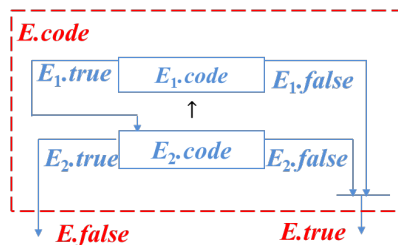


编译原理参考答案

清华大学出版社

第八章 静态语义分析和中间代码生成

4. 参考 8.3.3.4 节采用短路代码进行布尔表达式翻译的 L-翻译模式片段及用到的语义函数。若在基础文法中增加产生式 $E \rightarrow E \uparrow E$ ，试给出与该产生式相应的语义动作集合。其中， \uparrow 代表“与非”逻辑算符，其语义可用其他逻辑运算定义为 $P \uparrow Q = \text{not}(P \text{ and } Q)$ 。
答：



$E \rightarrow \{E_1.\text{true} = \text{newlabel}(); E_1.\text{false} = E.\text{true}\} E_1 \uparrow \{\text{lable}(E_1.\text{true}); E_2.\text{true} = E.\text{false}; E_2.\text{false} = E.\text{true}\} E_2$

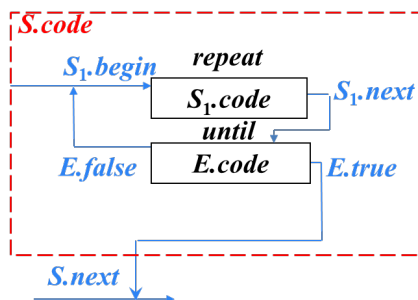
或者

$E \rightarrow \{E_1.\text{false} := E.\text{true}; E_1.\text{true} := \text{newlabel};\} E_1$
 $\uparrow \{E_2.\text{false} := E.\text{true}; E_2.\text{true} := E.\text{false};\} E_2$
 $\{E.\text{code} := E_1.\text{code} || \text{gen}(E_1.\text{true}';') || E_2.\text{code}\}$

5. 参考 8.3.3.5 节进行控制语句（不含 break）翻译的 L-翻译模式片段及所用到的语义函数。若在基础文法中增加产生式 $S \rightarrow \text{repeat } S \text{ until } E$ ，试给出与该产生式相应的语义动作集合。

注：控制语句 $\text{repeat} \langle \text{循环体} \rangle \text{until} \langle \text{布尔表达式} \rangle$ 的语义为：至少执行 $\langle \text{循环体} \rangle$ 一次，直到 $\langle \text{布尔表达式} \rangle$ 成真时结束循环。

答：

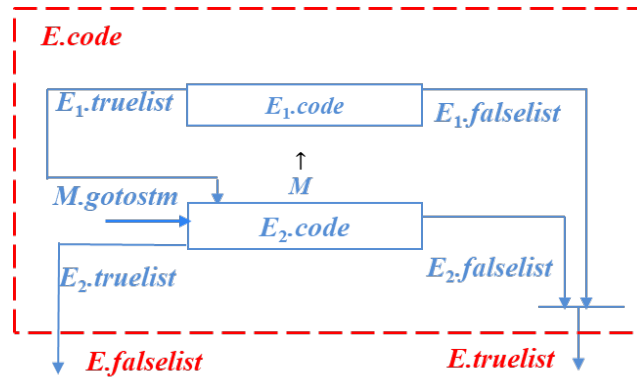


$S \rightarrow \text{repeat} \{ S_1.\text{begin} = \text{newlabel}(); \text{lable}(S_1.\text{begin}); S_1.\text{next} = \text{newlabel}(); \} S_1 \text{until} \{ \text{lable}(S_1.\text{next}); E.\text{true} = S.\text{next}; E.\text{false} = S_1.\text{begin} \} E$

或者

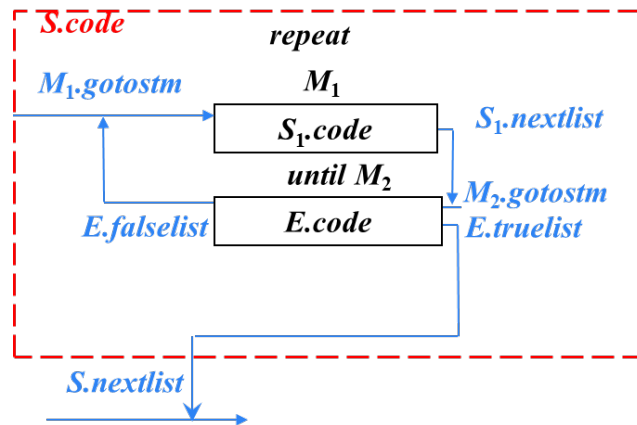
$S \rightarrow \text{repeat} \{ S_1.\text{next} := \text{newlabel}; \} S_1 \text{until} \{ E.\text{true} := S.\text{next}; E.\text{false} := \text{newlabel}; \} E \{ S.\text{code} := \text{gen}(E.\text{false} ':') || S_1.\text{code} || \text{gen}(S_1.\text{next} ':') || E.\text{code} \}$

6. 参考 8.3.3.6 节采用拉链与代码回填技术进行布尔表达式和控制语句翻译的 S-翻译模式片段及所用到的语义函数，重复题 4 和题 5 的工作。



$E \rightarrow E_1 \uparrow M E_2 \{ \text{backpatch}(E_1.\text{truelist}, M.\text{gotostm}); E.\text{truelist} = \text{Merge}(E_1.\text{falselist}, E_2.\text{falselist}); E.\text{falselist} = E_2.\text{truelist} \}$

$M \rightarrow \varepsilon \{ M.\text{gotostm} = \text{nextstm}; \}$



$S \rightarrow \text{repeat } M_1 S_1 \text{ until } M_2 E$

$\{ \text{backpatch}(E.\text{falselist}, M_1.\text{gotostm}); \text{backpatch}(S_1.\text{nextlist}, M_2.\text{gotostm}); S.\text{nextlist} = E.\text{truelist} \}$

$M \rightarrow \varepsilon \{ M.\text{gotostm} = \text{nextstm}; \}$