1. B🡪B1 or T

{

Backpatch(B1．false,T．codebegin);

B．codebegin = B1．codebegin;

B．ture = merge(B1．true,T．true);

B．false = T．false;

}

(2) B 🡪 T

{

B．true = T．true;

B．codebegin = T．codebegin;

B．false = T．false;

}

(3) T🡪T1 and F

{

Backpatch(T1．true,F．codebegin);

T．codebegin = T1．codebegin;

T．true = F．true;

T．false = merge(T1．false,F．false);

}

(4) T🡪F

{

T．true = F．true;

T．codebegin = F．codebegin;

T．false = Ffalse;

}

(5) F🡪not F1

{

F．true = F1．false;

F．codebegin = F1．codebegin;

F．false = F1．true;

}

(6) E🡪(E1)

{

E．true = E1．true;

E．codebegin = E1．codebegin;

E．false = E1．false;

}

(7) E🡪id1 rop id2

{

E．true = nextstat;

E．codebegin = nextstat;

E．false = neststat+1;

Emit(‘If’id1．place’rop’id2．place’goto’-);

Emit(‘goto’-);

}

(8) E🡪true

{

E．true = nextstat;

E．codebegin = nextstat;

Emit(‘goto’-);

}

(9) E🡪false

{

E．false = nextstat;

E．codebegin = nextstat;

Emit(‘goto’-);

}