编译原理第四章第三次作业

李昊宸 2017K8009929044

4.5.2 对于文法 S -> S S + | S S \* | a和下面的最右句型，指出其归约时使用到的句柄

1）S S S + a \* +

2）S S + a \* a +

3）a a a \* a + +

答：

最右句型，归约时从最左侧进行归约

1）最左侧的右部是S S + ，所以句柄为S S +，产生式为S -> S S +

2）最左侧的右部是S S + ，所以句柄为S S +，产生式为S -> S S +

3）全为终结符，故最后一步用到的产生式必然为S -> a，故句柄为a

4.6.2 对于文法 S->S S + | S S \* | a，增广该文法，构造SLR项目集。计算GOTO函数，给出文法的语法分析表。该文法是SLR文法吗？

答：

该文法的增广文法：

S‘ -> S

S -> S S +

S -> S S \*

S -> a

构建SLR项目集：

I0：

S’ -> . S

S -> . S S +

S -> . S S \*

S -> . a

I1:

S’ -> S .

S -> S . S +

S -> S . S \*

S -> . S S +

S -> . S S \*

S -> . a

I2:

S -> a .

I3:

S -> S S . +

S -> S S . \*

S -> S . S +

S -> S . S \*

S -> . S S +

S -> . S S \*

S -> . a

I4:

S -> S S + .

I5:

S -> S S \* .

计算GOTO函数：

GOTO (I0，S) = I1

GOTO(I0，a) = I2

GOTO(I1，S) = I3

GOTO(I1，a) = I2

GOTO(I1，$) = acc

GOTO(I3，S) = I3

GOTO(I3，a) = I2

GOTO(I3，\*) = I5

GOTO(I3，+) = I4

构建语法分析表：

FOLLOW(S’) = { a , + , \* , $ }

FOLLOW(S) = { a , + , \* , $ }

对文法标号：

1）S->S S +

2）S->S S \*

3）S-> a

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 状态 | ACTION | | | | GOTO |
| a | + | \* | $ | S |
| 0 | s2 |  |  |  | 1 |
| 1 | s2 |  |  | acc | 3 |
| 2 | r3 | r3 | r3 | r3 |  |
| 3 | s2 | s4 | s5 |  | 3 |
| 4 | r1 | r1 | r1 | r1 |  |
| 5 | r2 | r2 | r2 | r2 |  |

该文法是SLR文法，因为在语法信息表里没有矛盾的项目

4.7.1 对于文法 S -> S S + | S S \* | a 构造

1）规范 LR项目集族，并构建其语法分析表

2）LALR 项集族

答：

1）

该文法的增广文法：

S‘ -> S

S -> S S +

S -> S S \*

S -> a

构建LR项目集族：

搜索符的产生：A - > α.Bβ

I0：

S’ -> . S，$ 该产生式闭包生成下面三个，B为S，β为ε，a为$，搜索符为FIRST($)

S -> . S S +，$/a 由上个产生式闭包生成后，又可以循环生成以下两个，B为S，β为S+，a为$，搜索符为FIRST(S+$) = FIRST(S) = a

S -> . S S \*，$/a

S -> . a，$/a

I1: 0 -> S

S’ -> S .，$/a

S -> S . S +，$/a 闭包产生倒数三个产生式，B为S，β为+，a为$/a，搜索符为FIRST(+)

S -> S . S \*，$/a 类似的，搜索符为FIRST(\*)

S -> . S S +，a/+/\* 搜索符a是像生成I0时类似的循环生成出来的

S -> . S S \*，a/+/\*

S -> . a

I2: 0 -> a

S -> a .，$/a

I3: 1 -> S

S -> S S . +，$/a

S -> S S . \*，$/a

S -> S . S +，a/+/\*

S -> S . S \*，a/+/\*

S -> . S S +，a/+/\*

S -> . S S \*，a/+/\*

S -> . a，a/+/\*

I4: 1-> a

S -> a . ，a/+/\*

I5: 3 -> \*

S -> S S \* .，$/a

I6: 3 -> +

S -> S S \* .，$/a

I7: 3 -> S

S -> S S . +，a/+/\*

S -> S S . \*，a/+/\*

S -> S . S +，a/+/\*

S -> S . S \*，a/+/\*

S -> a .，a/+/\*

S -> . S S +，a/+/\*

S -> . S S \*，a/+/\*

S -> . a，a/+/\*

I8: 7 -> \*

S -> S S \* .，a/+/\*

I9: 7 -> +

S -> S S \* .，a/+/\*

对文法标号：

1）S->S S +

2）S->S S \*

3）S-> a

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 状态 | ACTION | | | | GOTO |
| a | + | \* | $ | S |
| 0 | s2 |  |  |  | 1 |
| 1 | s4 |  |  | acc | 3 |
| 2 | r3 |  |  | r3 |  |
| 3 | s2 | s4 | s5 |  | 3 |
| 4 | r3 | r3 | r3 |  |  |
| 5 | r1 |  |  | r1 |  |
| 6 | r2 |  |  | r2 |  |
| 7 | s4 | s8 | s9 |  | 7 |
| 8 | r1 | r1 | r1 |  |  |
| 9 | r2 | r2 | r2 |  |  |

2）

LALR集族：

将具有相同核心的LR(1)项目合并。

I2与I4可以合并

I3与I7可以合并

I5与I8可以合并

I6与I9可以合并

I0：

S’ -> . S，$

S -> . S S +，$/a

S -> . S S \*，$/a

S -> . a，$/a

I1: 0 -> S

S’ -> S .，$/a

S -> S . S +，$/a

S -> S . S \*，$/a

S -> . S S +，a/+/\*

S -> . S S \*，a/+/\*

S -> . a

I24: 0/1 -> a

S -> a .，$/a

I37: 1/37 -> S

S -> S S . +，$/a/+/\*

S -> S S . \*，$/a/+/\*

S -> S . S +，a/+/\*

S -> S . S \*，a/+/\*

S -> . S S +，a/+/\*

S -> . S S \*，a/+/\*

S -> . a，a/+/\*

I58: 37 -> \*

S -> S S \* .，$/a/+/\*

I69: 37 -> +

S -> S S \* .，$/a/+/\*

合并后没有动作冲突，状态个数也与SLR的状态个数相同