1. Sometimes software optimization can dramatically improve the performance of a computer system. Assume that a CPU can perform a multiplication operation in 15 ns, and an addition operation in 2 ns.
   (a) (5%) How long will it take for the CPU to calculate the result of $d = axb + axc$?
   (b) (10%) Could you optimize the equation so that it will take less time?

2. (15%) Convert the following MIPS codes to C codes. Assume that set_array is the first function called, and $s0, s0, and s1$ correspond to "i", "a", and "b" in C codes, respectively.

```c
set_array:
    addi $sp, $sp, -52
    sw $fp, 48($sp)
    sw $ra, 44($sp)
    sw $a0, 40($sp)
    addi $fp, $sp, 48
    add $s0, $zero, $zero
    addi $t0, $zero, 10
loop:
    sll $t1, $s0, 2
    add $t2, $sp, $t1
    add $a0, $a0, $zero
    add $s1, $s0, $zero
    jal compare
    sw $v0, 0($t2)
    addi $s0, $s0, 1
    bne $s0, $t0, loop
    lw $a0, 40($sp)
    lw $ra, 44($sp)
    lw $fp, 48($sp)
    addi $sp, $sp, 52
    jr $ra
compare:
    addi $sp, $sp, -8
    sw $fp, 4($sp)
    sw $ra, 0($sp)
    addi $fp, $sp, 4
```
3. (10%) Suppose that all of conditional branch instructions except beq and bne were removed from
MIPS instruction set along with slt and all of its variants. Show how to perform

\[
\text{slt } $t0, $s0, $s1
\]

by using the modified pseudo instructions in which slt is not available.

4. (10%) Table 1 shows the number of floating-point operations executed in three different programs
and the runtime for those programs on three different computers:

<table>
<thead>
<tr>
<th>Program</th>
<th>Floating-point operations</th>
<th>Execution time in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Computer A</td>
</tr>
<tr>
<td>Program 1</td>
<td>10x10^9</td>
<td>1</td>
</tr>
<tr>
<td>Program 2</td>
<td>40x10^9</td>
<td>10</td>
</tr>
<tr>
<td>Program 3</td>
<td>80x10^9</td>
<td>100</td>
</tr>
</tbody>
</table>

Suppose that the total number of floating-point operations executed in the workload is equally divided
among the three programs. That is, program 1 runs 8 times for every time program 3 runs, and
program 2 runs twice for every time program 3 runs. Find which computer is fastest for this
workload and by what factor.

5. Assume the following energy consumption for activity in Instruction Memory, Registers, and Data
Memory. You can assume that the other components of the datapath spend a negligible amount of
energy.

<table>
<thead>
<tr>
<th>Activity</th>
<th>I-Mem</th>
<th>1 Register Read</th>
<th>Register Write</th>
<th>D-Mem Read</th>
<th>D-Mem Write</th>
</tr>
</thead>
</table>
(a) (5%) How much energy is spent to execute an add instruction in a single-cycle design and in the five-stage pipelined design?
(b) (5%) What is the worst-case MIPS Instruction in terms of energy consumption, and what is the energy spent to execute it?
(c) (5%) If energy reduction is very important, how would you change the pipelined design? What is the percentage reduction in the energy spent by a lw instruction after this change?

6. (a) (10%) This question covers your understanding of dependence detection between instructions. Using the code below, list all of the dependence types (RAW, WAR, WAW). List the dependences in the respective table (example INST-X to INST-Y) by writing in the instruction numbers involved with the dependence.

```
10: A = B + C;
11: C = A - B;
12: D = A + C;
13: A = B * C * D;
14: C = F / D;
15: F = A ^ G;
```

<table>
<thead>
<tr>
<th>RAW Dependence</th>
<th>WAR Dependence</th>
<th>WAW Dependence</th>
</tr>
</thead>
<tbody>
<tr>
<td>From Instr</td>
<td>To Instr</td>
<td>From Instr</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(b) (10%) Given three instructions, how many unique comparisons (between register sources and destinations) are necessary to find all of the RAW, WAR, and WAW dependences. Answer for the case of three instructions, and then derive a general equation for $N$ instructions. Assume that all instructions have one register destination and two register sources.
7. (15%) For the MIPS datapath shown below, several lines are marked with “X”. For each one:
- Describe in words the negative consequence of cutting this line relative to the working, unmodified processor.
- Please write a small program code that will fail.
- Please write a small program code that will still work.