

CS 573
Final Exam
May 14, 2004

The following exam is open book and open notes. You may feel free to use whatever additional reference material you wish, but **no electronic aids** are allowed. Please note the following instructions. There will be a ten point deduction for failure to comply with them:

- start each problem on a new sheet of paper
- write your social security number, but not your name, on each sheet of paper you turn in
- show your work whenever appropriate. There can be no partial credit unless I see how answers were arrived
- be succinct. You may lose points for facts that, while true, are not relevant to the question at hand

You have until 12:30 to finish the exam. The questions are equally weighted.

1. Several years ago, reading Usenet, I came across the following claim:

```
Subject: Re: Is RISC dead? (was: Re: K7's FP performance inches past P-III's)
Date: 1999/05/19
Author: Piercarlo Grandi <piercarl@Dial.PIPEx.com>
Well, my take here (and I said so in "comp.arch" quite a while ago) is that _architecture_ is
dead: with essentially infinite silicon/design budgets virtually any architecture, including
x86, can be made to perform; in other words only implementation matters.
```

Evaluate this claim. Is it the case that, today, the instruction set has become unimportant? In answering the question, consider the instruction set as the interface between the programmer's intent and the out-of-order execution engine executing it.

2. Consider the following assembly language code. For three-operand instructions, the result is the first operand.

```
add r0, r1, r2
sub r4, r1, r2
add r3, r0, r4
add r0, r4, r3
```

- (a) Show all of the dependencies and antidependencies in this code.
- (b) If you have an infinite number of decoders, functional units, etc. and
 - i. Fetching an infinite number of instructions from memory takes one cycle
 - ii. Decoding them all takes one cycle
 - iii. Executing an instruction, and making its result available to another instruction requiring it, takes one cycle

Draw a Gantt (timing) diagram showing how many cycles the code will take to execute.

3. Traditionally, a computer's virtual memory architecture is fixed, and translation lookaside buffer (TLB) management is in hardware. So, for instance, Intel documents their two-level page table structure, and the presence of the TLB is all but invisible to the implementer of the operating system.

A recent development has been management of the TLB in software. In this model, there is no fixed page table structure; instead, any time a TLB miss occurs there is a software interrupt to update the TLB entries, and the structure of the page tables (whether they be multi-level forward page tables, inverted page tables, hashed page tables, or some other structure) is handled completely by the operating system.

Discuss the strengths and weaknesses of the two approaches.

4. How many 4×4 switch boxes would be required to implement a 64×64 dynamic hypercube network?

5. Consider the following parallel processes P1, P2, and P3 executing on three processors:

P1

$x = 1;$
 $y = 3;$
 $z = x - y;$

P2

$y = 4;$
 $x = 2;$
 $z = x + y;$

P3

$x = 3;$
 $y = 7;$
 $z = y - x;$

(x , y , and z are all shared variables)

- (a) Add one or more synchronization operations S to all three processes which will guarantee that the final value of variable z is 5.
- (b) Using the memory read/write/sync notation we've been using in this class (for instance, in the solution to HW3 question 3), show the operations which will be performed.

6. Error Correcting Codes

- (a) Suppose you have an eight bit byte using a SECDED Hamming correcting code, as follows:

M8	M7	M6	M5	C8	M4	M3	M2	C4	M1	C2	C1	P
0	1	0	1	0	1	0	1	1	1	0	1	1

This may have no errors, a one-bit error, or a two-bit error. Which is it? If it has a one-bit error, correct it.

- (b) Suppose a value using a SECDED Hamming code has a three bit error. When the receiver tries to read this value:
 - i. How many bits will it "think" are in error?
 - ii. What will happen when it tries to correct it?

7. Consider the code from Question 2. Suppose you were to translate this code for execution in a dataflow computer.

- (a) Draw a dataflow diagram for the code.
- (b) Assume the dataflow computer's arithmetic instructions each have an address, and have the following fields:
 - i. The operation to be performed (+ and - in this case)
 - ii. The address of the instruction to which the result should be sent, and whether the result should be the instruction's first or second operand.

What are the dataflow computer's instructions? Assume the instruction addresses are 1, 2, 3, and 4.