$B \rightarrow B_1$ || $M\ B_2 \mid B_1$ && $M\ B_2 \mid$! $B_1 \mid$ (B_1) $\mid E_1$ **rel** $E_2 \mid$ **true** $\mid$ **false**
$M \rightarrow \epsilon$

1) $B \rightarrow B_1$ || $M\ B_2$ { *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*; }

1) $S \rightarrow$ **if** (B) $M\ S_1$ { *backpatch*(*B*.*truelist*, *M*.*instr*);
S.*nextlist* = *merge*(*B*.*falselist*, S_1.*nextlist*); }

2) $S \rightarrow$ **if** (B) $M_1\ S_1\ N$ **else** $M_2\ S_2$
{ *backpatch*(*B*.*truelist*, M_1.*instr*);
backpatch(*B*.*falselist*, M_2.*instr*);
temp = *merge*(S_1.*nextlist*, *N*.*nextlist*);
S.*nextlist* = *merge*(*temp*, S_2.*nextlist*); }

3) $S \rightarrow$ **while** M_1 (B) $M_2\ S_1$
{ *backpatch*(S_1.*nextlist*, M_1.*instr*);
backpatch(*B*.*truelist*, M_2.*instr*);
S.*nextlist* = *B*.*falselist*;
gen('`goto` M_1.*instr*); }

4) $S \rightarrow$ { L } { *S*.*nextlist* = *L*.*nextlist*; }

5) $S \rightarrow A$; { *S*.*nextlist* = **null**; }

6) $M \rightarrow \epsilon$ { *M*.*instr* = *nextinstr*; }

7) $N \rightarrow \epsilon$ { *N*.*nextlist* = *makelist*(*nextinstr*);
gen('`goto _`'); }

8) $L \rightarrow L_1\ M\ S$ { *backpatch*(L_1.*nextlist*, *M*.*instr*);
L.*nextlist* = *S*.*nextlist*; }

9) $L \rightarrow S$ { *L*.*nextlist* = *S*.*nextlist*; }