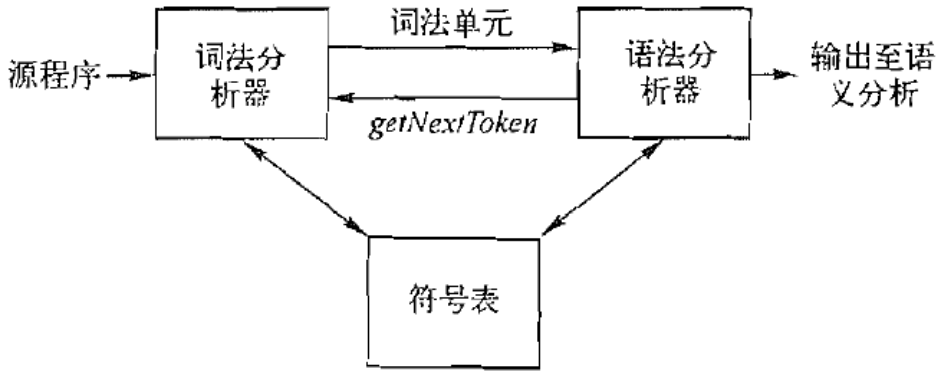
词法分析

魏恒峰

hfwei@nju.edu.cn

2020 年 11 月 3 日

输入: 程序文本/字符串 s & **词法单元 (token) 的规约**



输出: 词法单元流

token : ⟨token-class, attribute-value⟩

词法单元	非正式描述	词素示例
if	字符 `i`, `f`	`if`
else	字符 `e`, `l`, `s`, `e`	`else`
comparison	`<` 或 `>` 或 `<=` 或 `>=` 或 `==` 或 `!=`	`<=`, `!=`
id	字母开头的字母 / 数字串	`pi`, `score`, `D2`
number	任何数字常量	`3.14159`, `0`, `6.02e23`
literal	在两个 `"`之间，除`"` 以外的任何字符	`"core dumped"`

token : ⟨token-class, attribute-value⟩

词法单元	非正式描述	词素示例
if	字符 i, f	if
else	字符 e, l, s, e	else
comparison	< 或 > 或 <= 或 >= 或 == 或 !=	<=, !=
id	字母开头的字母 / 数字串	pi, score, D2
number	任何数字常量	3.14159, 0, 6.02e23
literal	在两个 "之间，除" 以外的任何字符	"core dumped"

int/if 关键词

ws 空格、制表符、换行符

comment “//” 开头的一行注释或者 “/* */” 包围的多行注释

```c
int main(void)
{
    printf("hello, world\n");
}
```

```c
int main(void)
{
    printf("hello, world\n");
}
```

int ws **main/id** **LP** **void** **RP** ws

LB ws

ws **id** **LP** **literal** **RP** **SC**ws

RB

```c
int main(void)
{
    printf("hello, world\n");
}
```

int ws **main/id** **LP** **void** **RP** ws

LB ws

ws **id** **LP** **literal** **RP** **SC**ws

RB

本质上, 这就是一个**字符串（匹配/识别）算法**

词法分析器的三种设计方法

手写词法分析器

词法分析器的生成器

自动化词法分析器

生产环境下的编译器 (如 gcc) 通常选择**手写词法分析器**

识别字符串 s 中符合**某种词法单元模式**的**所有词素**

```
if ab42>=42
    xyz =3.14
else xyz = 2.718
```

ws **if** **else** **id** **integer** **real** **relop**

识别字符串 s 中符合**某种词法单元模式**的**所有词素**

```
if ab42>=42
    xyz =3.14
else xyz = 2.718
```

ws　**if**　**else**　**id**　**integer**　**real**　**relop**

识别字符串 s 中符合**某种词法单元模式**的**第一个词素**

识别字符串 s 中符合**某种词法单元模式**的**所有词素**

```
if ab42>=42
    xyz =3.14
else xyz = 2.718
```

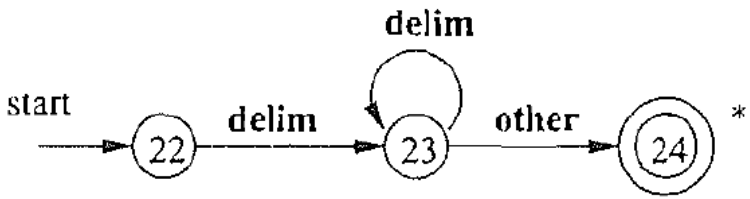
ws　**if**　**else**　**id**　**integer**　**real**　**relop**

识别字符串 s 中符合**某种词法单元模式**的**第一个词素**

识别字符串 s 中符合**特定词法单元模式**的**第一个词素**

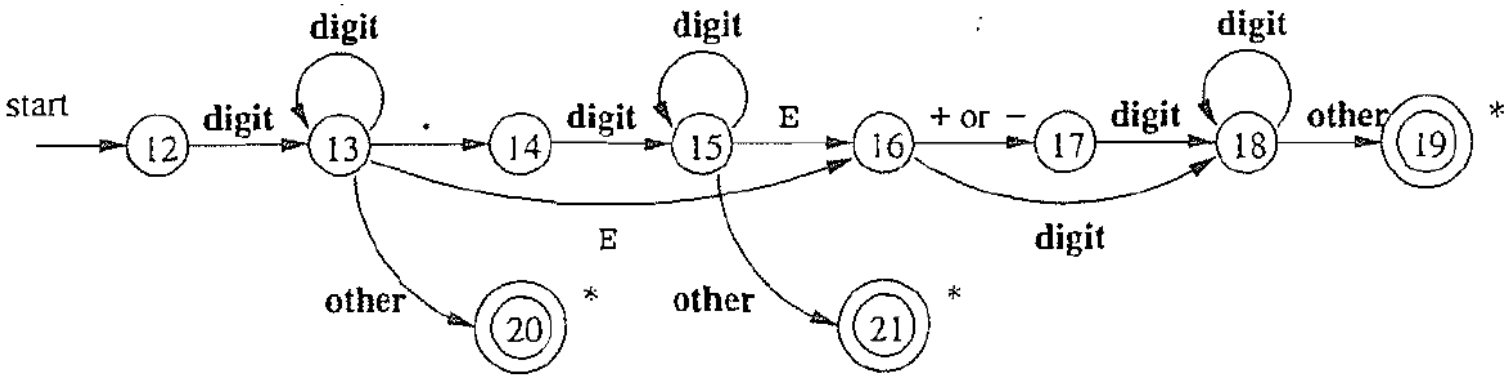
识别字符串 s 中符合**特定词法单元模式**的**第一个词素**

ws: **blank tab newline**



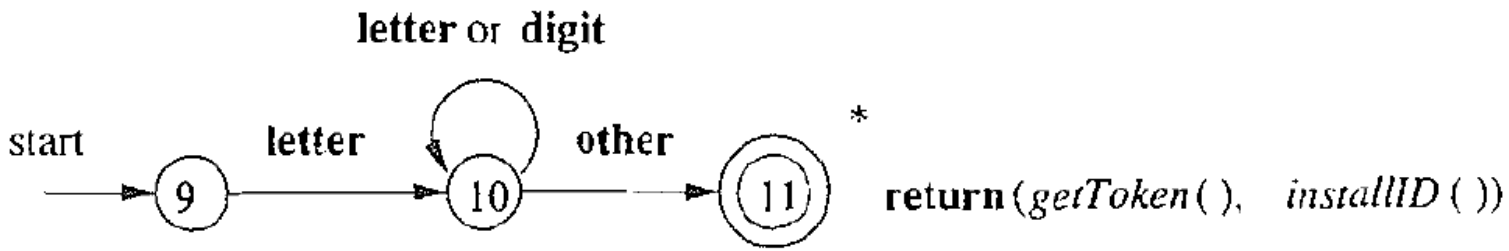
识别字符串 s 中符合**特定词法单元模式**的**第一个词素**

num: 整数 (允许以 0 开头)



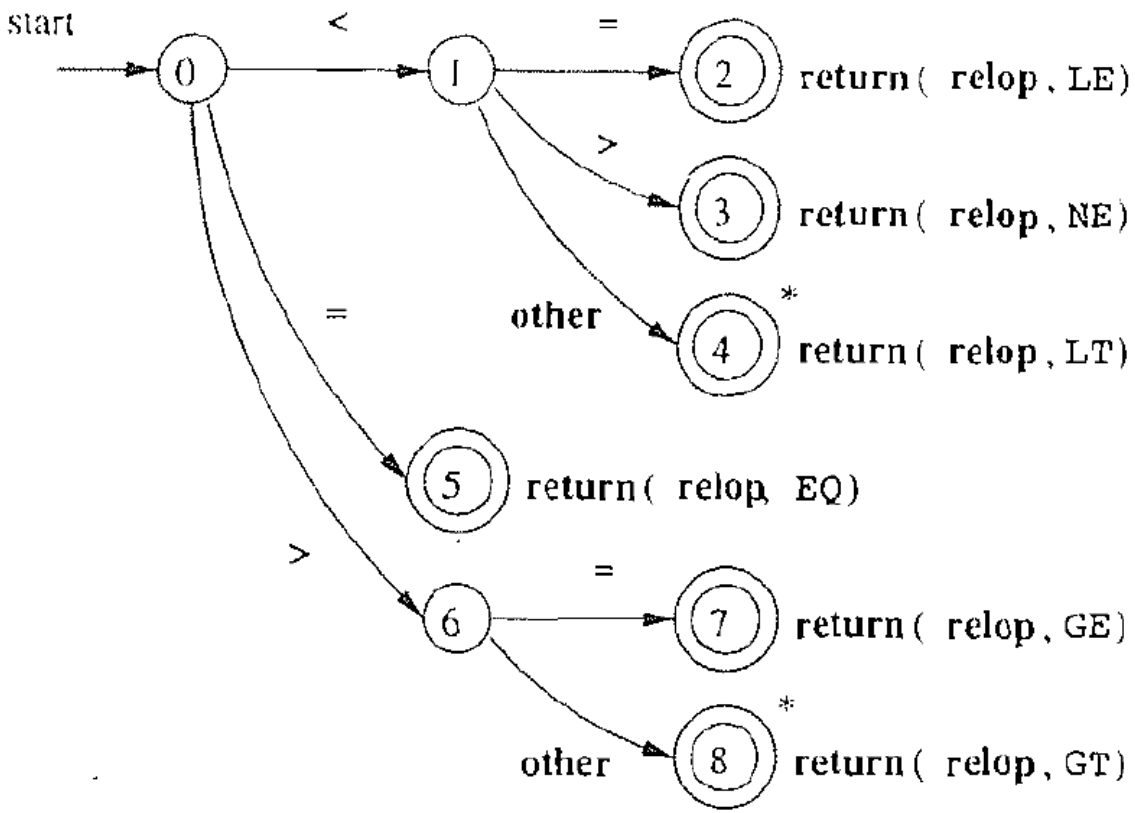
识别字符串 s 中符合**特定词法单元模式**的**第一个词素**

id: 字母开头的字母/数字串

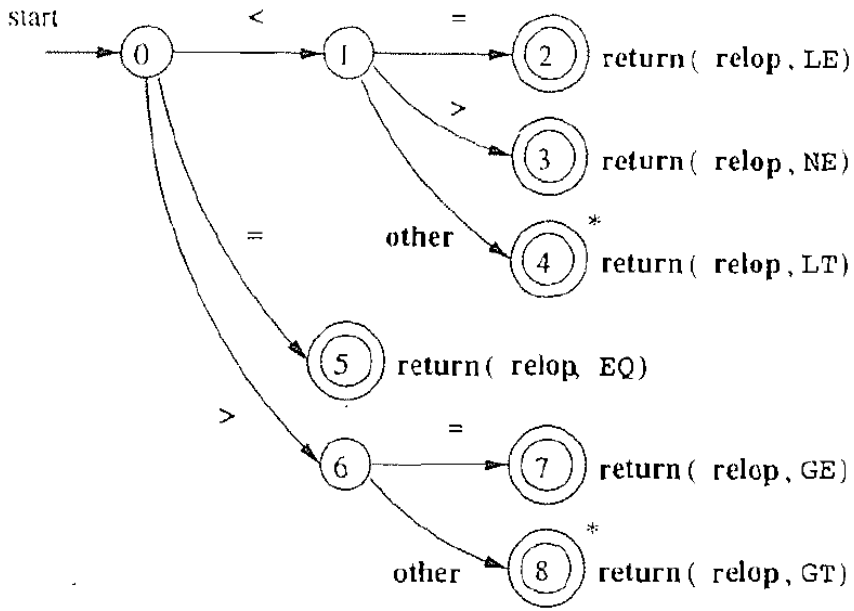


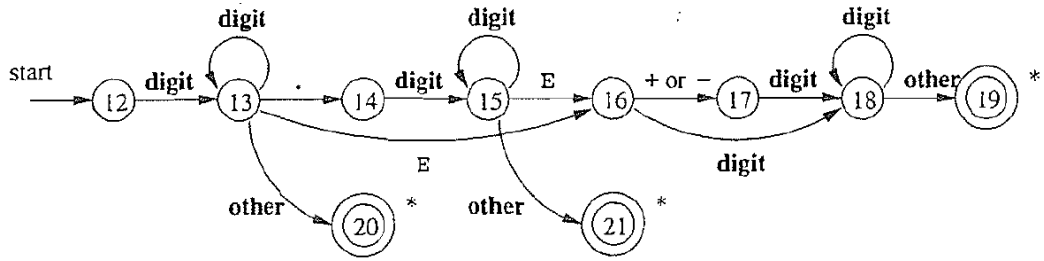
识别字符串 s 中符合**特定词法单元模式**的**第一个词素**

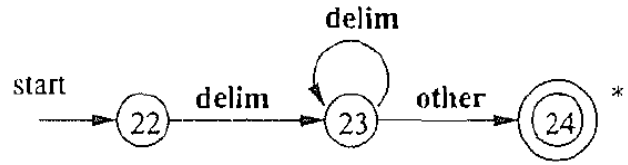
relop: $<$ $>$ $<=$ $>=$ $==$ $<>$

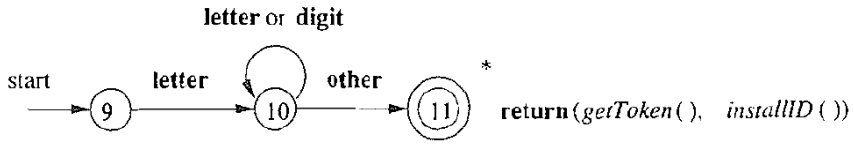


识别字符串 s 中符合**某种词法单元模式**的**第一个词素** (SCAN())

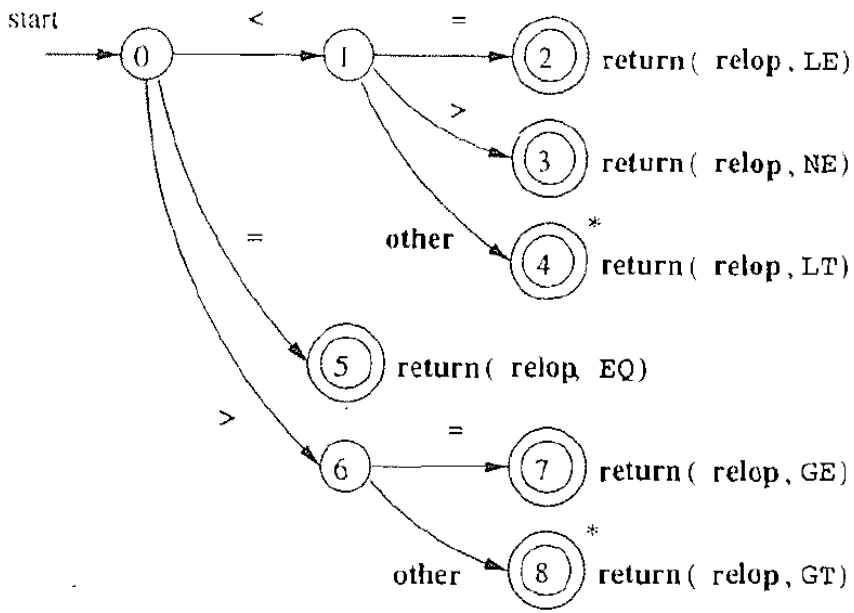


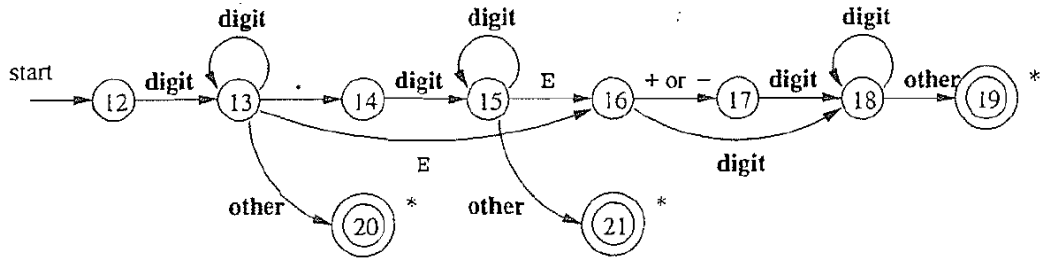


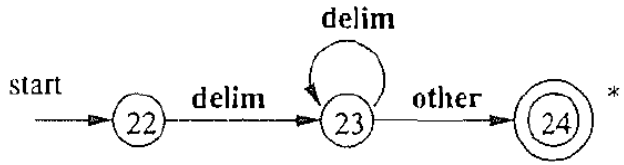


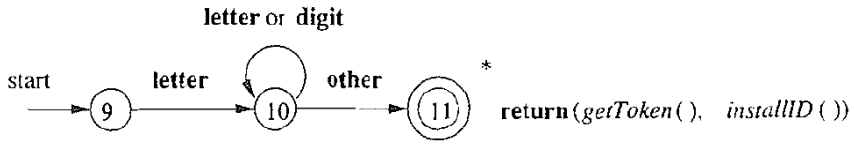


识别字符串 s 中符合**某种词法单元模式**的**第一个词素** (SCAN())



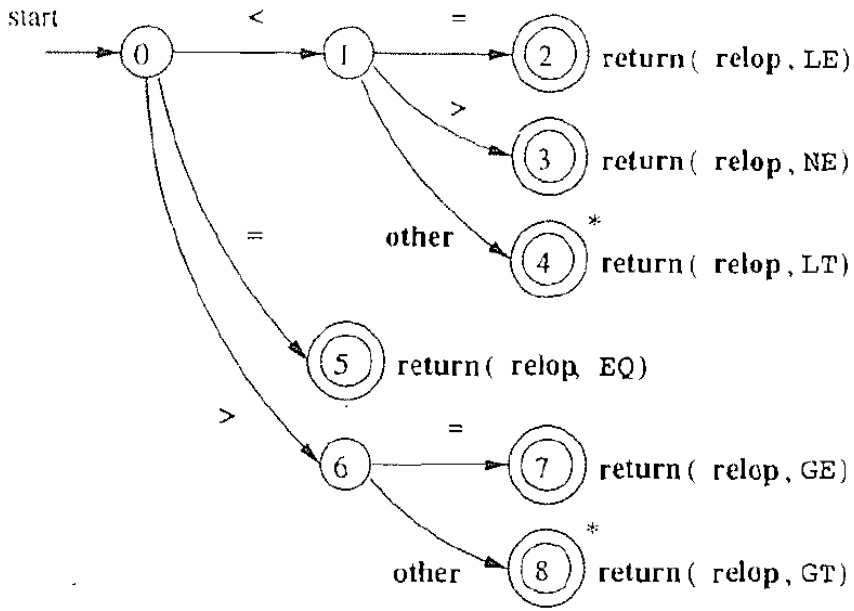


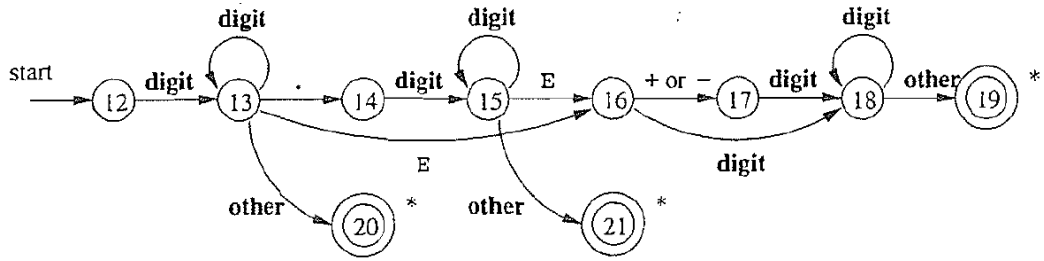


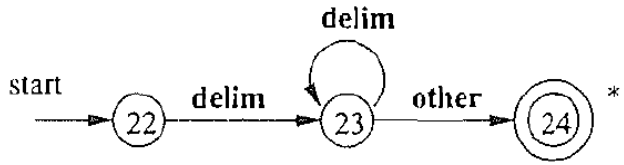


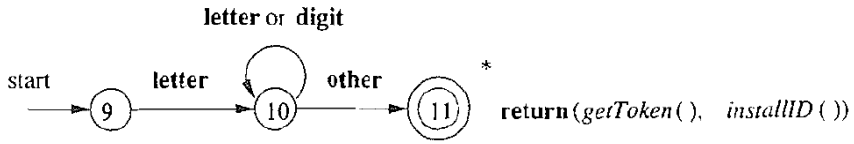
根据**下一个字符**即可判定词法单元的类型

识别字符串 s 中符合**某种词法单元模式**的**第一个词素** (SCAN())









根据**下一个字符**即可判定词法单元的类型

否则，报告**该字符有误**，并忽略该字符

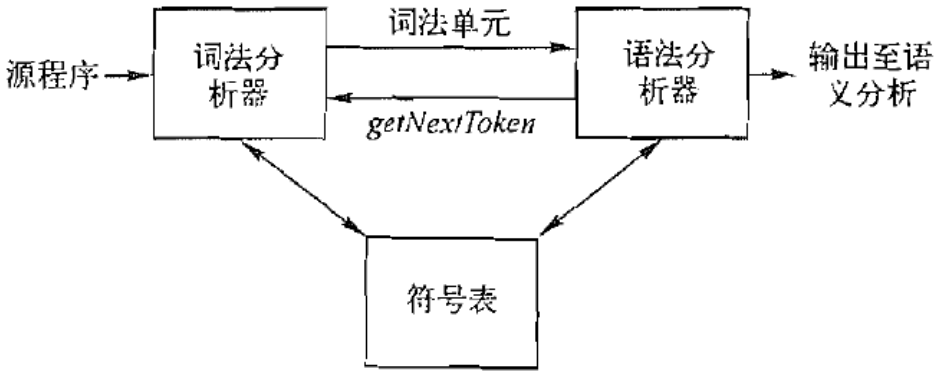
识别字符串 s 中符合**某种词法单元模式**的**所有词素**

最外层循环调用 SCAN()

或者, 由语法分析器按需调用 SCAN()

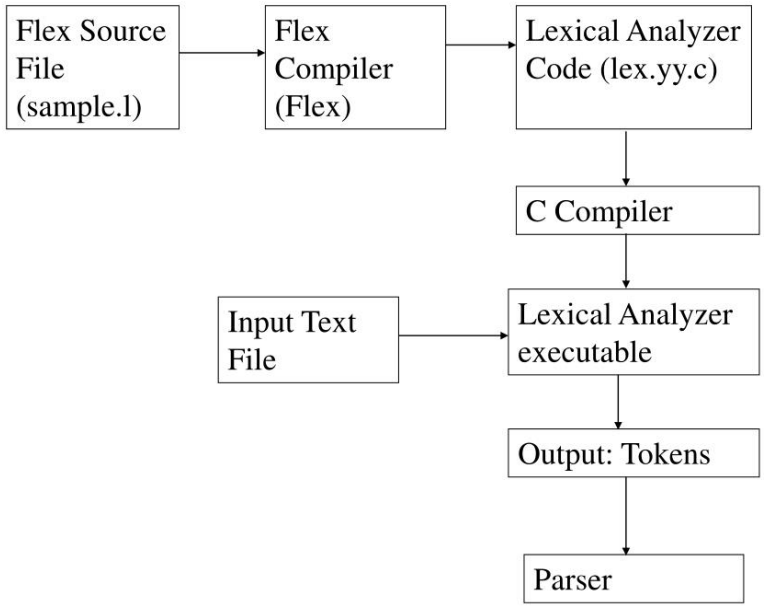

flex

输入: 程序文本/字符串 s & 词法单元的规约



输出: 词法单元流

输入：词法单元的规约

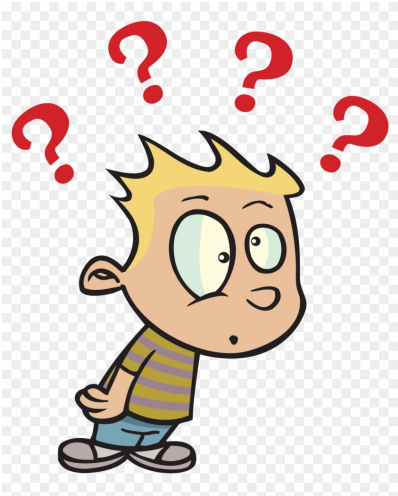


输出：词法分析器

词法单元的规约

词法单元	非正式描述	词素示例
if	字符 i, f	if
else	字符 e, l, s, e	else
comparison	< 或 > 或 <= 或 >= 或 == 或 !=	<=, !=
id	字母开头的字母 / 数字串	pi, score, D2
number	任何数字常量	3.14159, 0, 6.02e23
literal	在两个 "之间，除" 以外的任何字符	"core dumped"

词法单元的规约



词法单元	非正式描述	词素示例
if	字符 i, f	if
else	字符 e, l, s, e	else
comparison	< 或 > 或 <= 或 >= 或 == 或 !=	<=, !=
id	字母开头的字母 / 数字串	pi, score, D2
number	任何数字常量	3.14159, 0, 6.02e23
literal	在两个 "之间，除" 以外的任何字符	"core dumped"

我们需要词法单元的**形式化**规约

id: 字母开头的字母/数字串

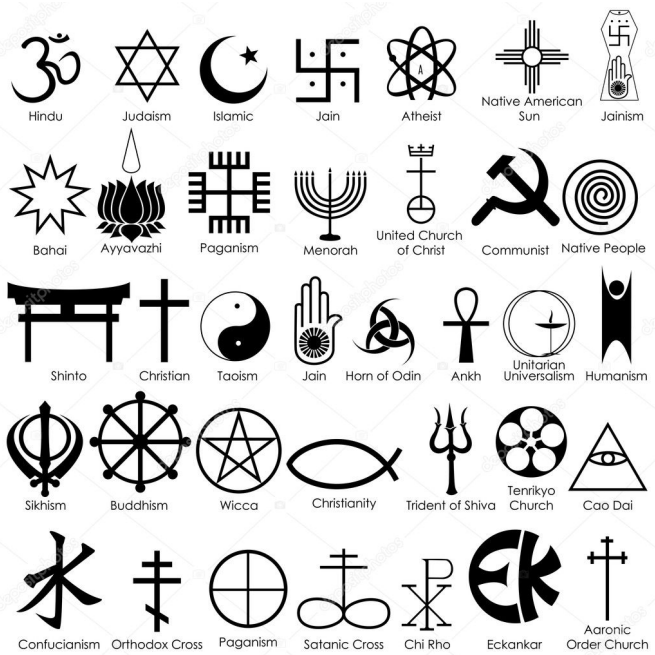
id 定义了一个集合, 我们称之为**语言 (Language)**

它使用了字母与数字等符号集合, 我们称之为**字母表 (Alphabet)**

该语言中的每个元素 (即, 标识符) 称为**串 (String)**

Definition (字母表)

字母表 Σ 是一个有限的符号集合。



Definition (串)

字母表 Σ 上的串 (s) 是由 Σ 中符号构成的一个**有穷**序列。

空串：$|\epsilon| = 0$

Definition (串上的 “连接” 运算)

$$x = dog, y = house \qquad xy = doghouse$$

$$s\epsilon = \epsilon s = s$$

Definition (串上的 “连接” 运算)

$$x = dog, y = house \qquad xy = doghouse$$

$$s\epsilon = \epsilon s = s$$

Definition (串上的 “指数” 运算)

$$s^0 \triangleq \epsilon$$

$$s^i \triangleq ss^{i-1}, i > 0$$

Definition (语言)

语言是给定字母表 Σ 上一个任意的**可数的**串集合。

$$\emptyset \qquad \{\epsilon\}$$

Definition (语言)

语言是给定字母表 Σ 上一个任意的**可数的**串集合。

$$\emptyset \qquad \{\epsilon\}$$

$$\mathbf{id} : \{a, b, c, a1, a2, \dots\}$$

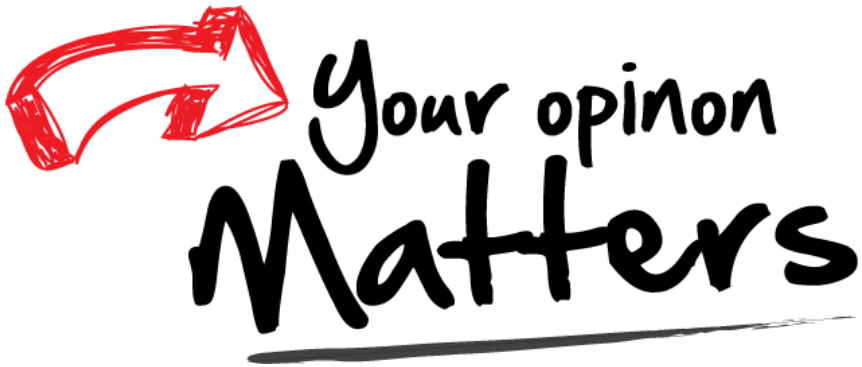
$$\mathbf{ws} : \{\mathbf{blank}, \mathbf{tab}, \mathbf{newline}\}$$

$$\mathbf{if} : \{if\}$$

语言是串的集合

因此, 我们可以通过集合操作构造新的语言。


Thank
You!



Office 926

hfwei@nju.edu.cn