1) (60 points)

Observe the following loop:

```assembly
ADD R5, R0, R0
Loop:  L.D F0, 0(R1)
       L.D F2, 0(R2)
       ADD.D F4, F0, F2
       L.D F6, 0(R3)
       MUL.D F8, F4, F6
       S.D F8, 0(R4)
       DADDUI R1, R1, #-8
       DADDUI R2, R2, #-8
       DADDUI R3, R3, #-8
       DADDUI R4, R4, #-8
       BNE R1, R5, Loop
```

Assume that ADD.D takes 4 EX cycles and MUL.D takes 5 EX cycles, all other instructions take 1 EX cycle. Also assume 5-stage MIPS pipeline, where branches are resolved in ID stage.

a) (10 points) How many cycles does one iteration of this loop take?

b) (15 points) Can you shuffle instructions and change offsets to make this loop run in fewer cycles? Show the rearranged loop and answer how many cycles does one iteration take now? Don’t forget to fill branch delay slot.

c) (30 points) Unroll the loop minimum number of times to get rid of the remaining stalls. Show the unrolled loop and answer how many cycles does one iteration take now?
2) (40 points)
Assume a processor with unified L1 and L2 cache, running at 2GHz. CPIideal is 1. Assume L1 access time of 0.5ns, L2 access time of 10ns and memory access time of 100ns. Bus between L1 and CPU operates at 1GHz and is 32-bit wide, bus between L1 and L2 operates at 400MHz and is 128-bit wide, and bus between L2 and memory operates at 200MHz and is 256-bit wide. CPU requests 2B words. L1 block size is 32B and L2 block size is 64B. L1 miss rate is 0.1% and L2 miss rate is 20%. Both L1 and L2 are write-back, write-allocate caches. Percentage of dirty blocks in L1 is 10% and in L2 is 30%. There is a write buffer that can eliminate L1 hit time for 95% of writes.

a) (15 points) Calculate average memory access time for reads

b) (15 points) Calculate average memory access time for writes

c) (10 points) Calculate average time per instruction, if instruction mix contains 10% of loads and 15% of stores.