

第六章（中间代码生成）作业

6-3

P263: 6.6.1 (中文版厚书)

P246: 6.6.1 (中文版薄书)

练习 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**。

注: 原题第 1) 小题。

产生式	语义规则
$P \rightarrow S$	$S.next = newlabel()$ $P.code = S.code \parallel label(S.next)$
$S \rightarrow \text{assign}$	$S.code = \text{assign}.code$
$S \rightarrow \text{if} (B) S_1$	$B.true = newlabel()$ $B.false = S_1.next = S.next$ $S.code = B.code \parallel label(B.true) \parallel S_1.code$
$S \rightarrow \text{if} (B) S_1 \text{ else } S_2$	$B.true = newlabel()$ $B.false = newlabel()$ $S_1.next = S_2.next = S.next$ $S.code = B.code$ $\quad \parallel label(B.true) \parallel S_1.code$ $\quad \parallel gen('goto' S.next)$ $\quad \parallel label(B.false) \parallel S_2.code$
$S \rightarrow \text{while} (B) S_1$	$begin = newlabel()$ $B.true = newlabel()$ $B.false = S.next$ $S_1.next = begin$ $S.code = label(begin) \parallel B.code$ $\quad \parallel label(B.true) \parallel S_1.code$ $\quad \parallel gen('goto' begin)$
$S \rightarrow S_1 S_2$	$S_1.next = newlabel()$ $S_2.next = S.next$ $S.code = S_1.code \parallel label(S_1.next) \parallel S_2.code$

图 6-36 控制流语句的语法制导定义

P263: 6.6.1 (中文版厚书)

P246: 6.6.1 (中文版薄书)

练习 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:

! 2) 一个 for 循环语句, $\text{for} (S_1; B; S_2) S_3$ 。

注: 原题第 2) 小题。

产生式	语义规则
$P \rightarrow S$	$S.next = \text{newlabel}()$ $P.code = S.code \parallel \text{label}(S.next)$
$S \rightarrow \text{assign}$	$S.code = \text{assign.code}$
$S \rightarrow \text{if} (B) S_1$	$B.true = \text{newlabel}()$ $B.false = S_1.next = S.next$ $S.code = B.code \parallel \text{label}(B.true) \parallel S_1.code$
$S \rightarrow \text{if} (B) S_1 \text{ else } S_2$	$B.true = \text{newlabel}()$ $B.false = \text{newlabel}()$ $S_1.next = S_2.next = S.next$ $S.code = B.code$ $\parallel \text{label}(B.true) \parallel S_1.code$ $\parallel \text{gen}('goto' S.next)$ $\parallel \text{label}(B.false) \parallel S_2.code$
$S \rightarrow \text{while} (B) S_1$	$begin = \text{newlabel}()$ $B.true = \text{newlabel}()$ $B.false = S.next$ $S_1.next = begin$ $S.code = \text{label}(begin) \parallel B.code$ $\parallel \text{label}(B.true) \parallel S_1.code$ $\parallel \text{gen}('goto' begin)$
$S \rightarrow S_1 S_2$	$S_1.next = \text{newlabel}()$ $S_2.next = S.next$ $S.code = S_1.code \parallel \text{label}(S_1.next) \parallel S_2.code$

图 6-36 控制流语句的语法制导定义

P268: 6.7.1 (中文版厚书)

P251: 6.7.1 (中文版薄书)

练习 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) $a==b \ \&\& \ (c==d \ || \ e==f)$

注: 原题第 1) 小题。

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

图 6-43 布尔表达式的翻译方案