# GIMPLE

main ()

{

int a;

int b;

int c;

int p;

int q;

double r;

p = 6;

a = 10;

b = 20;

\_1 = a \* b;

c = \_1 + 25;

p = 6;

q = Z;

r = 3.45e+1;

\_2 = (int) r;

Z = \_2;

Z.0\_3 = Z;

\_4 = Z.0\_3 + 1;

Z = \_4;

}

# Input Code

int Z;

void main()

{

int a, b, c;

int p = 6;

int q;

double r;

a = 10;

b = 20;

c = a \* b + 25;

p = 6;

q = Z;

r = 34.5;

Z = r;

Z = Z + 1;

}

# Answer1]

Every variable is declared as independent variable at top even though they are declared all together at once in the code. Temporary variables are not declared at top (example \_1). Global variable is

not declared i.e. Z is not declared at top.

# Answer2]

Only one operation is performed in one statement. Complex expressions are broken down by using temporary variables. C expressions are converted to GIMPLE statements by considering syntax.

Whenever there are complex expressions temporary variables are introduced.

# Answer3]

Floats/doubles are displayed in scientific notation, and implicit integers are converted to explicit integers in GIMPLE.

# Answer4]

Read/write from/to global variable requires memory storage. GIMPLE has only one operation in one statement considering assignment also. Hence any operation on any variable has to go through indirect temporary assignment. In Z=p or q=Z there is only assigning of a value to a variable and no additional operation as in Z=Z+1.