# Gimple Code

;; Function main (main, funcdef\_no=0, decl\_uid=1794, cgraph\_uid=0, symbol\_order=0)

;; 4 loops found

;;

;; Loop 0

;; header 0, latch 1

;; depth 0, outer -1

;; nodes: 0 1 2 3 4 5 6 7 8 9 10 11 12

;;

;; Loop 2

;; header 10, latch 9

;; depth 1, outer 0

;; nodes: 10 9 8 6 7

;;

;; Loop 3

;; header 8, latch 7

;; depth 2, outer 2

;; nodes: 8 7

;;

;; Loop 1

;; header 4, latch 3

;; depth 1, outer 0

;; nodes: 4 3

;; 2 succs { 4 }

;; 3 succs { 4 }

;; 4 succs { 3 5 }

;; 5 succs { 10 }

;; 6 succs { 8 }

;; 7 succs { 8 }

;; 8 succs { 7 9 }

;; 9 succs { 10 }

;; 10 succs { 6 11 }

;; 11 succs { 12 }

;; 12 succs { 1 }

main ()

{

int j;

int i;

int sum;

int D.1809;

<bb 2> [0.00%]:

sum = 0;

goto <bb 4>; [0.00%]

<bb 3> [0.00%]:

sum = sum \* 2;

<bb 4> [0.00%]:

if (sum <= 99)

goto <bb 3>; [0.00%]

else

goto <bb 5>; [0.00%]

<bb 5> [0.00%]:

i = 0;

goto <bb 10>; [0.00%]

<bb 6> [0.00%]:

j = 0;

goto <bb 8>; [0.00%]

<bb 7> [0.00%]:

\_1 = i \* j;

sum = sum + \_1;

j = j + 1;

<bb 8> [0.00%]:

if (j <= 49)

goto <bb 7>; [0.00%]

else

goto <bb 9>; [0.00%]

<bb 9> [0.00%]:

i = i + 1;

<bb 10> [0.00%]:

if (i <= 24)

goto <bb 6>; [0.00%]

else

goto <bb 11>; [0.00%]

<bb 11> [0.00%]:

D.1809 = 0;

<L9> [0.00%]:

return D.1809;

}

# C Code

int main()

{

int sum = 0;

int i, j;

while(sum < 100) {

sum = sum \* 2;

}

for(i=0; i<25; i++) {

for (j=0; j<50; j++) {

sum = sum + i\*j;

}

}

}

# Answers:

# 1]

The blocks <bb 2>, <bb 3>, <bb 4> represent the while loop. A while is nothing but blocks with conditional control transfer. In the gimple code we see that there is jump from one block to other block taking into consideration true or false of the if or else statement it may also be a backward jump.

# 2]

The blocks 5,6,7,8,9,10 are the blocks for for loop. Blocks 5,9,10 are for outer for loops and and 6,7,8 for inner for loop. A for loop is a while loop with initializer separated out in one block ,the body of the loop ,the increment part at the end of the loop body and then jump back to the start of the loop.