CSCE 513 Computer Architecture **Test 1** **September 29, 2009**

Name:_____ Email:_____

Instructions

. No Calculators!!

- Make sure your exam is complete.
- No Calculators, cell phones, or other electronic devices.
- All questions are equally weighted.
- Answer in the space provided if at all possible.
- If a question is unclear please ask early in the test.
- Good Luck!

1. Suppose the percentage of time spent on floating point operations is 30% and suppose that an improvement to the Floating point unit improves the performance of the floating point unit by a factor of 2. What is the overall speedup?

2. Assuming the classical 5-stage pipeline with no forwarding except that in the second half of the WB, the value written to a register may be read. Given the code below:

```
loop:  LD     R6, 0(R1)
       ADD    R5, R6, R6
       ADD    R10, R6, R5
       ADD    R8, R8, R6
       DADDIU R1, R1, -8
       BNE    R1, R2, loop
```

   (a) show how the first iteration of the loop would proceed through the pipeline.

   (b) Identify a stall that could be improved by forwarding and explain in detail how the stall could be eliminated.

   (c) Explain how a branch delay slot improves performance in general and explain for this loop an instruction that could be moved to the delay slot.

3. (a) For a RISC machine where all instructions are 1 word, if the branch history table has 1024 entries explain how the address of a branch instruction is mapped to its entry in the branch history table.

(b) Draw the diagram of a 3-bit saturating counter branch predictor.

(c) What is meant by a (2,3) correlating branch predictor.

(d) what does a tournament branch predictor add?

4. For each of the three types of data hazards give the name and write a sample section of code that illustrates it.

   (a) .

   (b) .

   (c) .

   (d) Which of these can be eliminated by "register renaming" if there are sufficient registers?

   (e) What distinguishes a superscalar machine?

5. Given the code below

```
loop:  LD.D    F0, 0(R1)
       MUL.D   F4, F0, F0
       ADD.D   F6, F6, F0
       ADD.D   F8, F8, F4
       DADDIU  R1, R1, -8
       BNE     R1, R2, loop

Assuming latencies
Integer operations, branches, loads 1
ADD.D  2
MUL.D  4
```

Show the result of 5 steps of Tomasulo's algorithm

6. (a) What are the main improvements provided by the reorder buffers that are added to Tomasulo's?

(b) What is the additional "state/stage" of execution of an instructions when using a reorder buffer.

(c) what is the main difference in the bus in Tomasulo's and a traditional bus?