Problems (50 points)

1. (**30 pts**) Consider the state of following procedure executing within the SPIM pipeline (Figure 4.51) during cycle 11. Assume that the first instruction is fetched in cycle 0! Assume full hardware support for forwarding, branches predicted as not taken, and flushing when branches are taken. The first instruction is at address 0x00400000. The data segment starts at 0x0100. Assume the data path supports the addi and bne instructions similar to add and beq respectively.

```
                .data
start:  .word 21, 22, 23,24
str:    .asciiz "CmpE"
        .align 4
        .word 24, 0x77

                .text
        .globl main

main:   addi $t3, $0, 0x4
        addi $t0, $0, 0x0100
        addi $t1, $0, 0x0200
move:   lw $t5, 0($t0)
        sw $t5, 0($t1)
        addi $t0, $t0, 4
        addi $t1, $t1, 4
        addi $t3, $t3, -1
end:    bne $t3, $zero, move

        addi $v0, $0, 0xA
        syscall
```

   a. Insert required nops to ensure this program executes correctly. Explain why it will not execute correctly.

   There must be two stall/nops between the lw and the sw. This is a hazard for the MEM stage (as opposed to a hazard in EX).

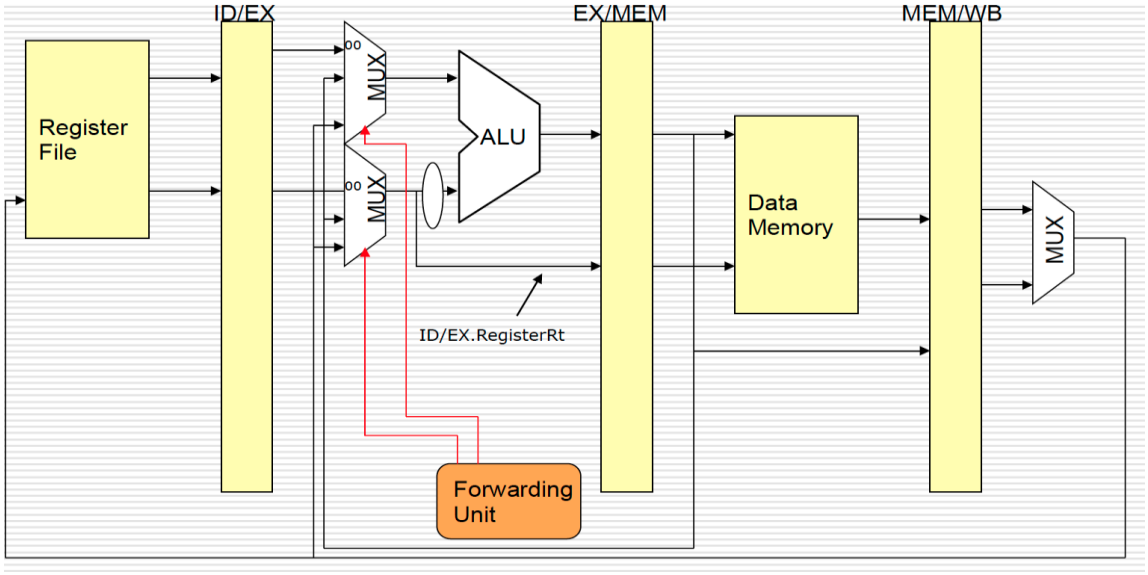
   b. What instructions are in each stage of the pipeline after the preceding correction is applied.

IF	sw $t5, 0($t1)
ID	lw $t5, 0($t0)
EX	bne $t3, $zero, move
MEM	addi $t3, $t3, -1
WB	addi $t1, $t1, 4

This is for the second time through the loop. Remember to count the nop the first time

through the loop. Note the stall cycle between the lw and sw does not show up until later in the pipeline – when sw advances to the at least the ID stage.

c. What are the values of the following fields? **Mux inputs are numbered from the top to bottom starting at 0**. MuxA is the first one from the top in EX. The figure below is just clarify notation for forwarding. Use Figure 4.51.



MEM/WB.MemToReg _____1_____

ForwardMuxA _____00_____

ForwardMuxB _____00_____

IF/ID.PC _____0x00400010_____

ID/EX.RegWrite _____0_____

EX/MEM.WriteData _____0x00000000___

EX/MEM.WriteRegister (Destination Address) _____0xB_____

d. Reschedule the code using nops to minimize/eliminate stall cycles.

```
lw $t5, 0($t0)
addi $t0, $t0, 4
sw $t5, 0($t1)
bne $t3, $zero, move
addi $t1, $t1, 4
addi $t3, $t3, -1
nop
```

Four nops need to be filled – one after the lw and three after the bne.

2. (**10 pts**) Add support for the jump instruction to the data path in Figure 4.51. To receive credit your solution should gave the following elements.
   a. Clearly show the hardware changes required of the datapath using Figure 4.51. If your drawing or handwriting is not legible you will not receive credit.
   b. Describe briefly how this would work.
      i. Operation in a few sentences
      ii. Changes to the controller and control signals.

   Add logic to compute the jump address in decode and feed it back to the PCSource multiplexor in IF. The mux control must now be expanded to two bits with the upper bit now being generated but the controller for the jump instruction. This will produce one stall cycle or need for one nop.

3. (**10 pts**) Extensive analysis of the instruction stream of code targeted for the pipelined MIPS with forwarding (Figure 4.51) has produced the following statistics. To gain some performance the EX and MEM stages have been partitioned into two stage pipelines. The overall pipeline is now 7 stages. 30% of load instructions produce a load-to-use hazard.

Instruction	Frequency
Loads	22%
Stores	13%
ALU Operations	42%
Branches	18%
Jumps	5%

   a. What is the penalty in cycles for i) data hazards, and ii) control hazards.

   Data hazards (load-to-use) now experience and additional cycle for a total of 2 cycles. Branches now have a penalty of 4 cycles (Figure 4.51). Jumps remain at 1 cycle.

b. 20% of these load delay slots can be filled via instruction scheduling. Similarly, we can use scheduling can fill 60% of the branch delay slots. Compute the CPI if the base (ideal) CPI is 1.0.

CPI = 1.0 + 0.22 * 0.3 * 0.8 * 2 + 0.18 * 0.4 * 4 + 0.05 * 1

Note: No late assignments will be accepted. You must make achieve a minimum average of 50% in the assignments to pass the course.