编译原理第六章第三次作业

李昊宸 2017K8009929044

6.6.1：在图 6-36 的语法制导定义中添加处理下列控制流构造的规则：

1）一个 repeat 语句：repeat S while B

2）一个 for 循环语句：for (S1; B; S2) S3

答：

1)

PRODUCTION	SEMANTIC RULES
S -> repeat S1 while B	S1.next = newlabel（） B.true = newlabel（） B.false = S.next S.code = label(B.true) || S1.code || label(S1.next) || B.code

2)

PRODUCTION	SEMANTIC RULES
S -> for (S1 ; B ; S2) S3	S1.next = newlabel () S2.next = S.next S3.next = newlabel () B.true = newlabel() B.false = S.next S.code = S1.code || label(S1.next) || B.code || label(B.true) || S3.code || label(S3.next) || S2.code || gen('goto' S1.next)

6.7.1: 使用图 6-43 中的翻译方案翻译下列表达式。给出每个子表达式的 truelist 和 falselist。你可以假设第一条被生成的指令的地址是 100.

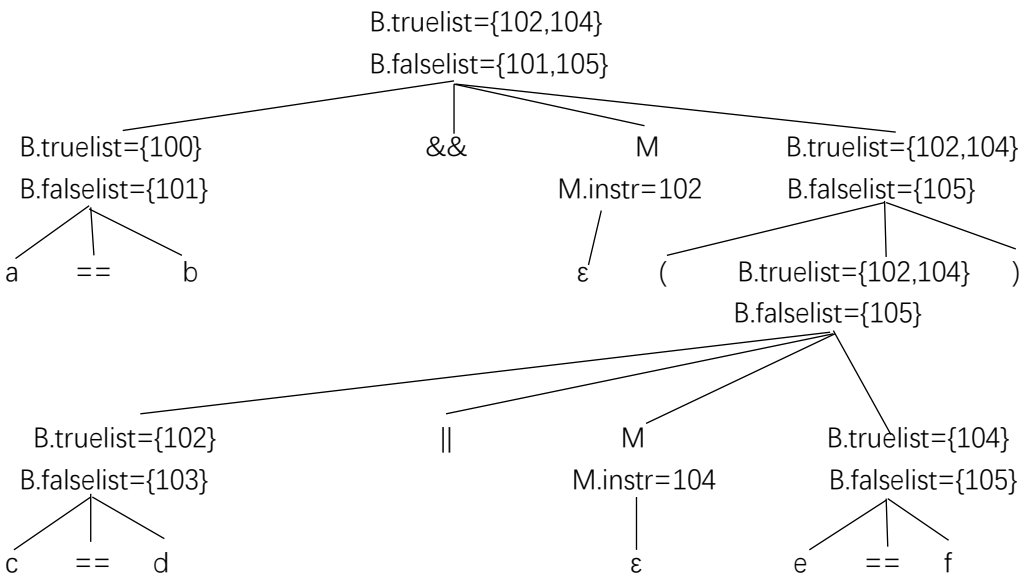
1) a==b && (c==d || e==f)
2) (a==b || c==d) || e==f
3) (a==b && c==d) && e==f

1)	$B \rightarrow B_1 \mid\mid M\ B_2$	{ *backpatch*(B_1.*falselist*, M.*instr*); B.*truelist* = *merge*(B_1.*truelist*, B_2.*truelist*); B.*falselist* = B_2.*falselist*; }
2)	$B \rightarrow B_1$ && $M\ B_2$	{ *backpatch*(B_1.*truelist*, M.*instr*); B.*truelist* = B_2.*truelist*; B.*falselist* = *merge*(B_1.*falselist*, B_2.*falselist*); }
3)	$B \rightarrow !\ B_1$	{ B.*truelist* = B_1.*falselist*; B.*falselist* = B_1.*truelist*; }
4)	$B \rightarrow (\ B_1\ )$	{ B.*truelist* = B_1.*truelist*; B.*falselist* = B_1.*falselist*; }
5)	$B \rightarrow E_1$ **rel** E_2	{ B.*truelist* = *makelist*(*nextinstr*); B.*falselist* = *makelist*(*nextinstr* + 1); *gen*('`if` E_1.*addr* **rel**.*op* E_2.*addr* '`goto _`'); *gen*('`goto _`'); }
6)	$B \rightarrow$ **true**	{ B.*truelist* = *makelist*(*nextinstr*); *gen*('`goto _`'); }
7)	$B \rightarrow$ **false**	{ B.*falselist* = *makelist*(*nextinstr*); *gen*('`goto _`'); }
8)	$M \rightarrow \epsilon$	{ M.*instr* = *nextinstr*; }

图 6-43 布尔表达式的翻译方案

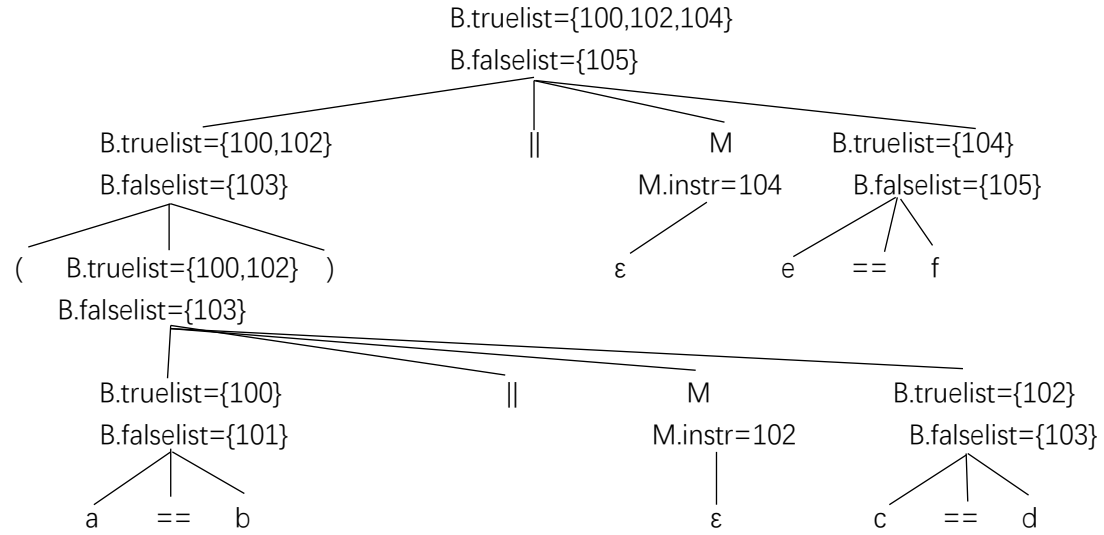
答:

1）a==b && (c==d || e==f)



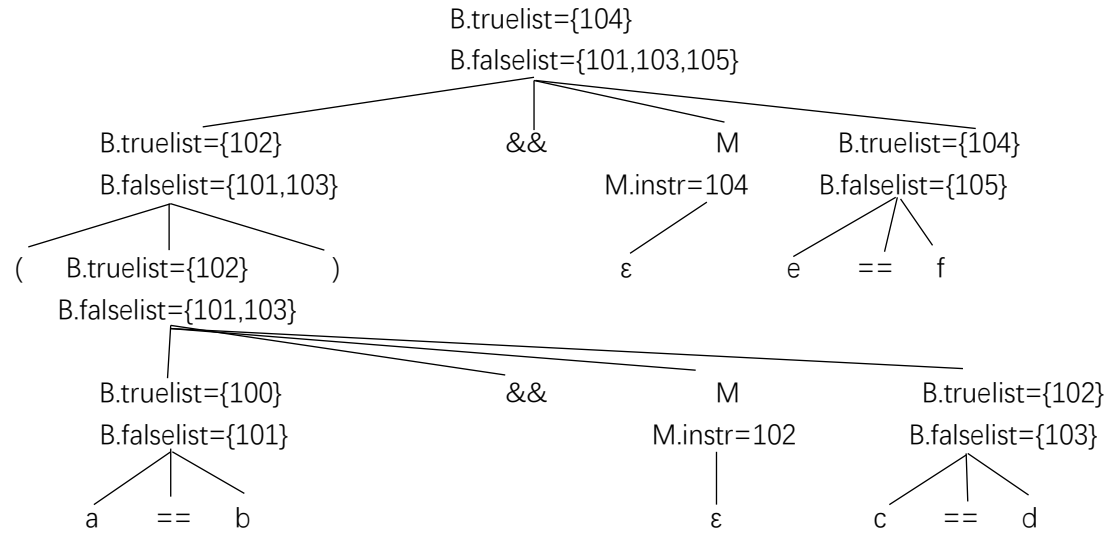
100: if a == b goto 102
101: goto –
102: if c == d goto –
103: goto 104
104: if e == f goto –
105: goto -

2）(a==b || c==d) || e==f



100: if a == b goto -
101: goto 102
102: if c == d goto –
103: goto 104
104: if e == f goto –
105: goto -

3) (a==b && c==d) && e==f



100: if a == b goto 102
101: goto -
102: if c == d goto 104
103: goto -
104: if e == f goto –
105: goto -