UNIVERSITY OF CALIFORNIA, DAVIS

Department of Electrical and Computer Engineering
EEC70 Assembly Language and Computer Organization Winter 2004

Homework Assignment #3
Due on Tuesday March 16th at 5:00pm

Name:

1. Describe the benefit of pipelining for DLX.

2. Should any changes be made to the following DLX code with forwarding to avoid stalls?

If so make them.

```
subi    r10,r10,1
addi    r11, r0, 5
add     r3, r4, r5
add     r6, r3, r5
```

3. Should any changes be made to the previous DLX code without forwarding to avoid stalls?

If so make them

Use the following assumptions about a DLX machine to answer questions 4-6

IF	takes 1 cycle
ID	takes 1 cycle
EX	takes 1 cycle
MEM	takes 1 cycle
WB	takes 1 cycle

4. What is the difference in number of cycles to execute 100 non dependent add instructions between a pipelined machine and a machine without pipelining?

5. What would that difference be if the Instruction and Data Cache were shared?

6. What is the difference in number of cycles to execute 10 dependent load instructions between a pipelined machine and a machine without pipelining?

7. What is the time to execute 100 instructions on the following pipelined machine?

IF	takes 1 cycle
ID	takes 1 cycle
EX	takes 2 cycles
MEM	takes 1 cycle
WB	takes 1 cycle

8. Describe each of the conflicts / hazards in DLX and write a portion of code which exhibits these conflicts.

9. Write code that sorts three numbers with as few RAW stalls as possible.

Assume the data is in r2, r3, r4

10. Describe the branch / jump penalty in DLX. Is there any way to avoid it? Explain

11. How many cycles would we stall in the following code if it took an entire cycle to write to the register file (without forwarding)?

```
        add     r2, r3, r4
        add     r5, r2, r4
```

12. Describe the Load penalty in DLX. Is there any way to avoid it? Explain