在 w 中字符的个数

举例 $|\varepsilon| = 0$，$|bbaba| = 5$

关于字符串的运算

✧ 连接（**concatenation**）

设 x, y为串，且 $x = a_1a_2 \ldots a_m$, $y = b_1b_2 \ldots b_n$,
则 x 与 y 的连接

$$xy = a_1a_2 \ldots a_m b_1b_2 \ldots b_n$$

✧ 连接运算的性质

- $(xy)z = x(yz)$
- $\varepsilon x = x \varepsilon = x$
- $|xy| = |x| + |y|$

字母表上的运算

✧ 幂运算

设 Σ 为字母表，n 为任意自然数，定义

（1）$\Sigma^0 = \{ \varepsilon \}$

（2）设 $x \in \Sigma^{n-1}$，$a \in \Sigma$， 则 $ax \in \Sigma^n$

（3）Σ^n 中的元素只能由（1）和（2）生成

举例　设 $\Sigma = \{ 0, 1 \}$，则

$\Sigma^0 = \{ \varepsilon \}$， $\Sigma^1 = \{ 0, 1 \}$，

$\Sigma^2 = \{ 00, 01, 10, 11 \}$， …

字母表上的运算

- ✧ ＊闭包 $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots$
- ✧ ＋闭包 $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots$
- ✧ $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$， $\Sigma^+ = \Sigma^* - \{\varepsilon\}$ ？
- ✧ 举例 设 $\Sigma = \{0, 1\}$，则

  $\Sigma^+ = \{0, 1, 00, 01, 10, 11, \ldots\}$

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

语 言

- ✧ 概念 设 Σ 为字母表，则任何集合 $L \subseteq \Sigma^*$ 是字母表 Σ上的一个语言（*language*）
- ✧ 举例
  - – 英文单词集 {*…, English, …, words , …* }
  - – C++ 语言程序集 { … } 字母表?
  - – 汉语四字成语集 { …, 语不惊人, … , 言之有物, … }
  - – 化学分子式集{ …, H_2O, …, *NaCl* , … } 字母表?
  - – {*any*, 任意}
- ✧ 比较 空语言 φ 与仅含空字的语言 { ε }

关于语言的运算

✧ 两个语言 L 和 M 的并（*union*）

$$L \cup M = \{ w \mid w \in L \vee w \in M \}$$

✧ 举例

设 $L = \{ 001, 10, 111 \}$, $M = \{\varepsilon, 001\}$, 则

$$L \cup M = \{ \varepsilon, 10, 001, 111 \}$$

关于语言的运算

✧ 两个语言 L 和 M 的连接（*concatenation*）

$$L \cdot M = \{w_1 w_2 \mid w_1 \in L \wedge w_2 \in M\}$$

✧ 通常记 $L \cdot M$ 为 LM

✧ 举例

设 $L = \{ 001, 10, 111 \}$, $M = \{\varepsilon, 001\}$,则

$LM = \{ 001, 10, 111, 001001, 10001, 111001 \}$

关于语言的运算

- 语言 *L* 的闭包（*closure*）

$L^* = L^0 \cup L^1 \cup L^2 \cup \ldots = \cup_{i \geq 0} L^i$，其中

$L^0 = \{\varepsilon\}$， $L^1 = L$, $L^2 = LL$, …

$L^n = L^{n-1}L$

- 举例

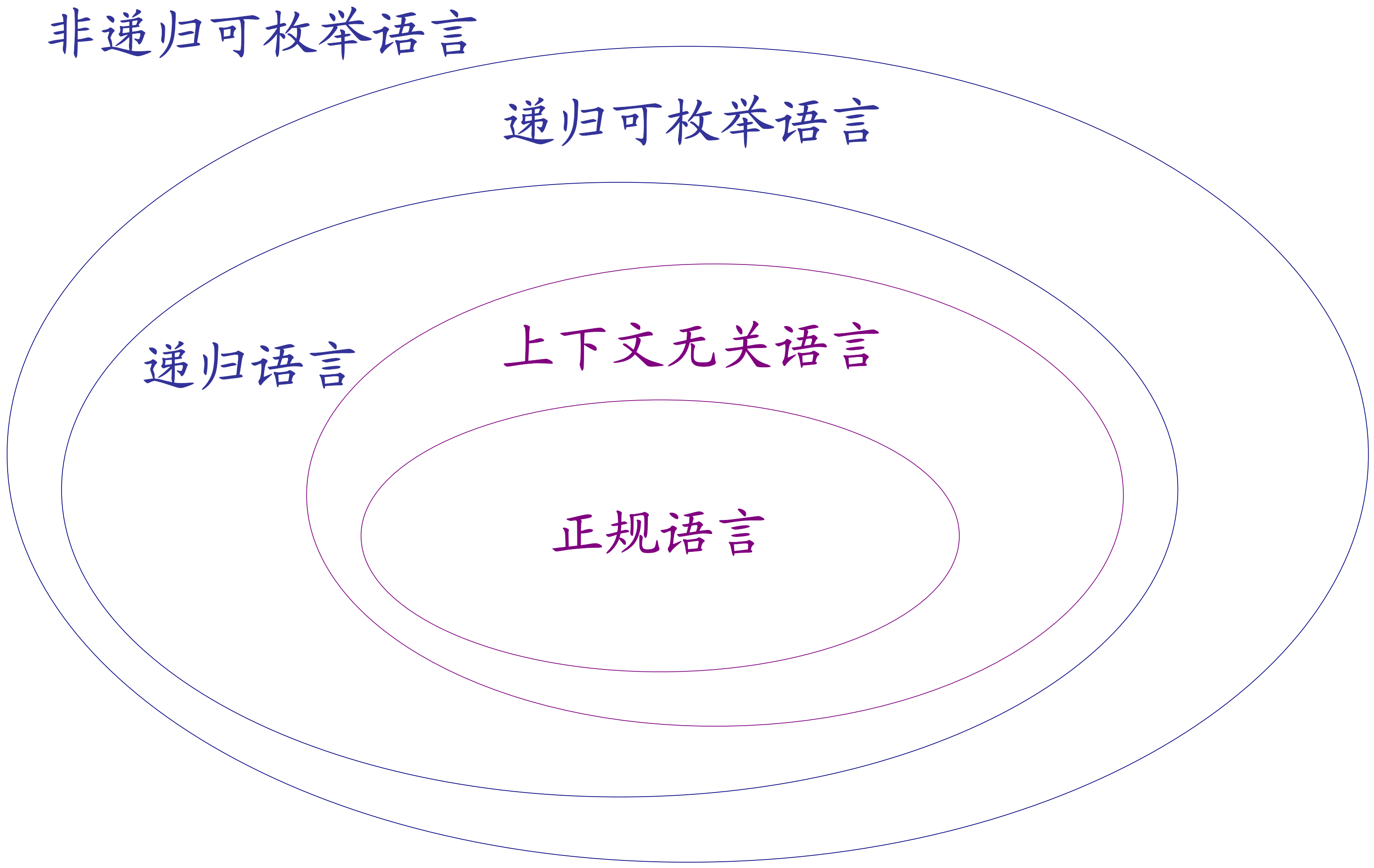
设 $L = \{ 0, 11 \}$，则

$L^* = \{ \varepsilon, 0, 11, 00, 011, 110, 1111, 000, 0011, 0110, 01111, 1100, 11011, 11110, 111111, \ldots \}$

主要知识点预览

- 课程涉及的语言
- 正规语言与有限自动机
- 正规语言与正规表达式
- 正规语言的性质与运算
- 上下文无关语言与上下文无关文法
- 上下文无关语言与下推自动机
- 上下文无关正规语言的性质与运算
- 图灵机及其语言
- 计算理论初步

课程涉及的语言



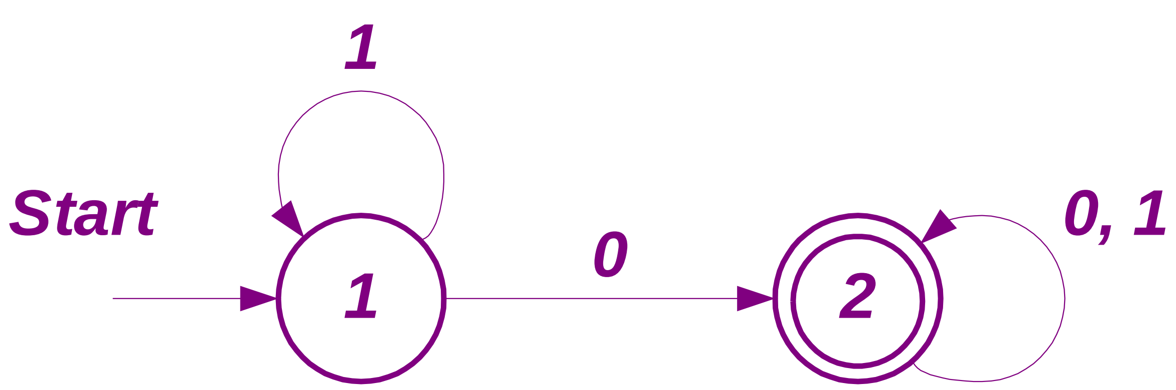
正视语言与有限自动机

设 $\Sigma = \{ 0, 1 \}$,

$L = \{ w \mid w$ 中至少有一个$0\}$,

如 $0011, 10, 110111 \in L$, 而$11, \varepsilon, 1111 \notin L$。

如下是一个定义该语言的有限状态自动机:



正规语言与正规表达式

设 $\Sigma = \{ 0, 1 \}$,

$L = \{ w \mid w$ 中至少有一个0},

如 0011, 10, 110111 $\in L$, 而11, ε, 1111 $\notin L$。

如下是一个定义该语言的正规表达式:

$$1^* 0 (0 + 1)^*$$

正规语言的性质与运算

- 正规语言
（ *regular languages*）
  - 正规语言的特性
  - 有关正规语言的判定性质
  - 正规语言的运算性质
  - 重要的算法

上下文无关语言与上下文无关文法

设 $\Sigma = \{ 0, 1 \}$,

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

如 $0011, 000111, 01 \in L$; $10, 1001, \varepsilon, 010 \notin L$.

如下是一个可接受该语言的上下文无关文法:

$$S \to 01$$
$$S \to 0S1$$

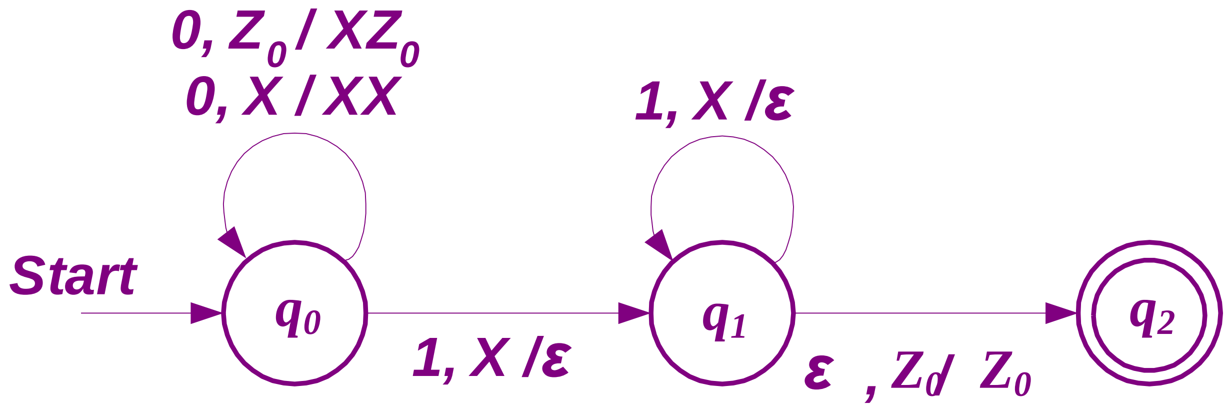
但没有任何有限自动机能够接受语言L.

设 $\Sigma = \{ 0, 1 \}$,

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

如 $0011, 000111, 01 \in L$; $10, 1001, \varepsilon, 010 \notin L$.

如下是一个可接受该语言的一个下推自动机:



上下文无关语言的性质与运算

上下文无关语言

context-free languages

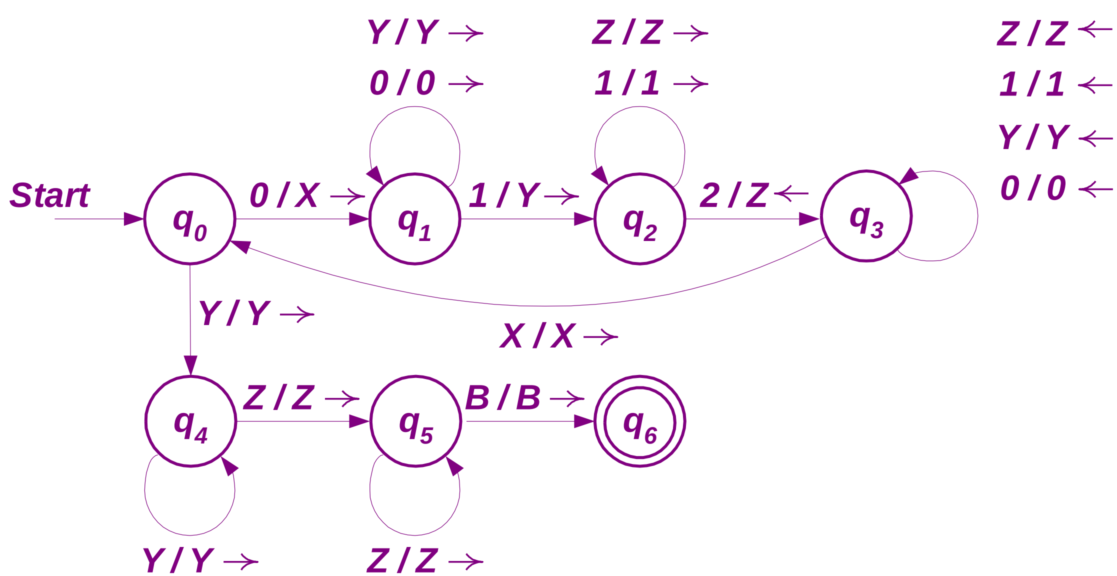
- 上下文无关语言的特性
- 有关上下文无关语言的判定性质
- 有关上下文无关语言的运算性质
- 有关的重要算法

图灵机及其语言

设 $\Sigma = \{\ 0, 1, 2\ \}$， $L = \{\ 0^n 1^n 2^n \mid n \geq 1 \}$，如 $012, 001122 \in L$, 而 $10, 1001$, ε, $010 \notin L$.

没有任何有限自动机和上下文无关文法能够接受语言 L.

存在一个图灵机可接受该语言.



计算理论初步

- ✧ 可计算性
  - – 如何定义可计算性
  - – 如何比较计算能力
  - – 普通计算机的计算能力
  - – 问题的可判定与不可判定性（是否存在算法）
- ✧ 计算复杂性
  - – *P* 问题与 *NP* 问题
  - – *NP*-完全问题与 *NP*-难问题

常用证明技术

- ✧ 基本证明方法
- ✧ 归纳证明方法

演 绎 证 明

✧ 概念 一个证明（*proof*）是命题的序列，其中的每一个命题或者是已知的命题，或者是由前面出现过的命题使用逻辑公理和规则得出.

已知的命题集合称为假设（*hypothesis*）或前提（*premise*），最后一个命题称为该前提的结论（*conclusion*）.

基本证明方法

✧ "If –Then" 命题

– 证明方法

把 If 部分作为已知的命题，把 Then 部分作为结论.

– 举例　如果 $x+y=1$，那么 $x^2-y^2=x-y$.

证明:

1　$x^2-y^2 = (x+y)(x-y)$　// 数学公理

2　$(x+y) = 1$　// 已知

3　$x^2-y^2 = x-y$　// 由1、2 和算术性质推出

基本证明方法

- “*If-And-Only-If*” 命题
  - 证明方法

    欲证 *A if and only if B*，可分别证明如下两个命题:

    1 *if A then B*，

    2 *if B then A*.

基本证明方法

✧ 有关集合的命题

设 R, S 为集合.

– 欲证 $R \subseteq S$，可证明如下命题:

$$\text{if } x \in R \text{ then } x \in S.$$

– 欲证 $R = S$，可分别证明如下两个命题:

$$1 \quad \text{if } x \in R \text{ then } x \in S$$

$$2 \quad \text{if } x \in S \text{ then } x \in R$$

基本证明方法

✧ 原命题的逆否命题

有时，证明原命题的逆否（contrapositive）命题更加方便.

– 欲证 *if A then B*，可证明如下命题:

if not B then not A.

基本证明方法

- 反证法（*proof by contradiction*）
  - 欲证 *if H then C*，可以把 *H* 和 *not C* 都作为已知的命题，把任何一个矛盾（*contradiction*）命题作为新的结论.

基本证明方法

- 举例证明或否证
  - 举例证明存在量化的命题

    如命题：存在整数 a，满足 $a^2 = 2^a$.

    证明：取 $a = 2$，满足 $a^2 = 2^a$.
  - 举反例否定全称量化的命题

    如命题：所有整数 a，都满足 $a2=2a$.

    否证：取 $a = 1$，不满足 $a^2 = 2^a$.

归纳证明方法

✧ 归纳定义

– 集合的归纳定义

由 3 部分构成:

1 基础（*basis*） // 直接定义集合中的元素（至少1个）

2 归纳（*induction*）// 从已知元素生成新元素的规则

3 极小性限制 // 申明集合中的元素只能由 1、2 生成

归纳证明方法

✧ 结构归纳法

对于归纳定义的集合 S，欲证: 对于任意 $x \in S$，满足性质 $P(x)$.

1 基础（*basis*）　// 若有直接定义 $a \in S$，则证明 $P(a)$

2 归纳（*induction*）// 若归纳定义中有规则

if $a_1, a_2, \ldots, a_n \in S$

then $f(a_1, a_2, \ldots, a_n) \in S$，

则证明

if $P(a_1), P(a_2), \ldots, P(a_n)$

then $P(f(a_1, a_2, \ldots, a_n))$

✧ 归纳定义（例）

– 归纳定义合法括号串的集合 **S**

1 基础　空串 $\varepsilon \in S$

2 归纳　若 $x \in S$，则 $(x) \in S$；
若 $x,y \in S$，则 $xy \in S$.

3 极小性限制

S 中的元素只能由 1、2 生成
或
S 是满足 1、2 的最小集合)

归纳证明方法

✧ 结构归纳证明（例）

– 命题：合法括号串集合 S 中每个括号串的“(”与“)”数目相等

证明：

1 基础

空串 ε 的“(”与“)”数目相等，都为0;

2 归纳

设 x,y 的“(”与“)”数目相等，前者为 m，后者为 n；

(x) 的“(”与“)”数目都为 m+1;

xy 的“(”与“)”数目都为 m+n.

归纳证明方法

✧ 基于自然数的归纳（一般数学归纳法）

– **自然数** 自然数集合 N 是满足如下条件的最小集合:

(1) $0 \in N$;
(2) 若 $n \in N$, 则n的后继 $n+1 \in N$

– **数学归纳法** 欲证对任意自然数 n，$P(n)$成立，

(1) 先证$P(0)$ 成立;
(2) 再证若 $P(n)$成立,则$P(n+1)$成立

– **另一种形式**

(1) 先证$P(0)$ 成立;
(2) 再证若对任意$k<n$，$P(k)$ 成立, 则$P(n)$成立

– 对任何良序集合，都可以有这两种形式

归纳证明方法

✧ 互归纳（*Mutual Induction*）证明法

- 为方便起见，归纳证明一个性质时有时需要证明相关的一组性质，它们能够相互配合，利于证明.

  比如自然数集合上有关系密切的性质

  $P_1(n)$，$P_2(n)$，…，$P_m(n)$.

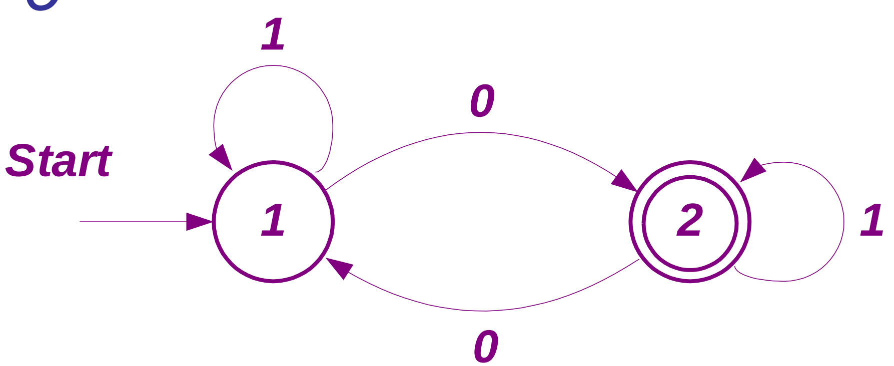
  若证明性质 $H(n) = P_1(\mathrm{n}) \wedge P_2(\mathrm{n}) \wedge \ldots \wedge P_m(n)$ 比证明性质 $P_1(\mathrm{n})$ 容易很多。这样，可以通过证明 $H(n)$，即同时证明 $P_1(n)$，$P_2(n)$，…，$P_m(n)$, 来证明性质 $P_1(\mathrm{n})$.

- 参考 *Example1.23*

课后练习

✧ 思考题（不用交）

采用互归纳法证明下面的自动机接受的串具有奇数个 0



提示 可用互归纳法证明如下两个命题:

（1）对任何串$w \in \{0,1\}^*$，若上述自动机读入 w 后处于状态1，则w 一定含有偶数个0;

（2）对任何串$w \in \{0,1\}^*$，若上述自动机读入 w 后处于状态2，则w 一定含有奇数个0.

课后练习

✧ 自测题

– 设L为任何语言，记 $L^+ = L^1 \cup L^2 \cup \ldots \cup L^n \cup \ldots$, $n \geq 1$。

试问：在什么条件下，$L^+ = L^* - \{\varepsilon\}$？

That's all for today.

Thank You