GEORGIA INSTITUTE OF TECHNOLOGY

College of Computing

CS6290/CS4290 — High-Performance Computer Architecture

Spring 2003

CS6290/CS4290	Issued: February 14, 2003
Homework 3	Due: February 21, 2003

Purpose: This homework reviews pipeline operation and instruction-level parallelism.

Problems:
1. Pipeline Operation.
2. Instruction Scheduling.
3. Write a paragraph about [Smith88].

Reading: H&P Appendix A; Section 4.1.
[Smith88]

Problem 1: Pipeline Operation

A: Problem A.1 in the book (from Appendix A). Note that the question describes three different branching strategies. Part A refers to the original DLX pipeline (Figure A.18) which has a three-cycle branch penalty when you "flush" the pipeline. Part B says to use predict-not-taken which means you suffer the three-cycle penalty only if the branch is taken. Part C says to use the standard DLX single delay slot scheme (circuit in Figure A.24, pipeline diagram in Figure A.13).

Problem 2: Instruction Scheduling

In each problem, show the stall cycles incurred in the original code before you reschedule the loop. I find it really handy to do this sort of work using a text editor instead of paper because I can easily move instructions up and down, etc.

A: Problem 4.8 in the book.

B: Problem 4.9, parts a, b and c only.

Problem 3: Precise Exceptions

[Smith88] J. E. Smith and A. R. Pleszkun, "Implementing Precise Interrupts in Pipelined Processors", *IEEE Transactions on Computers*, 37(5), pages 562-573, May, 1988.

The [Smith88] paper catalogs several ways to achieve precise exceptions, including "reorder buffers", "history buffers" and a "future file". When the ISCA conference was in Atlanta in 2000, Prof. Smith was awarded the Eckert-Mauchly Award* by the computer architecture community in part for his early work on precise exception handling (this paper). One or another of the ideas here appears in every superscalar design. Amusingly, in his talk, Smith displayed part of a four-page negative review he received for this paper which explained in lengthly detail why the problem should be considered *irrelevant* based on alternate solutions existing at the time.

A: What are the principle advantages of a "future file" over a "history buffer".

* Eckert and Mauchly developed the ENIAC and then started a company to develop the first commercial computer, UNIVAC. The Eckert-Mauchly award has been presented yearly since 1979. See the whole list at `http://computer.org/awards/awdem.htm`