CS 240 – Computer Organization

Lab 2 – Number Handling on MIPS

The primary goal of this lab is to learn how to use low-level MIPS assembly instructions to implement *logical* and *conditional statements*. This lab comes in three parts. There are 100 points in total.

Your tasks:

1) Complete the count_ones function where you are going to return the number of 1's inside an integer (32-bit number). **(30 points)**
   - For example, 0xFFFFFFFF should return 32.
   - You will need to use loops and branching for this one.

2) Complete the bcd2bin function where a 32-bit number, which is presenting 8 BCD numbers, is given and you need to return a binary number. **(40 points)**
   - For example,

     0x76543210 should return 0x48FF4EA or 10010001111111010011101010

3) Complete the bin2bcd function where a 32-bit binary number is given, and you should return an equivalent BCD number. **(40 points)**

You can expect the input to your function in register $t0. The result must be stored inside $t0 as well.

Submissions:

- Complete the provided lab2.s code and turn it in to canvas.

Important Things to Consider

1) Don’t forget to update the student name and id variables. These are used for grading purposes. FAILURE TO ADD NAME AND ID RESULTS IN NO SUBMISSION

```
student_name: .asciiz "Your Name"
student_id: .asciiz "Your Student ID"
```

2) Since we use a computer program to test your code, you are **ONLY** allowed to modify the highlighted area in the source code.
For example:

```
count_ones:
move $t0, $a0
############## Part 1: your code begins here #################
Your stuff goes here :D
############## Part 1: your code ends here  ##################
move $v0, $t0
jr $ra
```

3) Once completed, submit the lab2.s file to canvas.