Short Circuit Evaluation for Boolean Expressions

- (exp1 && exp2): value = if ($\sim$exp1) then FALSE else exp2
    - This implies that exp2 need not be evaluated if exp1 is FALSE
- (exp1 || exp2):value = if (exp1) then TRUE else exp2
    - This implies that exp2 need not be evaluated if exp1 is TRUE
- Since boolean expressions are used mostly in conditional and loop statements, it is possible to realize perform short circuit evaluation of expressions using control flow constructs
- In such a case, there are no explicit '||' and '&&' operators in the intermediate code (as earlier), but only jumps
- Much faster, since complete expression is not evaluated
- If unevaluated expressions have side effects, then program may have non-deterministic behaviour

Control-Flow Realization of Boolean Expressions

if ((a+b < c+d) || ((e==f) && (g > h-k))) A1; else A2; A3;

```
100:         T1 = a+b
101:         T2 = c+d
103:         if T1 < T2 goto L1
104:         goto L2
105:L2:      if e==f goto L3
106:         goto L4
107:L3:      T3 = h-k
108:         if g > T3 goto L5
109:         goto L6
110:L1:L5:   code for A1
111:         goto L7
112:L4:L6:   code for A2
113:L7:      code for A3
```