

Homework 4

PB17000297 罗晏宸

September 26 2019

1 Exercise 3.9

为习题 3.3 的文法构造预测分析器（递归下降分析程序）。

$$S \rightarrow S \textbf{ and } S \mid S \textbf{ or } S \mid \textbf{ not } S \mid \textbf{ true } \mid \textbf{ false } \mid (S)$$

解 该文法是二义的，等价的非二义文法为

$$E \rightarrow E \textbf{ or } T \mid T$$

$$T \rightarrow T \textbf{ and } F \mid F$$

$$F \rightarrow \textbf{ not } F \mid (E) \mid \textbf{ true } \mid \textbf{ false }$$

消除其中的左递归，得到

$$E \rightarrow TE'$$

$$E' \rightarrow \textbf{ or } TE' \mid \varepsilon$$

$$T \rightarrow FT'$$

$$T' \rightarrow \textbf{ and } FT' \mid \varepsilon$$

$$F \rightarrow \textbf{ not } F \mid (E) \mid \textbf{ true } \mid \textbf{ false }$$

列出文法中各非终结符的开始符号和后继符号集合

$$\begin{aligned}
 FIRST(E) &= \{ \text{not}, (, \text{true}, \text{false} \} & FOLLOW(E) &= \{), \$ \} \\
 FIRST(E') &= \{ \text{or}, \varepsilon \} & FOLLOW(E') &= \{), \$ \} \\
 FIRST(T) &= \{ \text{not}, (, \text{true}, \text{false} \} & FOLLOW(T) &= \{ \text{or},), \$ \} \\
 FIRST(T') &= \{ \text{and}, \varepsilon \} & FOLLOW(T') &= \{ \text{or},), \$ \} \\
 FIRST(F) &= \{ \text{not}, (, \text{true}, \text{false} \} & FOLLOW(F) &= \{ \text{and}, \text{or},), \$ \}
 \end{aligned}$$

对于产生式 $E' \rightarrow \text{or } TE' \mid \varepsilon$

$$\begin{aligned}
 &FIRST(\text{or } TE') \cap FIRST(\varepsilon) \\
 &= \{ \text{or} \} \cap \{ \varepsilon \} \\
 &= \emptyset \\
 &FIRST(\text{or } TE') \cap FOLLOW(E') \\
 &= \{ \text{or} \} \cap \{), \$ \} \\
 &= \emptyset
 \end{aligned}$$

对于产生式 $T' \rightarrow \text{and } FT' \mid \varepsilon$

$$\begin{aligned}
 &FIRST(\text{and } FT') \cap FIRST(\varepsilon) \\
 &= \{ \text{and} \} \cap \{ \varepsilon \} \\
 &= \emptyset \\
 &FIRST(\text{and } FT') \cap FOLLOW(T') \\
 &= \{ \text{and} \} \cap \{ \text{or},), \$ \} \\
 &= \emptyset
 \end{aligned}$$

对于产生式 $F \rightarrow \text{not } F \mid (E) \mid \text{true} \mid \text{false}$

$$\begin{aligned}
 &FIRST(\text{not } F) \cap FIRST((E)) \cap FIRST(\text{true}) \cap FIRST(\text{false}) \\
 &= \{ \text{not} \} \cap \{ (\} \cap \text{true} \cap \text{false} \\
 &= \emptyset
 \end{aligned}$$

因此该文法是 LL(1) 文法。为其构造递归下降预测分析器如下

```
1  void match (terminal t)
```

```

2      {
3          if (lookahead == t)
4              lookahead = nextToken();
5          else
6              error();
7          return;
8      }
9
10     void E()
11     {
12         if ((lookahead == not) || (lookahead == '(')
13             || (lookahead == true) || (lookahead ==
14                 false))
15         {
16             T();
17             E'();
18         }
19         else
20             error();
21         return;
22     }
23
24     void E'()
25     {
26         if (lookahead == or)
27         {
28             match(or);
29             T();
30             E'();
31         }
32         else if ((lookahead == ')') || (lookahead
33             == '$'))
34             return;

```

```

32         else
33             error();
34         return;
35     }
36
37     void T()
38     {
39         if ((lookahead == not) || (lookahead == '(')
40             || (lookahead == true) || (lookahead ==
41                 false))
42         {
43             F();
44             T'();
45         }
46         else
47             error();
48         return;
49     }
50
51     void T'()
52     {
53         if (lookahead == and)
54         {
55             match(and);
56             F();
57             T'();
58         }
59         else if (lookahead == or)
60         {
61             match(or);
62             T();
63             E'();
64         }

```

```

63         else if ((lookahead == ')') || (lookahead
64             == '$'))
65             return;
66         else
67             error();
68         return;
69     }
70     void F()
71     {
72         if (lookahead == not)
73         {
74             match(not);
75             F();
76         }
77         else if (lookahead == '(')
78         {
79             match('(');
80             E();
81             match(')');
82         }
83         else if (lookahead == true)
84             match(true);
85         else if (lookahead == false)
86             match(false);
87         else
88             error();
89         return;
90     }

```

2 Exercise 3.11

构造下面文法的 LL(1) 分析表。

$$S \rightarrow aBS \mid bAS \mid \varepsilon$$

$$A \rightarrow bAA \mid a$$

$$B \rightarrow aBB \mid b$$

解 列出文法中各非终结符的开始符号和后继符号集合

$$FIRST(S) = \{ a, b, \varepsilon \}$$

$$FOLLOW(S) = \{ \$ \}$$

$$FIRST(A) = \{ a, b \}$$

$$FOLLOW(A) = \{ a, b, \$ \}$$

$$FIRST(B) = \{ a, b \}$$

$$FOLLOW(B) = \{ a, b, \$ \}$$

填写分析表如下

表 1: LL(1) 分析表

	a	b	$\$$
S	$S \rightarrow aBS$	$S \rightarrow bSS$	$S \rightarrow \varepsilon$
A	$A \rightarrow a$	$A \rightarrow bAA$	
B	$B \rightarrow aBB$	$B \rightarrow b$	

3 Exercise 3.16

(a) 用习题 3.1 的文法构造 $(a, (a, a))$ 的最右推导, 说出每个右句型的句柄。

$$S \rightarrow (L) \mid a$$

$$L \rightarrow L, S \mid S$$

解 由此前构造分析树，可以得到句子 $(a, (a, a))$ 的最右推导

$$\begin{aligned} S &\Rightarrow_{\text{lm}} (L) \Rightarrow_{\text{lm}} (L, S) \Rightarrow_{\text{lm}} (L, (L)) \Rightarrow_{\text{lm}} (L, (L, S)) \Rightarrow_{\text{lm}} (L, (L, a)) \\ &\Rightarrow_{\text{lm}} (L, (S, a)) \Rightarrow_{\text{lm}} (L, (a, a)) \Rightarrow_{\text{lm}} (S, (a, a)) \\ &\Rightarrow_{\text{lm}} (a, (a, a)) \end{aligned}$$

给其中的 a 以下标，并给每个右句型的句柄添加下划线

$$\begin{aligned} S &\Rightarrow_{\text{lm}} (\underline{L}) \\ &\Rightarrow_{\text{lm}} (\underline{L}, S) \\ &\Rightarrow_{\text{lm}} (L, \underline{(L)}) \\ &\Rightarrow_{\text{lm}} (L, (\underline{L}, S)) \\ &\Rightarrow_{\text{lm}} (L, (L, \underline{a_3})) \\ &\Rightarrow_{\text{lm}} (L, (\underline{S}, a_3)) \\ &\Rightarrow_{\text{lm}} (L, (\underline{a_2}, a_3)) \\ &\Rightarrow_{\text{lm}} (\underline{S}, (a_2, a_3)) \\ &\Rightarrow_{\text{lm}} (\underline{a_1}, (a_2, a_3)) \end{aligned}$$

4 Non-textbook Exercise

(1) 删除以下文法 G 中的左递归，并由此得到文法 G_1 。

	文法 G : A 是开始符号
1	$A \rightarrow Ba$
2	$B \rightarrow dab$
3	$B \rightarrow Cb$
4	$C \rightarrow cB$
5	$C \rightarrow Ac$

(2) G_1 是否为 $LL(1)$ 的文法？如不是，适当修改该文法 G_1 ，使之成为 $LL(1)$ 的。

解

(1) 用 A 的产生式 $A \rightarrow Ba$ 代换 $C \rightarrow Ac$ 中的 A ，再用产生式 $B \rightarrow Cb$ 代换其中的 B ，用 B 的产生式 $B \rightarrow Cb$ 代换 $A \rightarrow Ba$ 中的 B ，再用产生式 $C \rightarrow Ac$ 代换其中的 C 得到如下文法

	文法 G: A 是开始符号
1	$A \rightarrow Ba$
2	$B \rightarrow dab$
3	$B \rightarrow Cb$
4	$C \rightarrow Cbac \mid dabac \mid cB$

删除其中的直接左递归，得到如下的文法

表 2: 消除左递归的文法 G1

	文法 G: A 是开始符号
1	$A \rightarrow Ba$
2	$B \rightarrow dab$
3	$B \rightarrow Cb$
4	$C \rightarrow dabacC'$
5	$C \rightarrow cBC'$
6	$C' \rightarrow bacC'$
7	$C' \rightarrow \varepsilon$

(2) 列出文法中各非终结符的开始符号和后继符号集合

$$\begin{aligned}
 FIRST(A) &= \{ c, d \} & FOLLOW(A) &= \{ \$ \} \\
 FIRST(B) &= \{ c, d \} & FOLLOW(B) &= \{ a, b \} \\
 FIRST(C) &= \{ c, d \} & FOLLOW(C) &= \{ b \} \\
 FIRST(C') &= \{ b, \varepsilon \} & FOLLOW(C') &= \{ b, \$ \}
 \end{aligned}$$

对于产生式 $B \rightarrow dab \mid Cb$

$$\begin{aligned}
 & FIRST(dab) \cap FIRST(Cb) \\
 &= \{ d \} \cap FIRST(C) \\
 &= \{ d \} \cap \{ c, d \} \\
 &= \{ d \}
 \end{aligned}$$

因此 G1 不是 LL(1) 文法。将 C 的产生式 $C \rightarrow dabacC' \mid C \rightarrow cBC'$ 代入 B 的产生式 $B \rightarrow dab \mid Cb$ ，提左因子得到文法 G2

表 3: 消除左递归并提左因子的 LL(1) 文法 G2

	文法 G: A 是开始符号
1	$A \rightarrow Ba$
2	$B \rightarrow cBC'b$
3	$B \rightarrow dabB'$
4	$B' \rightarrow acC'b$
5	$B' \rightarrow \varepsilon$
6	$C \rightarrow dabacC'$
7	$C \rightarrow cBC'$
8	$C' \rightarrow bacC'$
9	$C' \rightarrow \varepsilon$