



词法分析---有限状态自动机

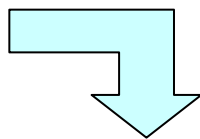
编译原理

华保健

bjhua@ustc.edu.cn

回顾：自动生成

声明式的规范



词法分析器

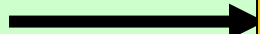


词法分析器的实现方法

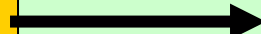
- 至少两种方法：
 - 手工实现算法
 - 自动生成法
- 我们继续讨论第二种方法
 - 首先要用到的第二个数学工具是有限状态自动机（FA）

有限状态自动机 (FA)

输入的字符串



FA



{Yes, No}

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

字母表

状态
集

初始
状态

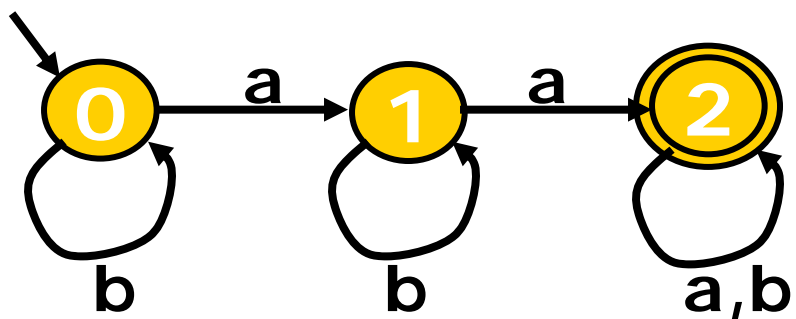
终结状
态集

转移函数

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

自动机例子

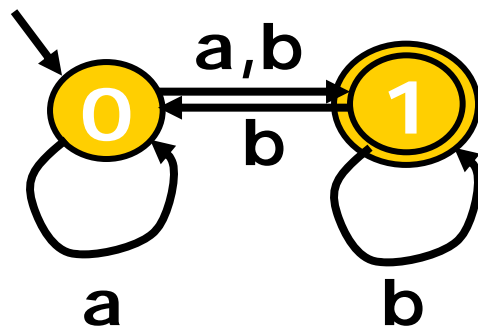
- 什么样的串可被接受?



- 转移函数:

- $\{ (q_0, a) \rightarrow q_1, (q_0, b) \rightarrow q_0, (q_1, a) \rightarrow q_2, (q_1, b) \rightarrow q_1, (q_2, a) \rightarrow q_2, (q_2, b) \rightarrow q_2 \}$

自动机第二个例子



■ 转移函数:

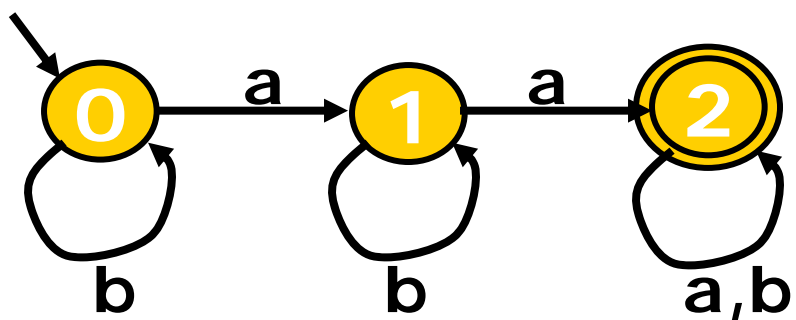
- $\{(q0, \mathbf{a}) \rightarrow \{q0, q1\},$
 $(q0, \mathbf{b}) \rightarrow \{q1\},$
 $(q1, \mathbf{b}) \rightarrow \{q0, q1\}\}$



有限状态自动机小结

- 确定状态有限自动机DFA
 - 对任意的字符，最多有一个状态可以转移
 - $\delta: S \times \Sigma \rightarrow S$
- 非确定的有限状态自动机NFA
 - 对任意的字符，有多于一个状态可以转移
 - $\delta: S \times (\Sigma \cup \epsilon) \rightarrow \wp(S)$

DFA的实现



状态\字符	a	b
0	1	0
1	2	1
2	2	2