United International University
Department of Computer Science and Engineering
CSE 313: Computer Architecture

Mid Term Examination Set: A Time: 1 Hour 45 Minutes

1. (a) Determine whether **overflow** occurs in the following **signed** operations in a 4-bit Computer Architecture with detailed calculations. [3]
(i) 8 - (-2) (ii) 9 - 3 (iii) -5 + (-6)

   (b) Show the detailed step of simulations for the **Optimized Multiplication Algorithm** for the following multiplication: 110111 × 110001 [3]

   (c) You have to divide 1101 by 0101. Find out the content of Remainder Register after the **Third step** of the division algorithm that uses the following block diagram at Figure 1. Show all necessary calculations. [4]

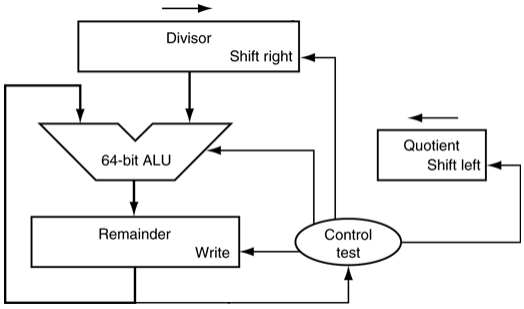

Figure 1: Division Algorithm Block Diagram

2. (a) A given application written in Java runs 15 seconds on a desktop processor. A new Java compiler is released that requires only 0.6 as many instructions as the old compiler. Unfortunately, it increases the CPI by 1.1. Calculate **how fast** can we expect the application to run using this new compiler? [2]

   (b) Calculate the **Execution Time** of the program with the following instructions on a 2 GHz processor. [3+2]

	Arithmetic	Load	Store	Branch
IC	50	110	80	16
CPI	1	1	4	2

We want to run the program 2 (two) times faster. Calculate how much we need to improve the **CPI** of **Arithmetic Instructions** for achieving the target.

   (c) Explain how has the **Power Wall** affected the computer designs. Write down the solution that was implemented by Computer Architects to solve this crisis. [2+1]

3. (a) A student of CSE, UIU has just learnt C programming who has no idea how the instructions of C programming are represented in memory. So he asked for your help. Your task is to convert the following C instructions into equivalent **Machine Code** in order to help your junior. [3]

```
F = A + B - C;
Arr[20] = Arr[5] - F;
```

You **must use** the following Reference Table 1 for required opcode and funct codes.

Instruction	Op Code	Funct Code
ADD	0	32
SUB	0	34
LW	35	-
SW	85	-

Table 1: Instruction Reference

(b) Now imagine after knowing MIPS, he wants to have **similar structure** for both I-type and R-type instructions. Write down 2 (two) **problems** that he would face. [2]

4. Consider the following code segments. Write down equivalent **MIPS code** for the following program. [5]

```
int main()
{
    int n = 4, m = 5 ;
    int result = add(n, m) ;
    return 0 ;
}

int add(int n, int m)
{
    if(n == 0) return 0 ;
    else return m+add(n-1, m) ;
}
```