(5 pts) Exercise 2-31

- **Suppose you are given the code for the following function:**
  int function1(int a, int b);

Write MIPS code to <u>call</u> function1(3, 7) and then store the result in $s0

(5 pts) Exercise 2-32

- **Now you have this definition for function1:**

```
int function1(int a, int b) {
    return (a – b);
}
```

Write MIPS code to <u>define</u> function1.

(10 pts) Exercise 2-33

- Write MIPS code to define the following function:

```
int cat(int a, int b) {
    if (a < b)
      return a;
    else
      return b;
    }
```

(5 pts) Exercise 2-36

- **Write the MIPS code to <u>define</u> the following function**
  int function2(int g, int h)
  { return g + function1(g, h); }

(You *will* need to store something on the stack – why?)

(5 pts) Exercise 2-37

- **Write the MIPS code to <u>define</u> the following function**
  int function3(int a, int b)
  { return function6(a) + function7(b); }

(You *will* need to store something on the stack – why?)

(10 pts) Exercise 2-38

- **Write the MIPS code to <u>define</u> the following function**
  int lemur(int a, int b)
  { return panda(a) + b; }

(5 pts) Exercise B-1

- **Show the truth table for NAND and NOR gates**

NAND

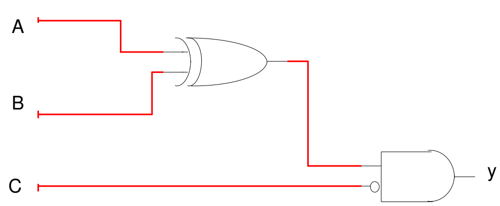
A	B	x
0	0	
0	1	
1	0	
1	1	

NOR

A	B	x
0	0	
0	1	
1	0	
1	1	

(5 pts) Exercise B-2

- **A.) Show the truth table for the following logic circuit**



A	B	C	y

- **B.) Write the Boolean equation for this circuit.**

(5 pts) Exercise B-3

- **Draw a circuit for the following formula:**
  $F = \overline{(\,(A + B) \bullet C\,) + D}$

(2 pts EXTRA CREDIT) Exercise B-4

- **Recall – how many entries are in a truth table for a function with n inputs?**
- **Consider – how many different truth tables are <u>possible</u> for a function with n inputs?**