Homework 1 (Solutions)

Solve the following problems:

1. Consider a machine which executes a program consisting of 50% floating point multiply, 20% floating point divide, and the remaining 30% are from other instructions.
   a. Management wants the machine to run 4 times faster. You can make the divide run at most 3 times faster and the multiply run at most 8 times faster. Can you meet management's goal by making only one improvement, and which one?
   Amdahl's Law states:
   Execution time after improvement = (Execution time affected by improvement)/(Amount of Improvement) + Execution time unaffected
   Assuming initially that the floating point multiply, floating point divide and the other instructions had the same CPI,
   Execution time after Improvement with Divide = (20)/3 + (50 + 30) = 86.67
   Execution time after Improvement with Multiply = (50)/8 + (20 + 30) = 66.67
   ~~The management's goal can be met by making the improvement with Multiply alone.~~
   The management's goal can't be met since none of the solution allows to reach an execution time of 25 (i.e. which corresponds to 4 times faster).

   b. If you make both the multiply and divide improvements, what is the speedup of the improved machine compared to the original machine?
   If we make both the improvements:
   Execution time after Improvement = (50)/8 + (20)/3 + (30) = 53.33
   The speedup relative to the original machine = (100)/(53.33) = 1.88

2. Consider the following assembly code:

```
r1 = 99
Loop:   r1 = r1 – 1
        branch r1 > 0, Loop
        halt
```

   a. During the execution of the above code, how many dynamic instructions are executed?
   The Loop instructions execute for a total of 99 times.
   The number of dynamic instructions is 200:
   - 1 for r1=99
   - 99x2 for the loop
   - 1 for halt

   b. Assuming a standard unicycle machine running at 100 KHz, how long will the above code take to complete?
   Execution Time = $200 \times 1/(100 \times 10^3)$ = 2 milliseconds

3. In the snippet of MIPS assembler code below, how many times is instruction memory accessed? How many times is data memory accessed? (Count only accesses to memory,

not registers.)

```
lw      $v1, 0($a0)
addi    $v0, $v0, 1
sw      $v1, 0($a1)
addi    $a0, $a0, 1
```

The instruction memory is accessed four times (as there are four instructions) and the data memory is accessed twice (once for the lw instruction and another time for the sw instruction)

4. Suppose that a new MIPS instruction, called bcp, was designed to copy a block of words from one address to another. Assume that this instruction requires that the starting address of the source block be in register $t1 and that the destination address be in $t2. The instruction also requires that the number of words to copy be in $t3 (which is > 0). Furthermore, assume that the values of these registers as well as register $t4 can be destroyed in executing this instruction (so that the registers can be used as temporaries to execute the instruction).

   a. Write the MIPS assembly code to implement a block copy without this instruction.

```
Loop:   lw      $t4, 0($t1)
        sw      $t4, 0($t2)
        addi    $t1, $t1, 4
        addi    $t2, $t2, 4
        subi    $t3, $t3, 1
        bne     $t3, $zero, Loop
```

   b. Write the MIPS assembly code to implement a block copy with this instruction.

```
        li      $t1, src
        li      $t2, dst
        li      $t3, count
        bcp
```

   c. Estimate the total cycles necessary for each realization to copy 100-words on the multicycle machine.

      Assuming each instruction in the MIPS code takes 1 cycle, for doing a 100-word copy the total number of cycles taken is:
      Without bcp: 6 × 100 = 600 cycles
      With bcp: 4 × 100 = 400 cycles