人工智能数学基础

预备知识 有穷自动机

杨雅君

yjyang@tju.edu.cn

天津大学 智能与计算学部

2024

内容提要

内容提要

非形式化描述

非形式化描述

- 指针式钟表

非形式化描述

- 指针式钟表
  - $12 \times 60 \times 60 = 43200$个状态

非形式化描述

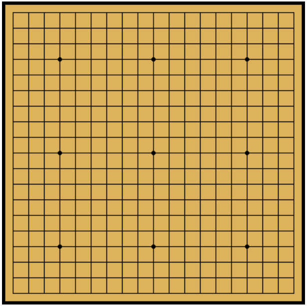
有穷多个状态

- 指针式钟表
    - $12 \times 60 \times 60 = 43200$个状态

非形式化描述

有穷多个状态、

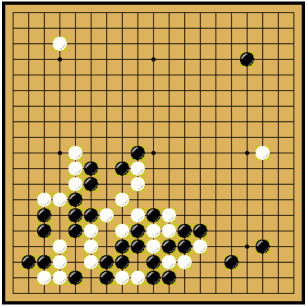
- 一局围棋



非形式化描述

有穷多个状态、

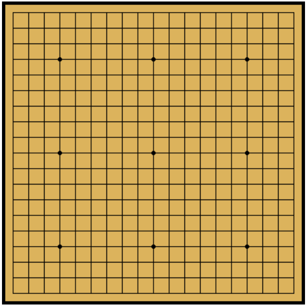
- 一局围棋
  - 3^{361}个状态



非形式化描述

有穷多个状态、初始状态

- 一局围棋
  - 3^{361}个状态



非形式化描述

有穷多个状态、初始状态

- 电梯的控制结构
  - 每层一个状态

非形式化描述

有穷多个状态、初始状态

- 电梯的控制结构
  - 每层一个状态

- 状态转移:

非形式化描述

有穷多个状态、初始状态

- 电梯的控制结构
  - 每层一个状态

- 状态转移：当前状态 + 输入信号 $\longrightarrow$ 下一状态

非形式化描述

有穷多个状态、输入信号、状态转移、初始状态

- 电梯的控制结构
    - 每层一个状态

- 状态转移：当前状态 + 输入信号 $\longrightarrow$ 下一状态

非形式化描述

小结：有穷状态系统四要素

1. 有穷多个状态
2. 输入信号
3. 状态转移
4. 初始状态

内容提要

1 非形式化描述

2 有穷自动机的定义

3 有穷自动机接受的语言

有穷自动机的定义

定义 3.1

一个有穷自动机（Finite Automata，简称 FA ）是一个五元组

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

有穷自动机的定义

定义 3.1

一个有穷自动机（Finite Automata，简称 FA ）是一个五元组

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

1. Q 是 有穷状态集

有穷自动机的定义

定义 3.1

一个**有穷自动机**（Finite Automata，简称 FA ）是一个**五元组**

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

1. Q 是 **有穷状态集**
2. Σ 是有穷的 **输入字母表**

有穷自动机的定义

定义 3.1

一个**有穷自动机**（Finite Automata，简称 FA ）是一个**五元组**

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

1. Q 是 **有穷状态集**
2. Σ 是有穷的 **输入字母表**
3. δ 是 **转移函数**，即映射 $\delta : Q \times \Sigma \rightarrow Q$

有穷自动机的定义

定义 3.1

一个**有穷自动机**（Finite Automata，简称 **FA** ）是一个**五元组**

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

1. Q 是 **有穷状态集**
2. Σ 是有穷的 **输入字母表**
3. δ 是 **转移函数**，即映射 $\delta : Q \times \Sigma \to Q$
4. $q_0 \in Q$ 是 **初始状态**

有穷自动机的定义

定义 3.1

一个有穷自动机（Finite Automata，简称 FA ）是一个五元组

$$M = (Q, \Sigma, \delta, q_0, F)$$

其中：

1. Q 是 有穷状态集
2. Σ 是有穷的 输入字母表
3. δ 是 转移函数，即映射 $\delta : Q \times \Sigma \to Q$
4. $q_0 \in Q$ 是 初始状态
5. $F \subseteq Q$ 是 接受状态集

有穷自动机的定义：举例

例 3.1

有穷自动机的一个实例

$$M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

有穷自动机的定义：举例

例 3.1

有穷自动机的一个实例

$$M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

其中，**转移函数 δ** 定义如下：

有穷自动机的定义：举例

例 3.1

有穷自动机的一个实例

$$M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

其中，**转移函数** δ 定义如下：

$$\begin{aligned}
\delta(q_0, 0) &= q_2, & \delta(q_0, 1) &= q_1 \\
\delta(q_1, 0) &= q_3, & \delta(q_1, 1) &= q_0 \\
\delta(q_2, 0) &= q_0, & \delta(q_2, 1) &= q_3 \\
\delta(q_3, 0) &= q_1, & \delta(q_3, 1) &= q_2
\end{aligned}$$

有穷自动机的定义：举例

例 3.1

有穷自动机的一个实例

$$M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

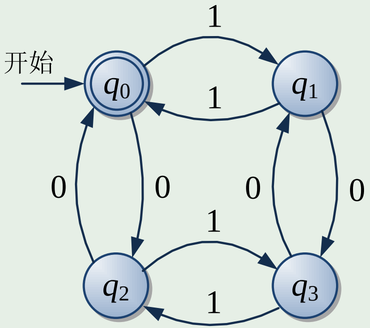
其中，**转移函数 δ** 定义如下：

$$\begin{aligned} \delta(q_0, 0) &= q_2, & \delta(q_0, 1) &= q_1 \\ \delta(q_1, 0) &= q_3, & \delta(q_1, 1) &= q_0 \\ \delta(q_2, 0) &= q_0, & \delta(q_2, 1) &= q_3 \\ \delta(q_3, 0) &= q_1, & \delta(q_3, 1) &= q_2 \end{aligned}$$

最关键的部分：**转移函数**

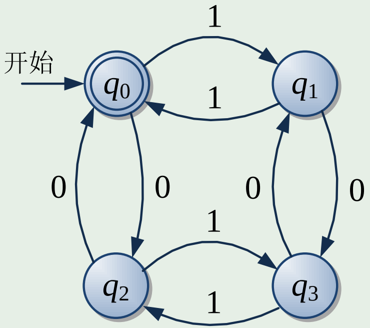
有穷自动机的定义：转移图

例 3.1



有穷自动机的定义：转移图

例 3.1



转移图表达了五元组的全部信息

有穷自动机的定义：扩充转移函数

定义 3.2

对于有穷自动机 $M = (Q, \Sigma, \delta, q_0, F)$

- 扩充转移函数 $\hat{\delta}$ 为映射 $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$

具体定义如下：

有穷自动机的定义：扩充转移函数

定义 3.2

对于有穷自动机 $M = (Q, \Sigma, \delta, q_0, F)$

- 扩充转移函数 $\hat{\delta}$ 为映射 $\hat{\delta} : Q \times \Sigma^* \to Q$

具体定义如下：

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

其中，$q \in Q$，$a \in \Sigma$，$w \in \Sigma^*$

有穷自动机的定义：扩充转移函数

定义 3.2

对于有穷自动机 $M = (Q, \Sigma, \delta, q_0, F)$

- **扩充转移函数** $\hat{\delta}$ 为映射 $\hat{\delta} : Q \times \Sigma^* \to Q$

具体定义如下：

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

其中，$q \in Q$，$a \in \Sigma$，$w \in \Sigma^*$

递归定义

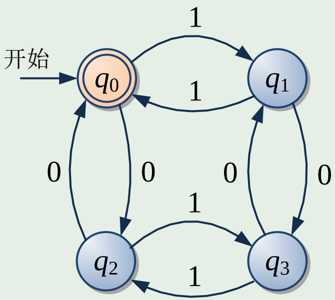
将原来 δ 中的第二个变元由一个字符扩充为一个字符串

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

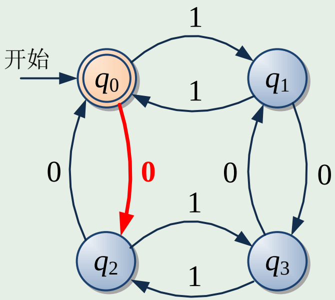
$\hat{\delta}(q_0, 010) =$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

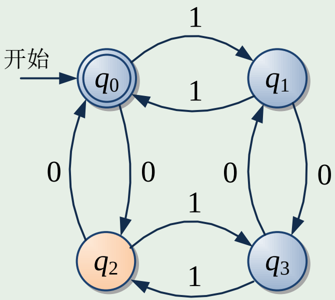
$$\hat{\delta}(q_0, 010) = \hat{\delta}(\delta(q_0, 0), 10)$$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

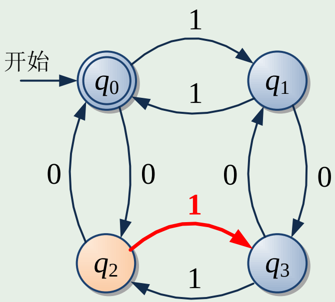
$$\hat{\delta}(q_0, 010) = \hat{\delta}(\delta(q_0, 0), 10)$$
$$= \hat{\delta}(q_2, 10)$$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

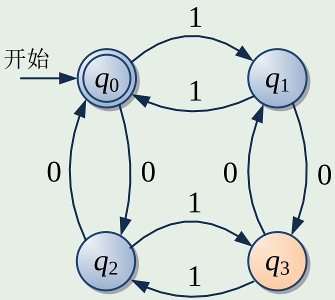
$$\hat{\delta}(q_0, 010) = \hat{\delta}(\delta(q_0, 0), 10)$$
$$= \hat{\delta}(q_2, 10)$$
$$= \hat{\delta}(\delta(q_2, 1), 0)$$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

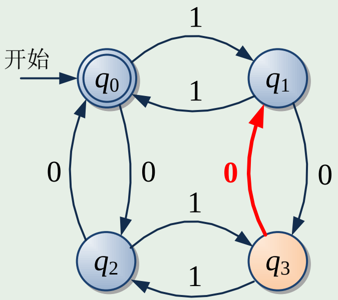
$$
\begin{aligned}
\hat{\delta}(q_0, 010) &= \hat{\delta}(\delta(q_0, 0), 10) \\
&= \hat{\delta}(q_2, 10) \\
&= \hat{\delta}(\delta(q_2, 1), 0) \\
&= \hat{\delta}(q_3, 0)
\end{aligned}
$$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



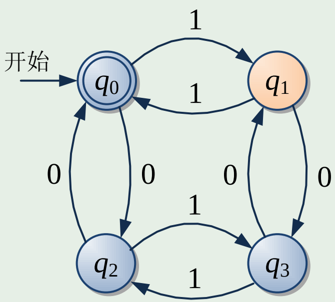
计算

$$
\begin{aligned}
\hat{\delta}(q_0, 010) &= \hat{\delta}(\delta(q_0, 0), 10) \\
&= \hat{\delta}(q_2, 10) \\
&= \hat{\delta}(\delta(q_2, 1), 0) \\
&= \hat{\delta}(q_3, 0) \\
&= \hat{\delta}(\delta(q_3, 0), \varepsilon)
\end{aligned}
$$

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

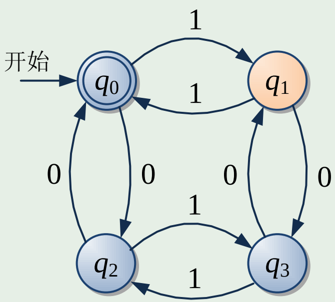
$$
\begin{aligned}
\hat{\delta}(q_0, 010) &= \hat{\delta}(\delta(q_0, 0), 10) \\
&= \hat{\delta}(q_2, 10) \\
&= \hat{\delta}(\delta(q_2, 1), 0) \\
&= \hat{\delta}(q_3, 0) \\
&= \hat{\delta}(\delta(q_3, 0), \varepsilon) \\
&= \hat{\delta}(q_1, \varepsilon)
\end{aligned}
$$

扩充转移函数：举例

1. $\hat{\delta}(q, \varepsilon) = q$
2. $\hat{\delta}(q, aw) = \hat{\delta}(\delta(q, a), w)$

例

对于例3.1的FA，



计算

$$
\begin{aligned}
\hat{\delta}(q_0, 010) &= \hat{\delta}(\delta(q_0, 0), 10) \\
&= \hat{\delta}(q_2, 10) \\
&= \hat{\delta}(\delta(q_2, 1), 0) \\
&= \hat{\delta}(q_3, 0) \\
&= \hat{\delta}(\delta(q_3, 0), \varepsilon) \\
&= \hat{\delta}(q_1, \varepsilon) \\
&= q_1
\end{aligned}
$$

扩充转移函数：含义

$\hat{\delta}(q, x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q, x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q, x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q, x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q, x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q,x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x|=1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

扩充转移函数：含义

$\hat{\delta}(q,x)$ 的值：

- 从状态 q 出发，用基本转移函数 δ
- 每越过 x 的一个符号后，改变一次状态
- 直到越过 x 的最后一个符号所得到的状态

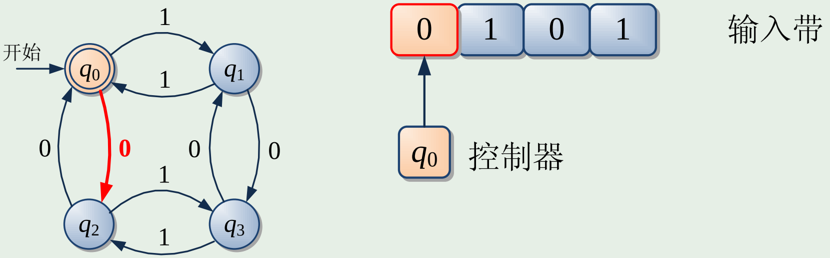
从 δ 到 $\hat{\delta}$ 的扩充很自然

- δ 就是 $\hat{\delta}$ 的特例（当 $|x| = 1$ 时）
- 今后不再强调 $\hat{\delta}$，而一律用 δ 表示

有穷自动机的模型：有穷控制器

例

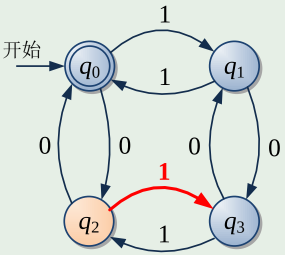
$$\delta(q_0, 0) = q_2$$

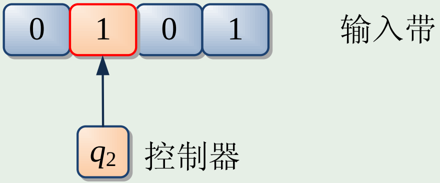


有穷自动机的模型：有穷控制器

例

$$\delta(q_2, 1) = q_3$$

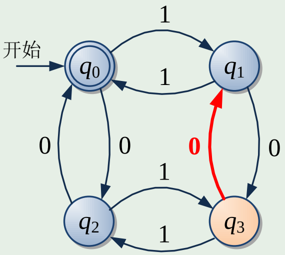


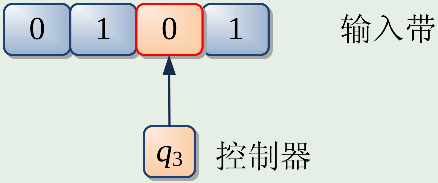


有穷自动机的模型：有穷控制器

例

$$\delta(q_3, 0) = q_1$$

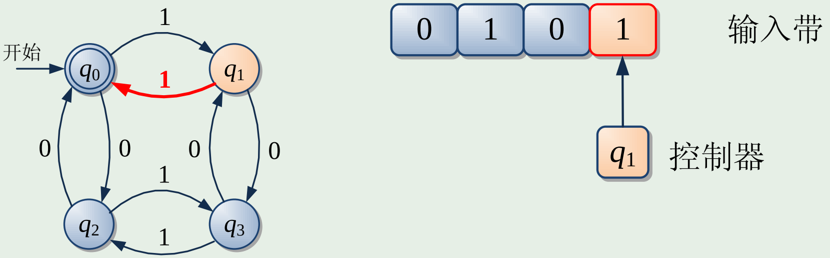




有穷自动机的模型：有穷控制器

例

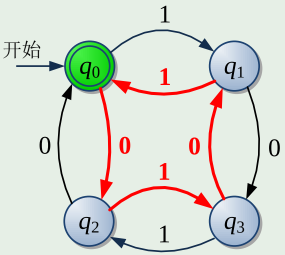
$$\delta(q_1, 1) = q_0$$

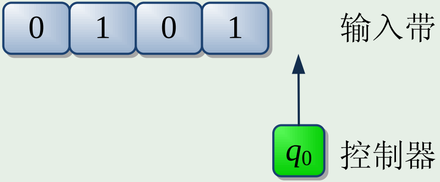


有穷自动机的模型：有穷控制器

例

$$\delta(q_0, 0101) = q_0$$

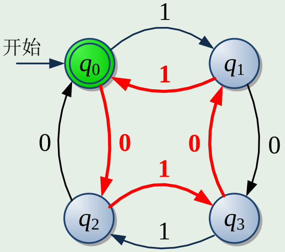


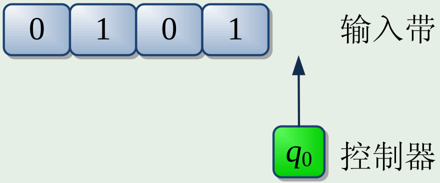


有穷自动机的模型：有穷控制器

例

$$\delta(q_0, 0101) = q_0$$



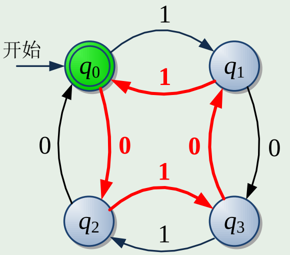


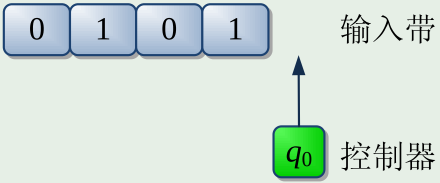
- 输入串 和 FA 是什么关系？

有穷自动机的模型：有穷控制器

例

$$\delta(q_0, 0101) = q_0$$





- 输入串 和 FA 是什么关系？ **接受状态集 F 所起的作用**

内容提要

有穷自动机接受的语言

定义 3.3

给出FA $M = (Q, \Sigma, \delta, q_0, F)$

$$\text{若 } \delta(q_0, x) = p \in F \quad (x \in \Sigma^*)$$

则称字符串 x 被 M **接受**

有穷自动机接受的语言

定义 3.3

给出FA $M = (Q, \Sigma, \delta, q_0, F)$

$$\text{若 } \delta(q_0, x) = p \in F \quad (x \in \Sigma^*)$$

则称字符串 x 被 M **接受**

- 被 M 接受的全部字符串的集合，称为 M **接受的语言**，记作 $L(M)$

$$L(M) = \{x \mid \delta(q_0, x) \in F\}$$

有穷自动机接受的语言

定义 3.3

给出FA $M = (Q, \Sigma, \delta, q_0, F)$

$$\text{若 } \delta(q_0, x) = p \in F \quad (x \in \Sigma^*)$$

则称字符串 x 被 M **接受**

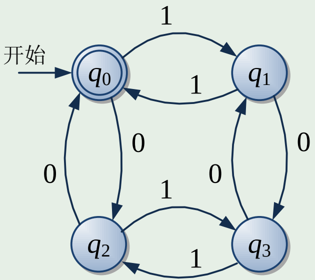
- 被 M 接受的全部字符串的集合，称为 M **接受的语言**，记作 $L(M)$

$$L(M) = \{x \mid \delta(q_0, x) \in F\}$$

- 刻画了FA和语言的关系

例

例3.1中FA

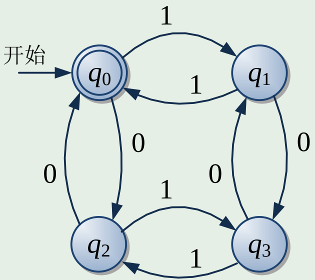


接受什么样的语言？

FA $\Longrightarrow$ 语言：举例

例

例3.1中FA

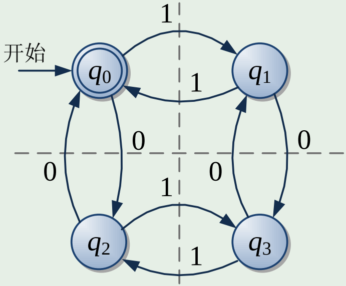


接受什么样的语言？

分析：4个状态起什么作用？

例

例3.1中FA



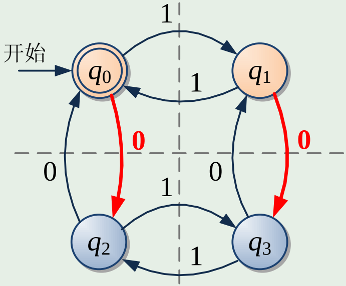
接受什么样的语言？

分析：4个状态起什么作用？

FA $\Longrightarrow$ 语言：举例

例

例3.1中FA



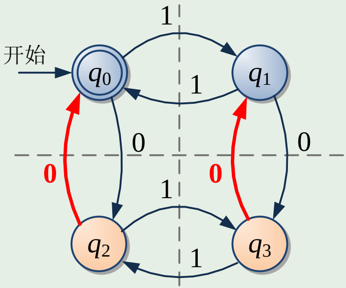
接受什么样的语言？

分析：4个状态起什么作用？

FA $\Longrightarrow$ 语言：举例

例

例3.1中FA

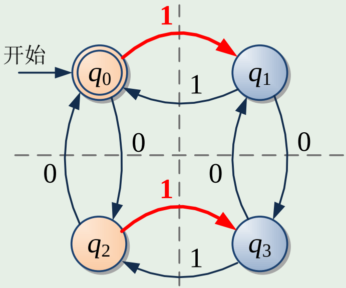


接受什么样的语言？

分析：4个状态起什么作用？

例

例3.1中FA

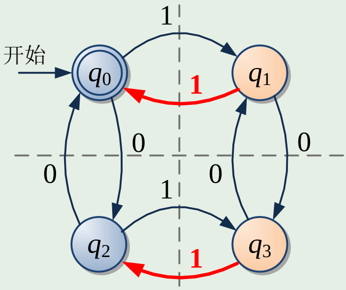


接受什么样的语言？

分析：4个状态起什么作用？

例

例3.1中FA



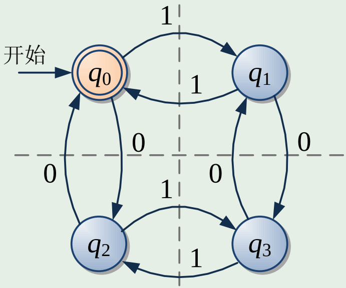
接受什么样的语言？

分析：4个状态起什么作用？

FA $\Longrightarrow$ 语言：举例

例

例3.1中FA



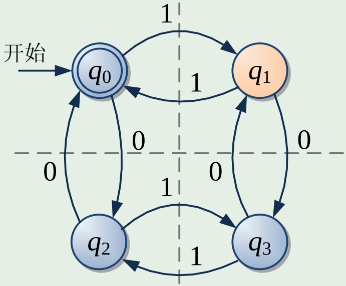
接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1

例

例3.1中FA



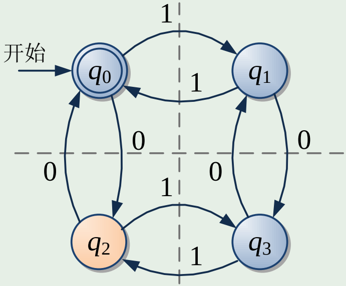
接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1
- q_1：已读过偶数个 0，奇数个 1

例

例3.1中FA



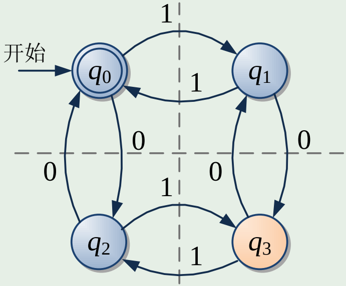
接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1
- q_1：已读过偶数个 0，奇数个 1
- q_2：已读过奇数个 0，偶数个 1

例

例3.1中FA



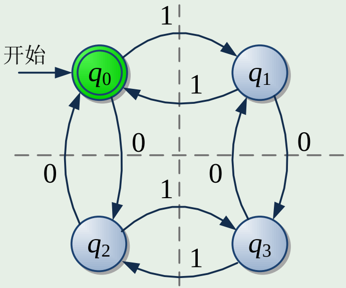
接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1
- q_1：已读过偶数个 0，奇数个 1
- q_2：已读过奇数个 0，偶数个 1
- q_3：已读过奇数个 0，奇数个 1

例

例3.1中FA



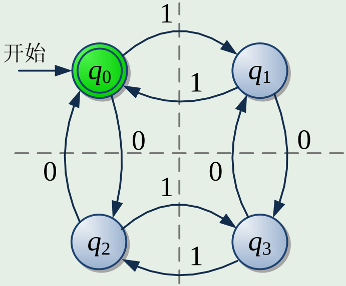
接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1
- q_1：已读过偶数个 0，奇数个 1
- q_2：已读过奇数个 0，偶数个 1
- q_3：已读过奇数个 0，奇数个 1

例

例3.1中FA



接受什么样的语言？

分析：4个状态起什么作用？

- q_0：已读过偶数个 0，偶数个 1
- q_1：已读过偶数个 0，奇数个 1
- q_2：已读过奇数个 0，偶数个 1
- q_3：已读过奇数个 0，奇数个 1

$L(M) = \{$一切含有偶数个 0 和偶数个 1 的字符串$\}$

FA $\Longleftrightarrow$ 语言

1. 给出 FA，指明它所接受的语言
   - FA $\Longrightarrow$ 语言
2. 给出语言，构造接受它的 FA
   - 语言 $\Longrightarrow$ FA

FA $\Longleftrightarrow$ 语言

1. 给出 FA ，指明它所接受的语言
   - FA $\Longrightarrow$ 语言
2. 给出语言，构造接受它的 FA
   - 语言$\Longrightarrow$ FA

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串010}\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串010}\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且} x \text{中至少包含一个子串010}\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且} x \text{中不能出现子串010}\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*,$且$x$中至少包含一个子串$010\}$

$L_2 = \{x \mid x \in \{0,1\}^*,$且$x$中不能出现子串$010\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到1：

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到1：暂时和要辨认的串无关，

  可仍保留在原状态，

  继续读下一个符号；

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到0：

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到0：要引起注意了，

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到0：要引起注意了，

  可能是子串010的开头，

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到0：要引起注意了，

  可能是子串010的开头，如何应对？

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1=\{x\mid x\in\{0,1\}^*,\text{且}x\text{中至少包含一个子串}010\}$$

$$L_2=\{x\mid x\in\{0,1\}^*,\text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

从左向右扫描输入串，从M_1的初始状态出发

- 遇到0：要引起注意了，

  可能是子串010的开头，如何应对？

  必须改变一个状态以应对这种情况，这个状态记为“0”**状态**。

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串010}\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串010}\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“0”状态：

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“0”状态：

  若读过1，进一步引起注意，

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“0”状态：

  若读过1，进一步引起注意，

  连续读过0、1的情况，更接近于子串010，

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, 且x中至少包含一个子串010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, 且x中不能出现子串010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“0”状态：

  若读过1，进一步引起注意，

  连续读过0、1的情况，更接近于子串010，如何应对？

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“0”状态：

  若读过1，进一步引起注意，

  连续读过0、1的情况，更接近于子串010，如何应对？

  用**状态“01”**表示。

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“01”状态：

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“01”状态：

  再遇到0，已经出现子串010，怎么样？

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“01”状态：

  再遇到0，已经出现子串010，怎么样？

  该输入串被接受，

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“01”状态：

  再遇到0，已经出现子串010，怎么样？

  该输入串被接受，如何应对？

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 在“01”状态：

  再遇到0，已经出现子串010，怎么样？

  该输入串被接受，如何应对？

  用**状态“010”**表示接受状态。

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010}

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010}

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 此后，

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 此后，

  再遇到任何符号（0或1），怎么样？

举例：由语言构造FA

例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

分析L_1：

- 此后，

  再遇到任何符号（0或1），怎么样？

  都仍然进入该接受状态。

举例：由语言构造FA

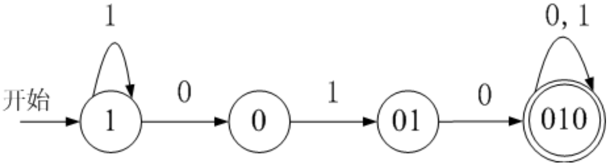
例 例3.2

给出两个集合：

$L_1 = \{x \mid x \in \{0,1\}^*$,且$x$中至少包含一个子串010$\}$

$L_2 = \{x \mid x \in \{0,1\}^*$,且$x$中不能出现子串010$\}$

要求构造两个FA M_1和M_2，分别接受L_1和L_2



FA M_1 接受L_1

举例：由语言构造FA

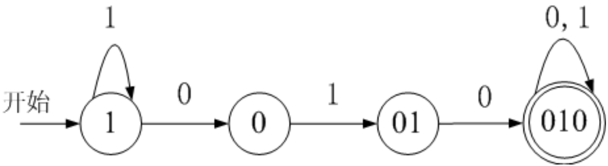
例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

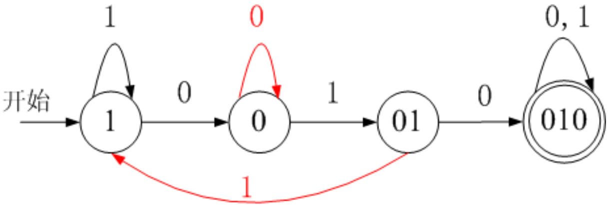
$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2



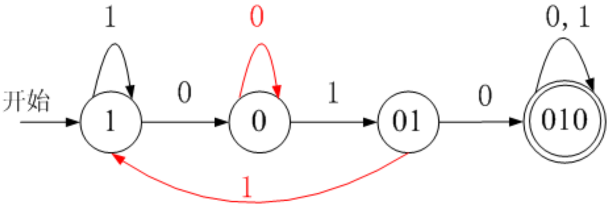
FA M_1 接受L_1 完整吗？还差什么？

举例：由语言构造FA



- “0”状态遇0，保持在“0”状态；
- “01”状态遇1，“半途而废，从头再来”，返回“1”状态。

举例：由语言构造FA



- “0”状态遇0，保持在“0”状态；
- “01”状态遇1，“半途而废，从头再来”，返回“1”状态。

若输入串中含有子串010，则一定能到达接受状态；

若输入串中不含子串010，则一定不能到达接受状态。

$$\therefore L(M_1) = L_1$$

举例：由语言构造FA

例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2

举例：由语言构造FA

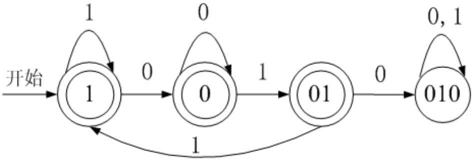
例 例3.2

给出两个集合：

$$L_1 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中至少包含一个子串}010\}$$

$$L_2 = \{x \mid x \in \{0,1\}^*, \text{且}x\text{中不能出现子串}010\}$$

要求构造两个FA M_1和M_2，分别接受L_1和L_2



FA M_2 接受L_2

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$$L = \{x \mid x \in \{0,1\}^+\text{，且把}x\text{看成二进制数时，}x\text{模5余0}\}$$

（即x为二进制数，能被5整除）

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$$L=\{x \mid x\in\{0,1\}^+\text{，且把}x\text{看成二进制数时，}x\text{模5余0}\}$$

（即x为二进制数，能被5整除）

提示：

当二进制数x的位数向右不断增加时，其值的增加有规律：

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$L = \{x \mid x \in \{0,1\}^+$，且把$x$看成二进制数时，$x$模5余0$\}$

（即x为二进制数，能被5整除）

提示：

当二进制数x的位数向右不断增加时，其值的增加有规律：

- 二进制$x0$，十进制$2x$；
- 二进制$x1$，十进制$2x+1$。

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$$L = \{x \mid x \in \{0,1\}^+\text{，且把}x\text{看成二进制数时，}x\text{模5余0}\}$$

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$$L = \{x \mid x \in \{0,1\}^+，\text{且把}x\text{看成二进制数时，}x\text{模5余0}\}$$

$x0$ $x1$

- x模5余0，$2x$模5余0，$2x+1$模5余1；
- x模5余1，$2x$模5余2，$2x+1$模5余3；
- x模5余2，$2x$模5余4，$2x+1$模5余0；
- x模5余3，$2x$模5余1，$2x+1$模5余2；
- x模5余4，$2x$模5余3，$2x+1$模5余4；

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

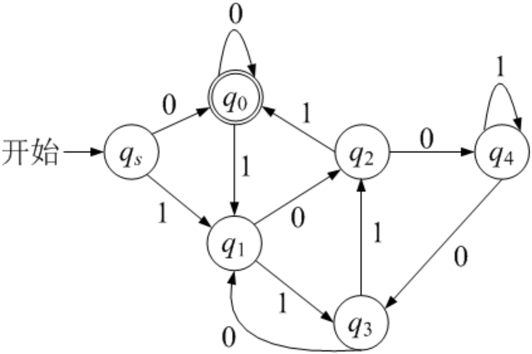
$$L = \{x \mid x \in \{0,1\}^+，\text{且把}x\text{看成二进制数时，}x\text{模5余0}\}$$

举例：由语言构造FA

例 例3.3

构造一个FA M，它接受的语言为：

$$L = \{x \mid x \in \{0,1\}^+，\text{且把}x\text{看成二进制数时，}x\text{模5余0}\}$$



总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数： $\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数： $\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数：$\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数： $\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数：$\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：$M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数：$\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：$M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数：$\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M=(Q,\Sigma,\delta,q_0,F)$
   - 转移图
   - 扩充转移函数：$\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M)=\{x\mid\delta(q_0,x)\in F\}$
   - FA $\Longleftrightarrow$ 语言

总结

1. 有穷自动机的定义
   - 五元组：
     $M = (Q, \Sigma, \delta, q_0, F)$
   - 转移图
   - 扩充转移函数： $\hat{\delta}$
2. 有穷自动机接受的语言
   - 接受状态集 F 的作用
   - $L(M) = \{x \mid \delta(q_0, x) \in F\}$
   - FA $\Longleftrightarrow$ 语言