Problem I

For the following C statement, what is the corresponding MIPS assembly code? Assume that the variables f, g, h, and i are given and could be considered 32-bit integers as declared in a C program. Use a minimal number of MIPS assembly instructions.

```
f = g + ( h – 5 );
```

Problem II

Write the MIPS assembly code that creates the 32-bit constant

$$0010\ 0000\ 0000\ 0001\ 0100\ 1001\ 0010\ 0100_{two}$$

and stores that value to register $t1. If the current value of the PC is 0x00000600, can you use a single branch instruction to get to the above address

Problem III

Assume that the stack and the static data segments are empty and that the stack and global pointers start at address 0x7fff ffc and 0x1000 8000, respectively. Assume the calling conventions are specified in Figure 2.11 in the textbook and that function inputs are passed using registers $a0-$a3 and returned in register $v0. Assume that leaf functions may only use saved registers.

```
int my_global = 100;
main()
{
     int a = 10;
     int b = 20;
     int c = 30;
     int d;
     d = my_function(a, b, c);
}
int my_function(int a, int b, int c)
{
     return a-b+c+my_global;
}
```

(1) Write MIPS assembly code for the code in the table above
(2) Show the content of the stack and the static data segments after each function call
(3) If the leaf function could use temporary registers ($t0, $t1, etc), write the MIPS code for the code in the table above.