编译原理与技术

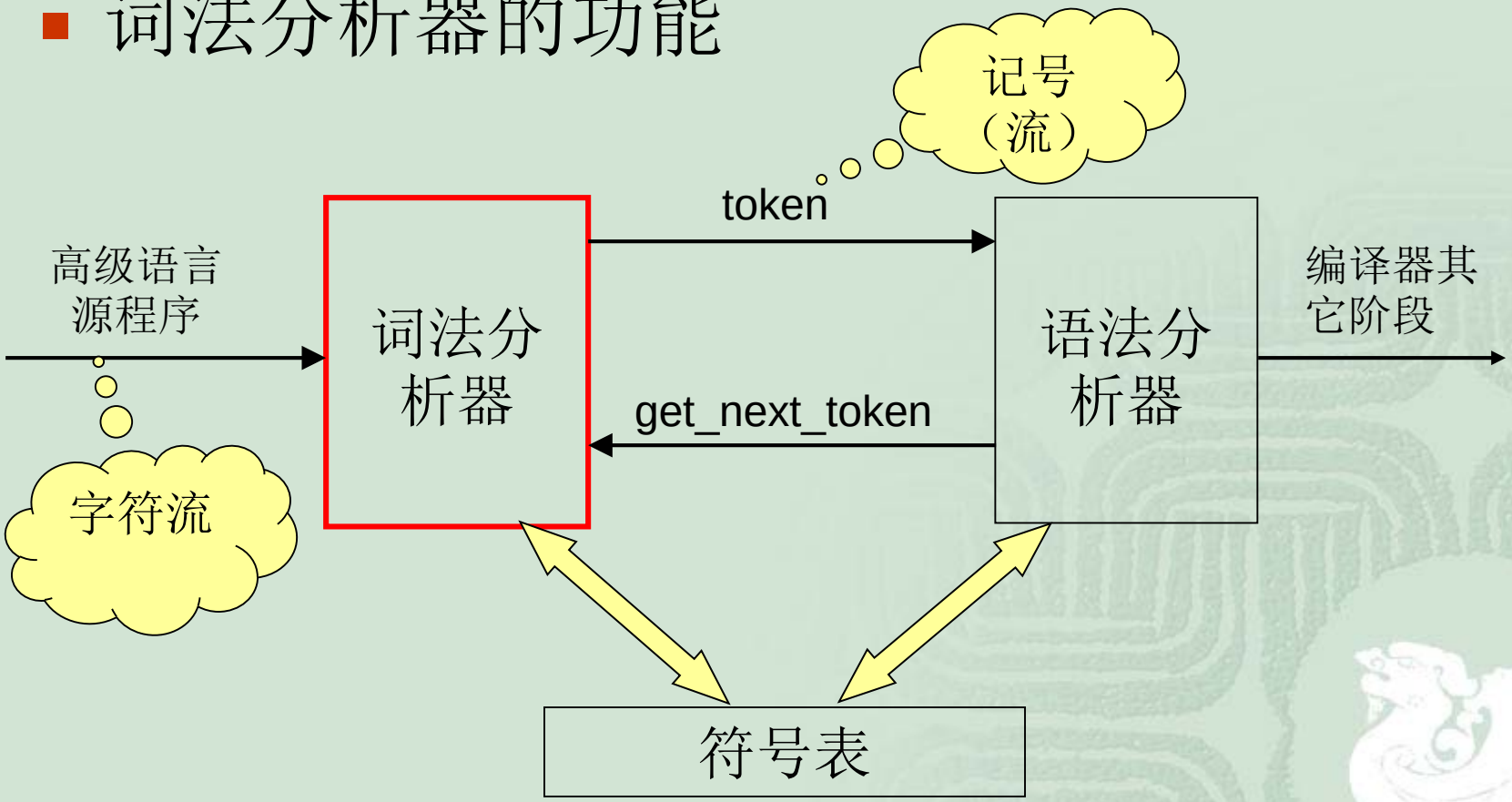
词法分析(1)

词法分析

- 词法分析器介绍
- 正规式与正规集
- 有限自动机
- 词法分析器的自动生成—Lex

词法分析器介绍

- 词法分析器的功能



词法分析器介绍

- 词法分析器的功能
  - ✓读源程序，产生记号序列
  - ✓剥去源程序中的注释（块、行）和“空白”符
  - ✓预处理—宏处理与文件包含

词法分析器介绍

- 词法分析器作为独立子程序
  - ✓简化设计
  - ✓提高编译效率
  - ✓增强可移植性

词法分析器介绍

- 记号、模式与单词
  - ✓记号—同类单词的总称
  - ✓模式—描述构成记号的字符串的规则
  - ✓单词—源程序中能匹配任一记号的字符串

程序语言的记号（1）

记号		单词	模式
关键字	WHILE	while	while
	FOR	for	for
标识符	ID	temp, i, max	字母开头的字母数字串
常数	NUM	3.14 100	数字字符串{.数字字符串}

程序语言的记号（2）

记号		单词	模式
运算符	MUL	*	*
	GT	>	>
界符	,	,	,
串常量	STRING	“hello” ‘there’	双（单）引号中间的字符串（不包括引号本身）

词法分析器介绍

- 词法分析器的二元输出

<记号，记号的属性>

单词（字符串）的类别

匹配记号的单词多于一个时，须提供额外的信息以区别之

词法分析器介绍

- 词法分析器的二元输出

<记号，记号的属性>

记号影响语法分析的决策

属性（如类型、偏移）则关系到记号的翻译

词法分析器介绍

- e.g.1 pascal源程序片段：

```
begin
    length := length + 1;
    if length<20 then read(nextch);
end;
```

e.g.1 pascal源程序片段的字符流(SP表示空格)							
b	e	g	i	n	\n	\t	l
e	n	g	t	h	SP	:	=
SP	l	e	n	g	t	h	SP
+	SP	1	;	\n	\t	i	f
SP	l	e	n	g	t	h	<
2	0	SP	t	h	e	n	SP
r	e	a	d	(	n	e	x
t	c	h	)	;	\n	e	n
d	;						

e.g. 1 词法分析器的输出记号流(1)

<BEGIN,->

<ID,指向符号表length条目的指针>

<ASSIGN,->

<ID,指向符号表length条目的指针> //不是多余的！！

<+, - >

<NUM, 1>　　// 属性是常量“值”本身

<; , - >

<IF, - >

e.g. 1 词法分析器的输出记号流(2)

<ID,指向符号表length条目的指针>

<LT, - >

<NUM, 20 >

<THEN, - >

<READ, - >

<(, - >

<ID,指向符号表nextch条目的指针>

<), - >

<END, - >

< ; , - >

词法分析器介绍

- 超前搜索
  - FORTRAN中的关键字“不保留”

    1） DO100K=1,10

    2） DO100K=1.10

    3） IF(5.EQ.M) I=10

    4） IF(5)=55
  - 有关算符的识别

    C/C++, java的++, --, >=, !=, == 等，与之对应

    \+ , - >, !, =

词法分析器介绍

- 词法错误
  - 可检测非法字符的出现
  - if VS fi
- 词法分析器的设计
  - 手工编写－采用汇编语言或高级语言
  - 自动生成－Lex

词法分析器介绍

- 状态转换图

  用于记号的识别。状态之间用带有标记（字符）的有向边连接；每读入一个字符会引起状态变化，直至单词（记号）被识别出来。

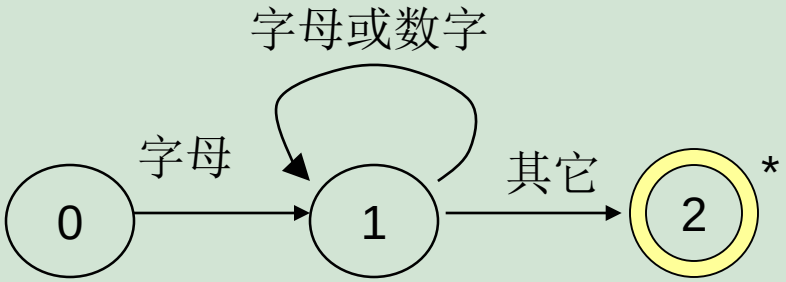
  开始状态：状态转换图的初始状态（尚未读字符）

  接受状态：某个单词被识别时所处的状态（终态）

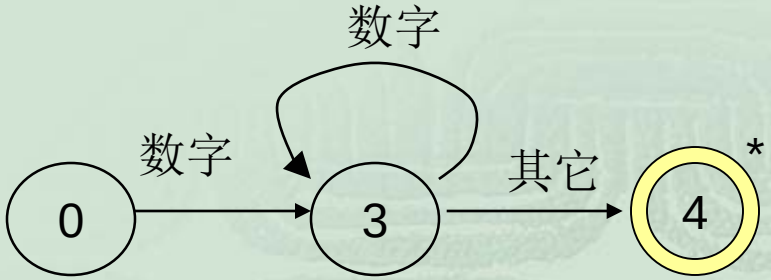
  单词（记号）的识别过程即是从开始状态出发到某接受状态的变化过程。

词法分析器介绍

- 状态转换图



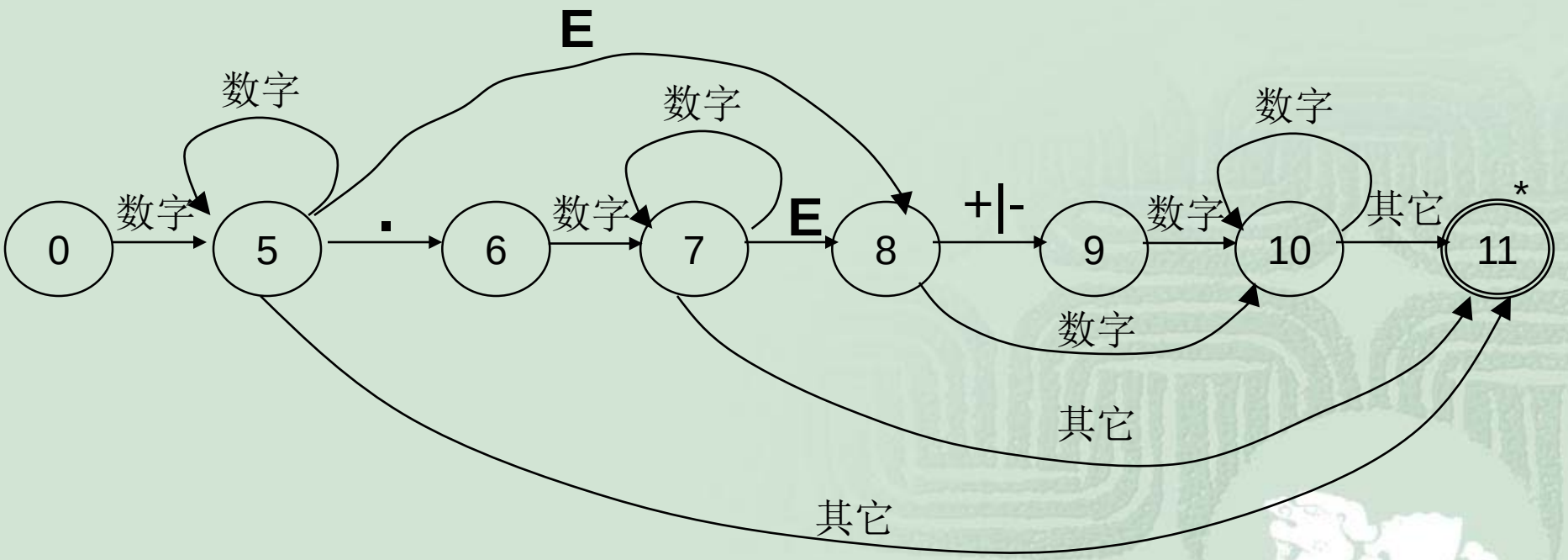
识别标识符的转换图



识别整数的转换图

词法分析器介绍

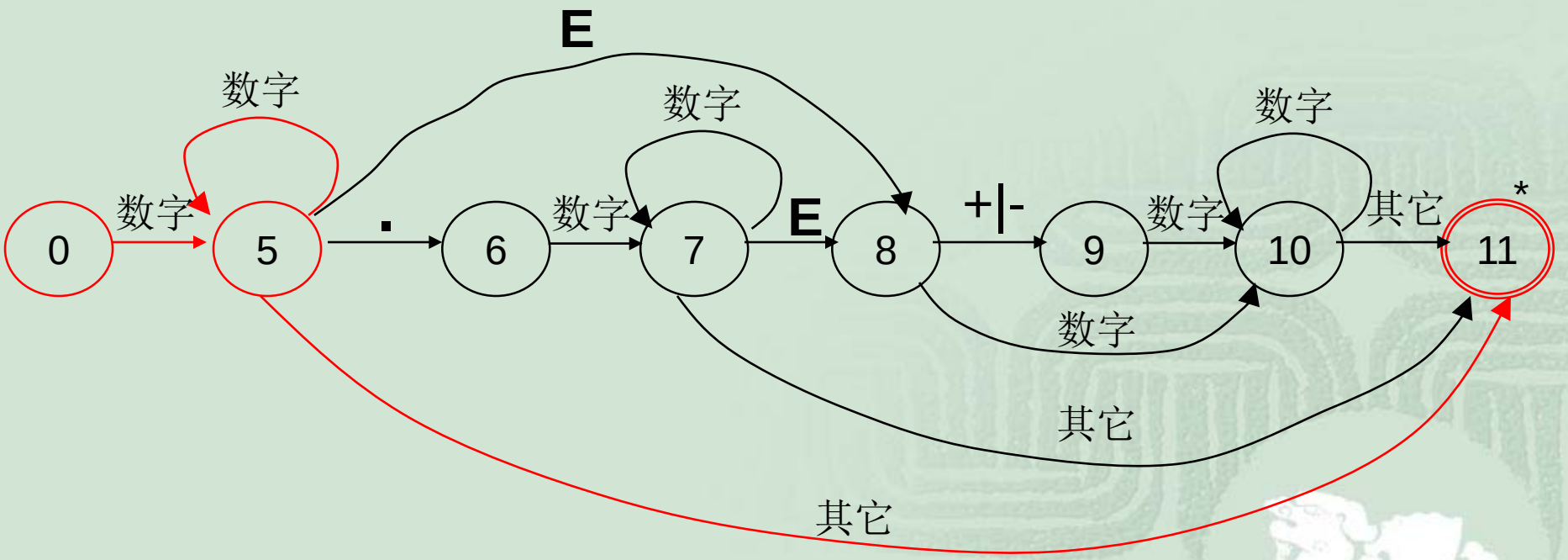
- 状态转换图



识别Pascal无符号数的转换图

词法分析器介绍

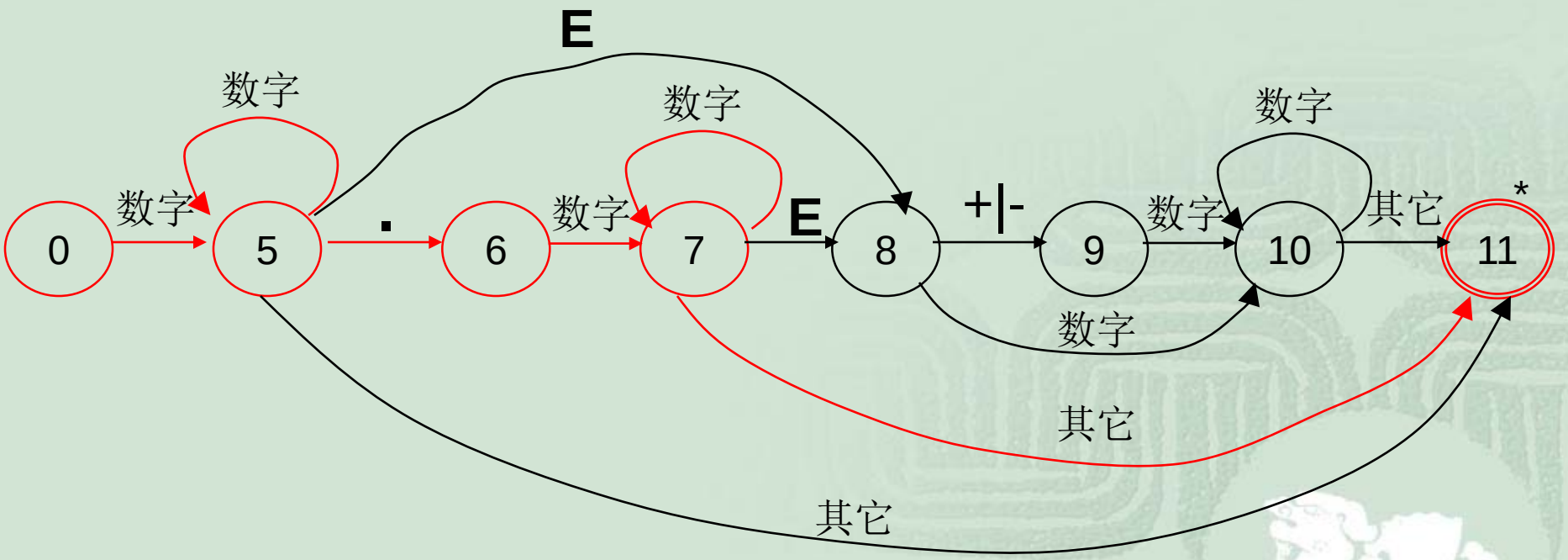
- 状态转换图



（红线）识别Pascal无符号整数的转换图

词法分析器介绍

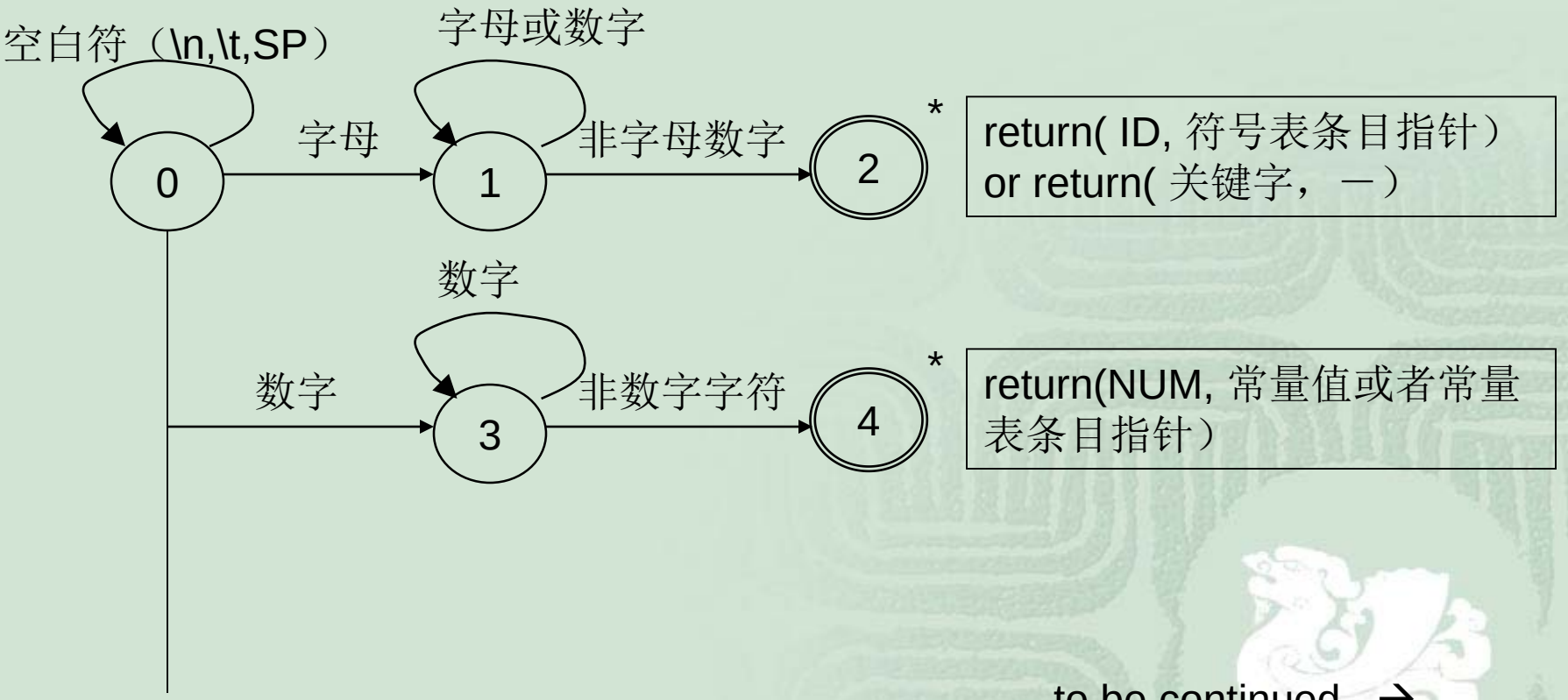
- 状态转换图



识别Pascal无符号小数的转换图

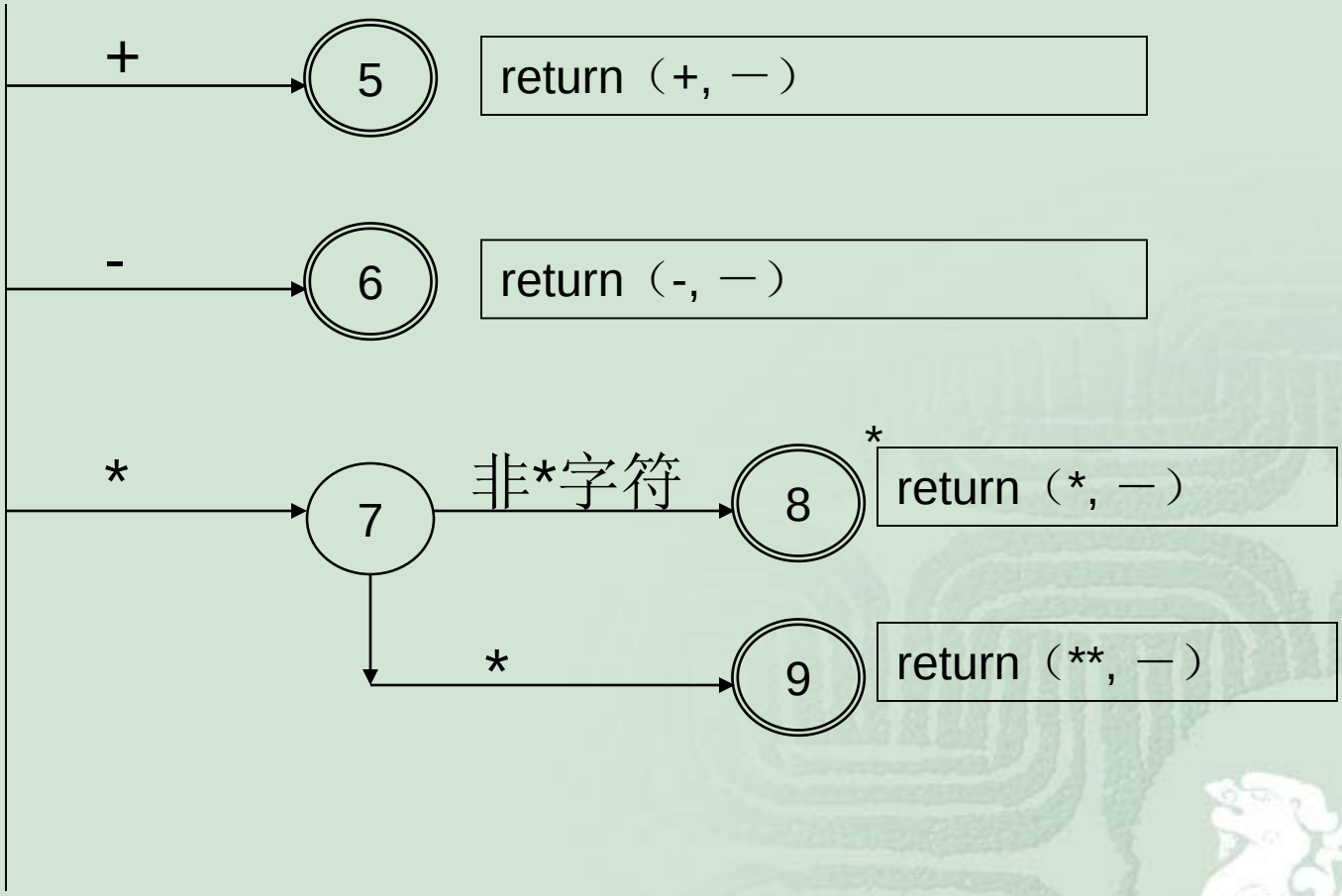
- 状态转换图的实现

e.g. 2 简单词法分析的转换图（识别关键字、标识符、无符号整数、算符和界符）



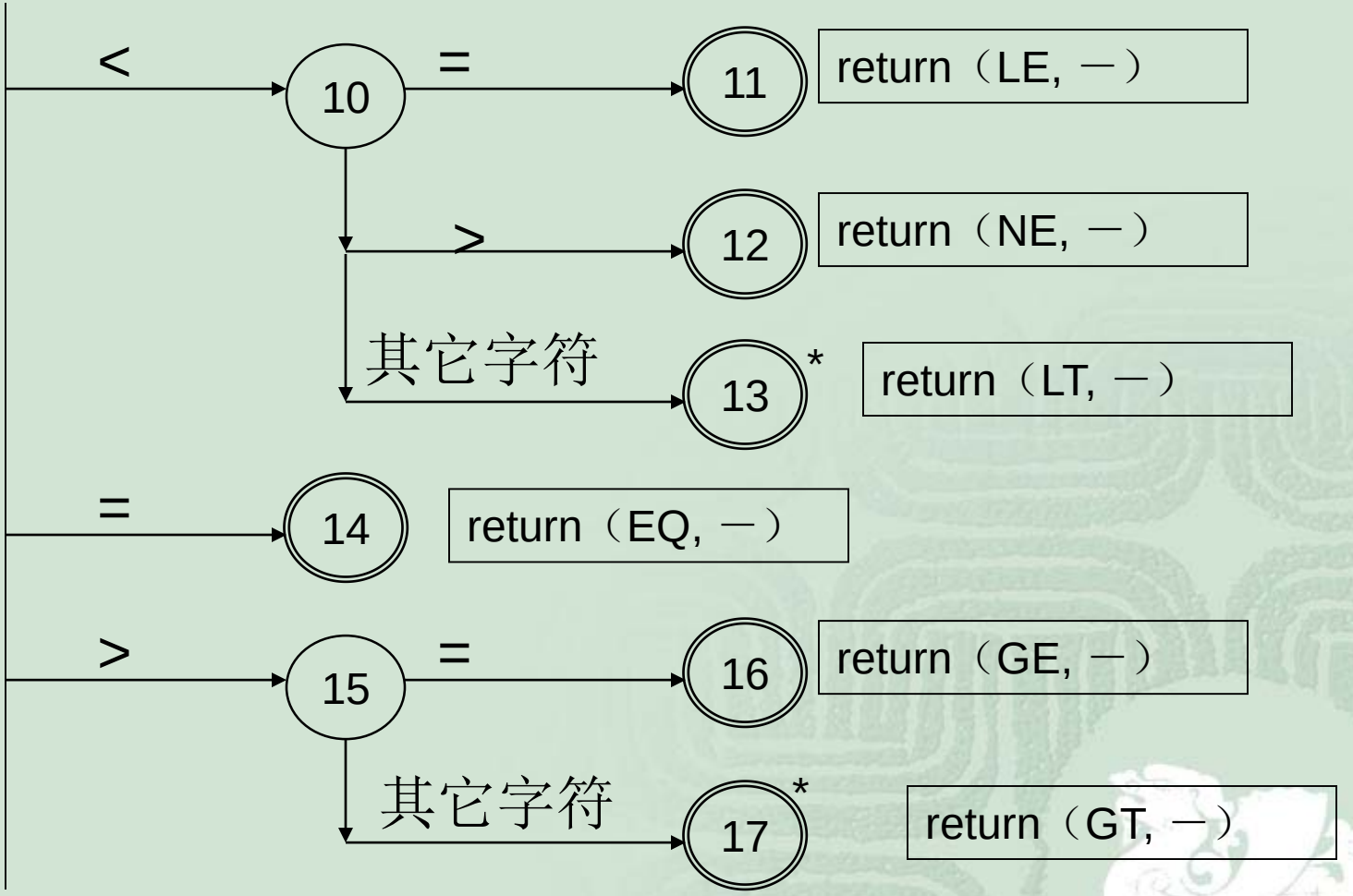
to be continued →

- e.g. 2简单词法分析的转换图



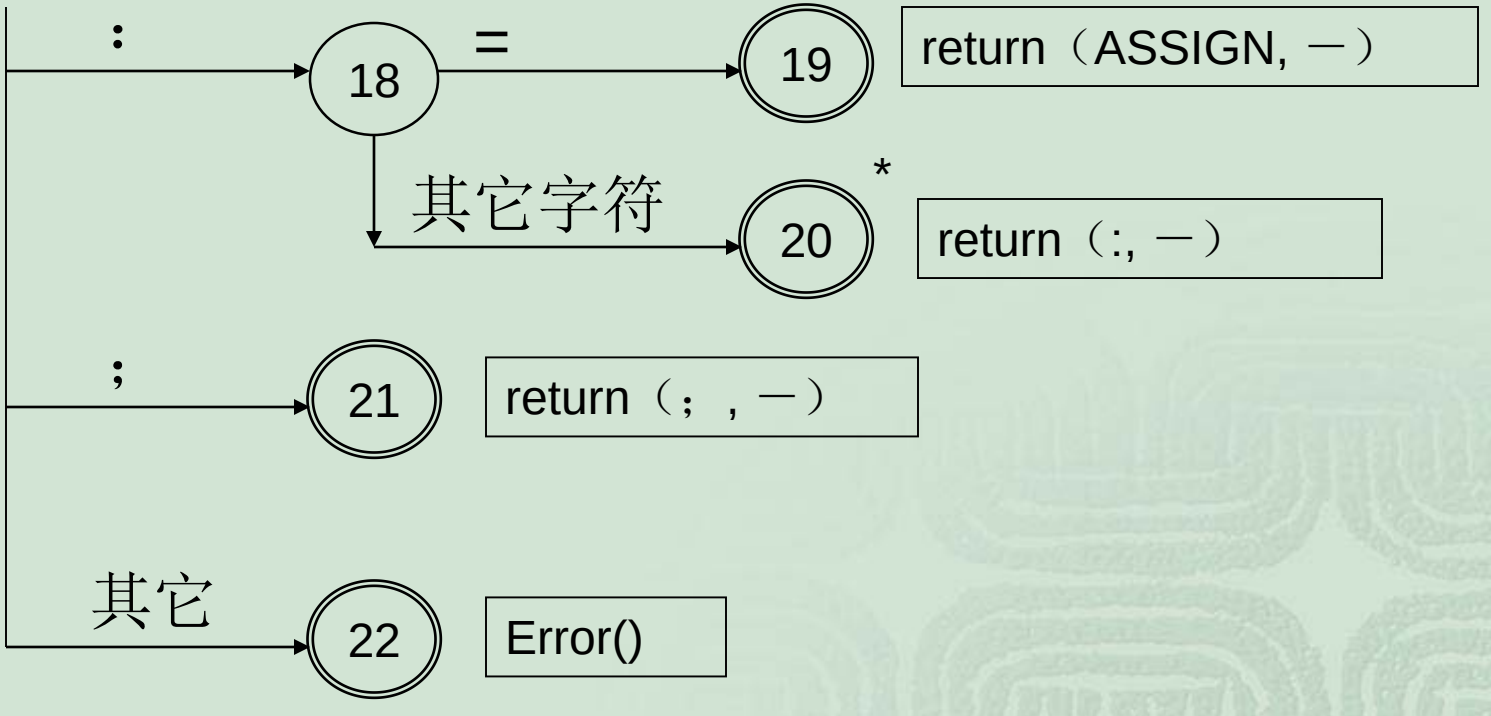
to be continued →

- e.g. 2简单词法分析的转换图



to be continued →

- e.g. 2简单词法分析的转换图



状态转换程序

串与语言

- 语言
  - 语言L＝{ s | s 是∑上任一字符串}，
    s称为语言L的一个句子。
  - 字母表∑－符号/字符的非空有限集合
    e.g. 二进制数的∑＝{0,1}，而十进制数的∑＝{0,1,…,9}
    ∑*－表示∑上所有字符串的集合；L⊆∑*
  - 字符串－ ∑上若干字符组成的有穷序列

串与语言

- 语言
  - 字符串

    e.g. $\Sigma=\{0,1\}$上的0，1串（二进制数）如0111，10101；$\Sigma=\{a,b\}$上的 a，b，aa，abab，…

    ✓空串－ε，$\varepsilon\in\Sigma^*$，

    ✓串长 - $|s|$ = {s中所含字符的个数}，$|\varepsilon|=0$

    ✓串的连接运算 - 任意串x，y，一般地，$xy\neq yx$，$x\varepsilon=\varepsilon x$

    ✓串的前缀 - 任意串x，从其第一个字符（最左字符）起的字符序列是其前缀。ε亦是。

    e.g. x = abc，则ε，a，ab，abc均是x的前缀

语言的运算

语言的运算	
描述 / 运算	语言L和语言M
连接（积）	$LM=\{ xy \mid x\in L \text{ 且 } y\in M \}$
合并（和）	$L\cup M=\{x \mid x\in L \text{ 或 } x\in M \}$
闭包	$L^*=L^0\cup L^1\cup L^2\cup\ldots=\bigcup_{i=0}^{\infty} L^i$
正闭包	$L^+=L^1\cup L^2\cup L^3\cup\ldots=\bigcup_{i=1}^{\infty} L^i$

- 语言

e.g. L={a,b,…,z}, D={0,1,…,9}, B={ _ }

LUD = {…}

LD={…}

L*={…}

L(LUD)*={…} (LU B)(LUDUB)*={…}

D^{+}={…}

正规式与正规集

- 正规式－用于描述记号的构成规则
- 正规集－正规式描述的语言（匹配正规式的串集）

正规式	正规集
ε	{ε}
∅	∅
a∈∑	{a}

正规式与正规集

如果R和S是∑上的正规式，分别对应∑上的正规集L(R)和L(S)，则

正规式	正规集
R | S	L(R) ∪ L(S)
R · S	L(R) · L(S)
R*	(L(R))*
(R)	L(R)

正规式与正规集

∑上的正规式，其运算有 | 、 · 和 *

运算符		优先级	结合性
或	|	低	左结合
连接	·	高	左结合
闭包	*	最高	左结合

正规式与正规集

∑上的正规式，满足如下代数定律——

代数定律	描述
交换律	R | S = S | R
结合律	R | (S|T) = (R|S) | T R (ST) = (RS) T
分配律	R (S|T) = (RS) | (RT) (R|S) T = (RT) | (ST)
同一律	ε R = R ε = R

正规式与正规集

∑上的正规式，也具有如下代数定律——

$(R^*)^* = R^*$

$(R \mid \varepsilon)^* = R^*$

$R^+ = R\ R^*$

正规式与正规集

- e.g.3 设∑={a, b}, 则

正规式	正规集
a(a|b)*	∑上以a开头的串集
ba*	∑上以b开头后跟任意个a的串集
(a|b)*a(a|b)(a|b)	∑上倒数第三个字符是a的串集

正规式与正规集

- e.g.3 设∑={a, b}，R = a(a|b)*，事实上有

L(R) = L(a(a|b)*)
= L(a) L((a|b)*)
= L(a) (L(a|b))*
= L(a) (L(a) ∪L(b))*
= {a} ({a} ∪ {b})*
= {a} ({ a, b })*
= {a} { ε, a, b, aa, ab, ba, bb, abb, … }
= {a,aa, ab, aaa, aab, aba, abb, aabb, …}

即L(R)是 ∑上以a开头的串集。

正规式与正规集

- 正规定义

$$d_1 \rightarrow r_1$$
$$d_2 \rightarrow r_2$$
$$\ldots$$
$$d_n \rightarrow r_n$$

各个d_i的名字不同；每个r_i是$\Sigma \cup \{d_1, d_2, \ldots d_{i-1}\}$上的正规式

正规式与正规集

- e.g.4 Pascal 标识符

letter → A | B | … | Z | a | b | … | z

英文字母集合

digit → 0 | 1 | … | 9

十进制数字集合

id → letter (letter | digit)*

标识符的正规定义

正规式与正规集

- e.g.5 Pascal 无符号数

  digit → 0 | 1 | … | 9

  digits → digit digit*

  fraction → '.' digits | ε

  exponent → (E (+ | - | ε)) digits | ε

  num → digits fraction exponent

数字串集合

小数部分（可空）

指数部分（可空）

正规式与正规集

- e.g.6 email 地址： qlzheng@ustc.edu.cn

  name → letter letter*

  field → ('.' name) *

  email → name '@' name field