1 Identify the basic blocks that correspond to the while loop. How is

the GIMPLE for a while loop different from simple if-else jumps?

Ans

BB 3 and 4 correspond to the while loop. BB2 avoids executing an iteration

unconditionally by having an unconditional goto to BB4which is the while

loop conditional. This is different from the if-else jumps,which always seem

to be conditional.

2 Identify the basic blocks for the nested for-loop. You should be

able to identify intializers, conditions and increments for each

loop as well as the shared inner body. Is there really anything

special about a for-loop as compared to a while-loop?

BB 5: Outer loop induction variable initialisation

BB 10: Outer loop conditional check

BB 6: Inner loop induction variable initialisation

BB 8: Inner loop conditional check

BB 7: Inner loop body + induction variable increment

BB 9: Outer loop induction variable increment

The control-flow graph dump shows distinct basic blocks which are

named sequentially starting from 1. Basic blocks roughly correspond

to labels that are created during the gimplification of control

statements.

f ()

{

int c;

int b;

int a;

int Z.0;

int D.1715;

<bb 2>:

a = Z;

if (a <= 9)

goto <bb 3>;

else

goto <bb 4>;

<bb 3>:

b = 5;

c = 17;

goto <bb 6>;

<bb 4>:

b = 6;

c = 20;

if (a == 0)

goto <bb 5>;

else

goto <bb 6>;

<bb 5>:

c = 0;

<bb 6>:

D.1715 = b \* 10;

Z.0 = D.1715 + c;

Z = Z.0;

return;

}