

GEORGIA INSTITUTE OF TECHNOLOGY

College of Computing

CS6290/CS4290 — High-Performance Computer Architecture

Spring 2003

CS6290/CS4290
Homework 1

Issued: January 6, 2003
Due: January 17, 2003

Purpose: This homework covers Amdahl's law, instruction sets and some VLSI issues.

Reading: H&P Chapters 1, 2 for background.
[Patterson85] to be discussed in class Friday.

Problems:

1. Amdahl's law.
2. Instruction sets.
3. Reading: [Patterson85].

Collaboration: (*As in the syllabus*) collaboration on projects and homework in **pairs** is encouraged. If you work in a pair, turn in one write-up with the names of both collaborators. You're welcome to discuss high-level concepts with other groups, but all homework solutions must be worked out and written up separately.

Problem 1: Amdahl's Law

A: Problem 1.2 in the book. Doublecheck that you have the third edition at this point – the problems are different from previous editions.

B: Your computer vendor offers three optional enhancements for the same price. The speedup of each enhancement and its applicability (in pre-enhanced time) is given below. If you can only buy one optional enhancement, which one should you buy? Show the post-enhanced time for each enhancement.

	Speedup	Usability
A:	3×	75% of pre-enhanced time
B:	10×	60% of pre-enhanced time
C:	100×	50% of pre-enhanced time

Problem 2: Instruction Sets

A: Problem 2.1 in the book. Note that all the variables are initially in memory and the results need to be stored in memory.

B: Problem 2.7 in the book. Tech's library is a goldmine for this one – try just browsing the shelves near QA76.9 on the fourth floor east. The web is also a good source. Good examples if you can find them are PDP-11, PDP-10, IBM 360, IBM 701, B6700, B5000. Find the year of manufacture, also – fame and glory to the one who finds the details for the oldest machine!

Problem 3: Reading

Read [Patterson85] in order to be prepared to discuss it on Friday, January 17th. Write a short paragraph (4-5 sentences) (i.e. short!) summarizing the arguments for *using* microcode. Why was everyone else arguing that microcode was a *good* idea when the various RISC projects were arguing the opposite?