Boolean Expressions

$$E \rightarrow E \text{ or } E \mid E \text{ and } E \mid \text{not } E \mid (E) \mid \text{id relop id} \mid$$
$$\text{true} \mid \text{False}$$

Translated in 2 ways,

  a) numerical representation

  b) control flow representation.

Numerical representation

$1 \rightarrow$ true, $0 \rightarrow$ false. Expressions will be

evaluated from left to right.

$a \text{ or } b \text{ and not } c$ $\Big\}$ three-address code

$t_1 = \text{not } c$

$t_2 = b \text{ and } t_1$

$t_3 = a \text{ or } t_2$

relational expression - if $a < b$ then 1 else 0.

$\left. \begin{array}{l} \end{array} \right\}$ three-address code

100 : if $a < b$ goto 103

101 : $t := 0$

102 : goto 104

103 : $t := 1$

104 :

Translation scheme using numerical representation for booleans.

$E \rightarrow E_1$ or E_2
{E.place := newtemp;
emit (E.place ': =' E_1.place 'or' E_2.place)}.

$E \rightarrow E_1$ and E_2
{E.place = new temp;
emit (E.place ': =' E_1.place 'and' E_2.place)}

$E \rightarrow$ not E_1
{E.place := newtemp;
emit (E.place ': =' 'not' E_1.place)}

$E \rightarrow (E_1)$
{E.place := E_1.place}

$E \rightarrow id_1$ relop id_2
{E.place := new temp;
emit ('if' id_1.place relop.op id_2.place 'goto' nextstat+3);
emit (E.place ': =' '0');
emit ('goto' nextstat + 2);
emit (E.place ':=' '1')}

$E \rightarrow$ true
{E.place := newtemp;
emit (E.place ':=' 1)}

$E \rightarrow false$

control flow representation

$$S \to \text{if } E \text{ then } S_1$$
$$\text{if } E \text{ then } S_1 \text{ else } S_2 \qquad \text{if else}$$
$$\text{while } E \text{ do } S_1 \qquad \text{while}$$

Syntax - directed definition for flow of control statements.

production	semantic rules
$S \to$ if E then S	$E.\text{true} := \text{newlabel};$
if (con) state	$E.\text{false} := S.\text{next};$
else S	$S_1.\text{next} := S.\text{next};$
	$S.\text{code} := E.\text{code} \,\|\,$
	$\text{gen}(E.\text{true} ':') \,\|\, S_1.\text{code}$

$S \rightarrow$ if E then S else S_2

$\qquad$ $E.true := newlabel;$

$\qquad$ $E.false := newlabel;$

$\qquad$ $S_1.next := S.next;$

$\qquad$ $S_2.next := S.next;$

$\qquad$ $S.code := E.code \, || \, gen(E.true ':') \, ||$

$\qquad\qquad$ $S_1.code \, || \, gen('goto' \, S.next)$

$\qquad\qquad$ $|| \, gen(E.false ':') \, || \, S_2.code$

$S \rightarrow$ while E do S_1
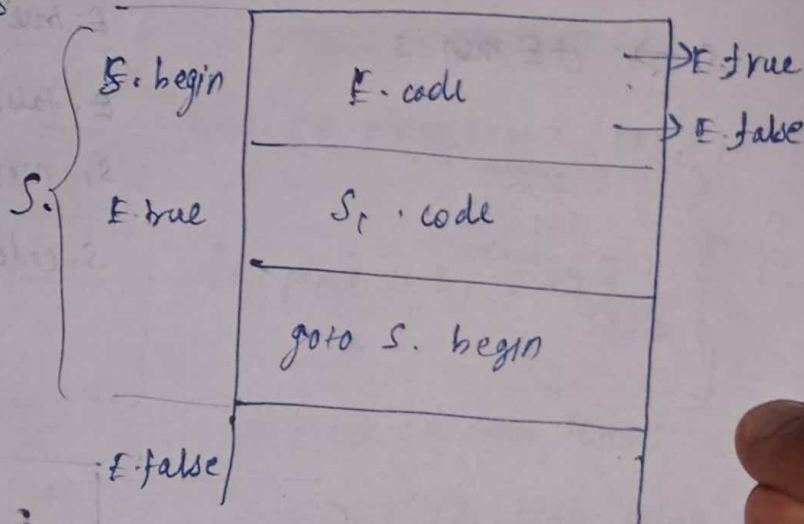
$S.begin := newlabel;$

$E.true := newlabel;$

$E.false := S.next;$

$S_1.next := S.begin;$

$S.code := gen(S.begin ':') \, || \, E.code \, ||$

$\qquad\qquad$ $gen(E.true ':') \, || \, S_1.code \, ||$

$\qquad\qquad$ $gen('goto' \, S.begin)$

Control Flow translation of Boolean Expression:

SDD to produce three-address code for booleans.

$E \rightarrow E_1 \text{ or } E_2$

$E_1.true := E.true$

$E_1.false := newfalselabel$

$E_2.true := E.true$

$E_2.false := E.false$;

$E.code := E_1.code \, || \, gen(E_1.false :)$

$|| \, E_2.code .$

0	0	0
0	1	1
1	0	1
1	1	1

$E \rightarrow E_1 \text{ and } E_2$

$E.true := newlabel ;$

$E_1.false := E.false ;$

$E_2.true := E.true ;$

$E_2.false := E.false ;$

$E.code := E_1.code \, || \, gen(E_1.true :)$

$|| \, E_2.code .$

0	0	0
0	1	0
1	0	0
1	1	1

$E \rightarrow \text{not } E_1$

$E_1.true := E.false$

$E_1.false := E.true ;$

$E.code := E_1.code .$

$E \rightarrow (E_1)$

$E_1.true := E.true ;$

$E_1.false := E.false ;$

$E.code := E_1.code$

$E \longrightarrow id_1 \; relop \; id_2$

$E.code := gen ('if' \; id_1 \cdot place$

$\qquad relop.op \; id_2 \cdot place$

$\qquad 'goto' \; E.true) \; || \; gen$

$\qquad ('goto' \; E.false)$

$E \longrightarrow true \qquad E.code := gen ('goto' \; E.true)$

$E \longrightarrow false \qquad E.code := gen ('goto' \; E.false).$