Assignment #2

1. Suppose we are considering a change to an instruction set. The base machine initially has only loads and stores to memory, and all operations work on the registers. Measurements of the load-store machine showing the instruction mix and clock cycle counts per instruction are given in the table below:

<table>
<thead>
<tr>
<th>Instruction Type</th>
<th>Frequency</th>
<th>Clock Cycle Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU Operations</td>
<td>43%</td>
<td>1</td>
</tr>
<tr>
<td>Loads</td>
<td>21%</td>
<td>2</td>
</tr>
<tr>
<td>Stores</td>
<td>12%</td>
<td>2</td>
</tr>
<tr>
<td>Branches</td>
<td>24%</td>
<td>2</td>
</tr>
</tbody>
</table>

Assume that 25% of the ALU Operations directly use a loaded operand that is not used again.

We propose adding ALU instructions that have one source operand in memory. These new register-memory instructions have a clock cycle count of 2. Suppose that the extended instruction set increases the clock cycle count for branches by 1 (so branches now take 3 cycles total), but it does not affect the clock cycle time. Would this change improve CPU performance? Justify your answer by showing your work.

3. Consider the following code fragment:

```assembly
loop: LD R1, 0(r2)
      ADDI R1, R1, #1
      ST 0(R2), R1
      ADDI R2, R2, #4
      SUB  R4, R3, R2
      BNEZ R4, loop
```

a) Show the timing of this instruction sequence for a processor that uses the 5-stage RISC-V pipeline, but without any data forwarding and without a delayed branch slot. Determine the number of cycles required to execute this loop.

b) Show the timing of this instruction sequence with all possible forwarding and bypassing hardware, and with a single-cycle delayed branch slot. Determine the number of cycles required to execute this loop.
NEW INSTRUCTION CAN REPLACE 1/4 OF ALU OPS OR 13.25% OF TOTAL. THIS WILL ALSO REDUCE LOADS BY 13.25%. OLD/NEW DATA:

<table>
<thead>
<tr>
<th></th>
<th>OLD</th>
<th>FREQ</th>
<th>LOAD</th>
<th>NEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU OPS</td>
<td>43</td>
<td>1</td>
<td>29.75</td>
<td>1</td>
</tr>
<tr>
<td>LOADS</td>
<td>21</td>
<td>2</td>
<td>7.75</td>
<td>2</td>
</tr>
<tr>
<td>STORES</td>
<td>12</td>
<td>2</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>BRANCHES</td>
<td>24</td>
<td>2</td>
<td>24</td>
<td>3</td>
</tr>
<tr>
<td>NEW INST</td>
<td>-</td>
<td>-</td>
<td>13.25</td>
<td>2</td>
</tr>
</tbody>
</table>

**OLD CPI = \( \frac{.43 \times 1 + .21 \times 2 + .12 \times 2 + .24 \times 2}{\text{ALU}} + \text{LD} + \text{ST} + \text{BR} \) = \( .43 + .42 + .24 + .48 = 1.57 \)**

**NEW CPI = \( .2975 \times 1 + .0775 \times 2 + .12 \times 2 + .24 \times 3 + .1325 \times 2 \) = \( .2975 + .155 + .24 + .72 + 1.2650 = 1.6775 \)**

CPI OF NEW MODIFICATION IS HIGHER, BUT # OF INSTRUCTIONS IS LOWER BY 13.25%, SO.

**EFFECTIVE CPI = 1.6775 \times 96.75\% = 1.655**

NEW MACHINE HAS HIGHER PERFORMANCE
Loop Count = 7

LD    0(reg)  R1
BEQZ ah, Loop
SUB  R5, R5, R6
ADDI R6, R6, 1
ADDI R1, R1, 1
LD R1, O(reg)

1 2 3 4 5 6 7 8
RESCHEDULE Beqz Delay

This assumes Wb/Td in same cycle
Loop Count = 16

LD    Bnez ah, Loop
SUB  R5, R5, R6
ADDI R6, R6, 1
ADDI R1, R1, 1
LD R1, O(reg)

0 1 2 3 4 5 6 7 8 9 10 11 R1 R2 R3 R4 R5 R6
No Forwarding

2