NAME: Solutions

STUDENT NUMBER:

EE557--FALL 2000

MIDTERM 2

Open books and notes
Time limit: 1hour and 20 minutes MAX. No extension.

Q1: /12
Q2: /8
Q3: /9
Q4: /8
Q5: /8
Q6: /5

TOTAL: /50

Grade: /25
QUESTION 1 (Tomasulo) 12 points

Consider the following loop.

```plaintext
LOOP    LD F2,0(R1)
        MULTD F4,F2,F0
        LD F6,0(R2)
        ADDD F6,F4,F6
        SD 0(R2), F6
        ADDI R1,R1,#8
        ADDI R2,R2,#8
        SGTI R3,R1,done
        BEQZ R3, LOOP
```

This is the same loop as the one in problem 5 of homework 3. As you did in this problem, fill the following table for Tomasulo algorithm, **assuming now that up to two instructions can be issued in any cycle**.

Show the schedule by filling the instruction status table. Each entry in the table shows the clock cycle when a given instruction went through a specific phase of execution. Show the content of the table up to the point when the sgti instruction write its results.

Assume that functional units are not pipelined, but that there are enough functional units to avoid any stalls due to structural hazards on functional units or reservation stations.

Use the latencies shown in Figure 4.63. These latencies mean that the execution time of a FPMPY is 7 clocks, that the execution time of FPADD is 5 clocks and that any integer instruction executes in one clock.

**Table 1: Instruction status table for question 1**

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Issue</th>
<th>Exec. complete</th>
<th>Write Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD F2,0(R1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MULTD F4,F2,F0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LD F6,0(R2)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDD F6,F4,F6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SD 0(R2), F6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDI R1,R1,#8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDI R2,R2,#8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SGTI R3,R1,done</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
QUESTION 2(software pipeline) 8points

Apply software pipelining to the loop in question 1 so that no true data dependencies on floating point operands remain in the body of the loop. Show the new loop below (you may ignore the startup and cleanup code).

LOOP    LD F2,0(R1)
          MULTD F4,F2,F0
          LD F6,0(R2)
          ADDD F6,F4,F6
          SD 0(R2), F6
          ADDI R1,R1,#8
          ADDI R2,R2,#8
          SGTI R3,R1,done
          BEQZ R3, LOOP
QUESTION 3. 9 points

1. Prove that if for two array elements $A[a \times i + b]$ and $A[c \times i + d]$ there is a true dependence then $\text{GCD}(c,a)$ divides $(d-b)$

2. In the following loop

   ```
   for (i=1, i<=100, i++)
   ```

   Apply the GCD test to detect loop-carried true dependencies (if any).

3. In the following loop

   ```
   for (i=1, i<=100, i++)
   ```

   Apply the GCD test to detect loop-carried true dependencies (if any).
QUESTION 4. (8points)

Consider the following loop:

\[
\text{for (i=1, i<=100, i++)} \\
\]

The compiled code looks as follows:

```markdown
LOOP: ] LD F2, 0(R1) /load A[i-3]
       LD F4, 24(R1) /load A[i]
       ADDD F4,F2,F4
       SD 24(R1), F4 /store A[i]
       ADDI R1,R1,8
       STGI R3,R1,R2 /R2 contains the maximum value for R1;
       BEQZ R3, LOOP /R3 is set to 1 if R1>R2
```

Please unroll the loop as much as possible, remove all useless instructions, rename registers, and then schedule the code to maximize ILP.

Answer:

Step 1. Unroll and remove useless instructions.

Step 2: Rename:

Step 3: schedule:
QUESTION 5. 8 points

A. An (m,n) branch prediction scheme uses m bits of global history and n bit history counters. Consider the following nested loop:

Consider the following loop.

```
LOOP1    I1
         I2
LOOP2    I3
         I4
         I5
         BEQZ R3, LOOP2
         BEQZ R4, LOOP1
```

When this loop is executed for the first time, all history counters are zero. The values of R3 and R4 are such that LOOP1 is executed 10 times and, in each iteration of LOOP1, LOOP2 is executed 100 times. Give the success rate for the following prediction scheme (the success rate is the number of good predictions divided by the number of branch instructions executed)
1. (0,1)
2. (2,2)

B. Explain what is meant by “branch folding”, how it is implemented in practice and what its effects are on performance.
QUESTION 6. 5 points

1 Consider the following loop.

```
LOOP     SUBI R1,R1,#8
         BEQZ R1, L1
         LD F2, 0(R1)
         J EXIT
L1:      LD F2, 1000(R1)
EXIT     ADDD F2, F2,F4
         SD 1000(R1),F2
         BNEZ R1, LOOP
```

Eliminate the BEQZ instruction and the J instruction by using the predicated load instruction, LDC Fi, 0(Rj), Rk (the load occurs unless Rk=0) and then write the new code below.