

C335 Homework #4

Points:	: 40 points
Due Date:	: Mar. 10 th
Submissions:	: Hardcopy (please type or write your solution clearly)

PART I (4 POINTS)

Given the following two 32 bit binary numbers:

1000 1101 0010 1000 0000 0000 0000 0000

0010 1000 1000 1000 0000 0000 0000 0001

Find the MIPS assembly instruction represented by each number.

PART II (6 POINTS)

Implement the following C code in MIPS assembly language, assuming that *compare* is the first function called:

```
int compare (int a, int b) {
    if (sub(a, b) >= 0)
        return 1;
    else
        return 0;
}

int sub (int a, int b) {
    Return a-b;
}
```

PART III (8 POINTS)

Draw the truth table and the logic circuit for the following function

$$F = A' \cdot B' + A \cdot B$$

(Note, for the logic circuit part, you could draw it by hand)

PART IV (8 POINTS)

Draw the truth table and the logic circuit for the following function

$$F = (A + B) \cdot (A' + C')$$

(Note, for the logic circuit part, you could draw it by hand)

PART V (6 POINTS)

Use perfect induction to prove $x \cdot (x' + y) = x \cdot y$

PART VI (8 POINTS)

We discussed the logical equations in **minterm form** for CarryOut and Sum of the 1-bit full adder. Can you write down the logical equations for these two output signals in **maxterm form**?

PART VII (BONUS 5 POINTS)

The following is a C code segment doing bubble sorting on an integer array:

```
for (i = 0; i < n - 1 ; i ++)  
{  
    for (j = 0; j < n - i - 1; j ++)  
    {  
        If (array[j] > array[j+1])  
        {  
            temp = array[j];  
            array[j] = array[j+1];  
            array[j+1] = temp;  
        }  
    }  
}
```

Compile this code segment into MIPS assembly language, assume integer variable *n* is in \$s0, integer variable *i* is in \$s1, integer variable *j* is in \$s2, the base address of integer array *array* is in \$s3.