Write your name and UNI on your solutions

Show your work for each problem; we are more interested in how you get the answer than whether you get the right answer.
1. (20 pts.) Imagine that the MIPS ISA were modified to disallow non-zero offsets in address calculations. All loads or stores of the form

\[ \text{lw } $x, a($y) \]

would need to become a two-instruction sequence such as:

\[ \text{addi } $y, $y, a \]
\[ \text{lw } $x, $y \]

(a) Show the modifications to the datapath and control to support this change. You should start from the implementation shown below and be sure not to change the behavior of any instructions other than the lw and sw.

(b) Assuming that SPECINT2000 consists of 25% loads and 10% stores, of which 40% and 50% respectively have non-zero offsets, what is percent change in SPECINT2000 execution time? You may assume no change in clock period as a result of your changes.
<table>
<thead>
<tr>
<th>Inst.</th>
<th>OP</th>
<th>RegWrite</th>
<th>RegDst</th>
<th>ALUSrc</th>
<th>Branch</th>
<th>MemWrite</th>
<th>MemToReg</th>
<th>ALUOp</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-type</td>
<td>000000</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1-</td>
</tr>
<tr>
<td>lw</td>
<td>100011</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>sw</td>
<td>101011</td>
<td>0</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-</td>
<td>00</td>
</tr>
<tr>
<td>beq</td>
<td>000100</td>
<td>0</td>
<td>-</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>-</td>
<td>01</td>
</tr>
</tbody>
</table>
2. (15 points) Assuming the following control signals are (individually) stuck at 0, meaning they always have the value 0. For each, give an example of an instruction that would malfunction as a result.

(a) \textit{MemToReg}

(b) \textit{ALUOp}_{1:0}

(c) \textit{PCSrc}
3. (20 points) The table below describes the performance of two processors, A and B, and two compilers, C and D, on a standard benchmark program. Which is the best compiler-machine combination? Provide quantitative support for your choice.

<table>
<thead>
<tr>
<th>Processors</th>
<th>GHz</th>
<th>Compiler C Instructions</th>
<th>Avg. CPI</th>
<th>Compiler D Instructions</th>
<th>Avg. CPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor A</td>
<td>3.4</td>
<td>7000</td>
<td>1.2</td>
<td>5000</td>
<td>1.5</td>
</tr>
<tr>
<td>Processor B</td>
<td>2.6</td>
<td>1500</td>
<td>2.2</td>
<td>1000</td>
<td>4.0</td>
</tr>
</tbody>
</table>
4. (20 points) Assuming the single-cycle processor examined in class, and the two sets of latencies shown below (processors A and B), indicate the critical path for each instruction listed below. All latencies provided are in units of picoseconds.

<table>
<thead>
<tr>
<th></th>
<th>I-Mem</th>
<th>Adder</th>
<th>Mux</th>
<th>ALU</th>
<th>Regfile</th>
<th>D-Mem</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor A</td>
<td>400</td>
<td>100</td>
<td>30</td>
<td>120</td>
<td>200</td>
<td>350</td>
<td>100</td>
</tr>
<tr>
<td>Processor B</td>
<td>500</td>
<td>150</td>
<td>100</td>
<td>180</td>
<td>220</td>
<td>1000</td>
<td>65</td>
</tr>
</tbody>
</table>

(a) An and instruction on processor A.

(b) A beq instruction on processor A.

(c) A sw instruction on processor B.

(d) A lw instruction on processor B.
5. (25 points) Consider the following MIPS program:

```mips
addi $s0, $0, 0  # i = 0
add $s1, $0, $0   # sum = 0
addi $t0, $0, 10  # $t0 - 10
loop:
    slt $t1, $s0, $t0  # if (i < 10), $t1 == 1, else $t1 = 0
    beq $t1, $0, done  # if (i >= 10), branch to done
    add $s1, $s1, $s0  # sum += i
    addi $s0, $s0, 1   # i++
    j loop
done:
```

(a) How many cycles does it take to execute the following program on the multicycle MIPS processor?

(b) What is the CPI of this program?