PART 2: Evaluating Execution Time (Objective - 1) (25 Points)

A)Complete the following table. Provide how many instructions of the given categories are executed while finding the GCD(1071, 462) for both loop and recursive functions, seperately. You should provide two tables, one for non-recursive the other for recursive function.
(5 Points)

For loop funtciton

Category	Instructions	CPI(Clock Cycle per instruction)	Instruction Count
A: Addition, subtraction and comparison	add, addu, sub, slt, etc..	1	14
B: Multiplication and division	div, mul, mult	4	0
C: Memory	lw, sw	8	12
D: Branch and Jump	beq, bnq, j, jal	2	26

For recursive function

Category	Instructions	CPI(Clock Cycle per instruction)	Instruction Count
A: Addition, subtraction and comparison	add, addu, sub, slt, etc..	1	18
B: Multiplication and division	div, mul, mult	4	4
C: Memory	lw, sw	8	15
D: Branch and Jump	beq, bnq, j, jal	2	21

B) Then calculate the total clock cycles for each function (non-recursive and non-recursive), seperately. (10 Points)

Total Cycle Cyles = (1x14) + (4x0) + (8x12) + (2x26) = 162   (for loop)

Total Cycle Cyles = (1x18) + (4x4) + (8x15) + (2x21) = 196 (for recursive)

C) Calculate how much faster would the functions be if a better data cache reduced the average memory operations time to 4 cycles? (10 Points)

Total Cycle Cyles = (1x14) + (4x0) + (4x12) + (2x26) =114   (for loop)   Average

162/144 * 100 = 142% faster   144 + 142 / 2 = 143

Total Cycle Cyles = (1x18) + (4x4) + (4x15) + (2x21) = 136 (for recursive)   So,in average it

196/136 * 100 = 144% faster   increasing speed by 143

PART 3: Representing Numbers (Objective - 3) (15 Points)

A) Show the two's complement representations and their hexadecimal forms for Maximum and Minimum integer values (5 Points)

Maximum integer value is 2,147,483,647
2's complement : 01111111111111111111111111111111
Hexadecimal form: $(7fffffff)_{16}$

Minimum integer value is -2,147,483,648
2's complement : 10000000000000000000000000000000
Hexadecimal form : -80000000

B) Show the 32 bit floating point representation of decimal number -11.1ten (10 Points)

-11.1 = -1.11 x 2^1 (Normalized)

1 XXXXXXX   11000000000000000000000

   8 bits for act exp   23 bits for fraction

act exp=exp-bias

1=exp-127

Exp=128=10000000

$(-11.1)_{10}$=1 10000000 11000000000000000000000