(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 call 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 define function1.
(10 pts) Exercise 2-33

• Write MIPS code to define the following function:
  ```c
  int cat(int a, int b) {
    if (a < b)
      return a;
    else
      return b;
  }
  ```
Exercise 2-36

- Write the MIPS code to define the following function
  
  ```c
  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 define the following function

  ```
  int function3(int a, int b)
  {
    return function6(a) + function7(b);
  }
  ```

(You will need to store something on the stack – why?)
• Write the MIPS code to define 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

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>x</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>x</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(5 pts) Exercise B-2

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

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>y</th>
</tr>
</thead>
</table>

- B.) Write the Boolean equation for this circuit.
(5 pts) Exercise B-3

- Draw a circuit for the following formula:
  \[ F = ( (A + B) \cdot C ) + D \]

(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 possible for a function with \( n \) inputs?