

CS 4290/6290: High-Performance Computer Architecture

Spring 2004

Midterm Quiz

This is a closed-book exam. There are 6 problems on this quiz and the maximum total number of points is 40. Write your answers in the space provided. Use the extra pages for scratch space is needed.

Problem 1. [3 points] Amdahl's Law

For each of the following statements, indicate whether it is true or false:

- A) A speedup of 40 on 40% of the program will result in an overall speedup of at least 2.
- B) A speedup of 20 on 50% of the program will result in an overall speedup of at least 2.
- C) A speedup of 10 on 60% of the program will result in an overall speedup of at least 2.

Problem 2. [3 points] Processor Performance Equation

For each of the following statements, indicate whether it is true or false:

- A) Program execution time increases when the clock rate increases
- B) Program execution time increases when the CPI increases
- C) Program execution time increases when the instruction count (IC) increases

Problem 3. [4 points] Register Renaming

For each of the following statements, indicate whether it is true or false:

- A) Register renaming eliminates stalls due to name dependences through memory
- B) Register renaming eliminates stalls due to output (WAW) dependences on registers
- C) Register renaming eliminates stalls due to anti (WAR) dependences on registers
- D) Register renaming eliminates stalls due to flow (RAW) dependences on registers

Problem 4. [3 points] Pipelining

For each of the following statements, indicate whether it is true or false:

- A) Splitting the shortest stage of a five-stage pipeline will result in a higher clock rate.
- B) With single-issue, in-order execution, and the classical five-stage pipeline with no bypassing, WAW hazards never cause any "bubbles" (stalls) in the pipeline.
- C) With a single-issue, in-order execution, and the classical five-stage pipeline with bypassing, RAW hazards never cause any "bubbles" (stalls) in the pipeline.

Problem 5. [20 points] Tomasulo’s Algorithm

A processor uses Tomasulo’s algorithm in its floating-point unit. The processor has one CDB, but can issue two instructions per cycle. The figure shows the current state of the processor, at the very end of a clock cycle. The adder is not doing anything, but the multiplier has just finished an operation. A new cycle begins now.

Instruction	Instruction Status		
	Issue	Execute	Wr. Result
MUL F0, F1, F2	Yes	Yes	
ADD F1, F0, F3	Yes		
ADD F2, F3, F2	Yes		
ADD F3, F1, F4			
ADD F0, F1, F2			

Register	Status	
	V _i	Q _i
F0		Mul1
F1		Add1
F2		Add2
F3	20	
F4	43	
F5	12	
F6	3.75	
F7	0.01	

Reservation Stations						
Name	Busy	Op	V _i	V _k	Q _i	Q _k
Add1	Yes	ADD		20	Mul1	
Add2	Yes	ADD	20	75		
Add3	No					
Mul1	Yes	MUL	125	75		
Mul2	No					

A) [8 points] What will happen in this new cycle when the processor tries to issue the next two instructions? Update the figure to reflect the resulting state of the processor. If one or both of the two instructions can not issue in this cycle, briefly explain why it/they can not issue.

B) [5 points] The multiplier and the adder are both free to begin a new computation in this cycle. Which instructions can begin to execute in the adder and which in the multiplier? Update the figure to reflect the resulting state after beginning execution for these instructions.

C) [7 points] What is going to be broadcast on the CDB in this cycle? Update the figure to reflect the state after the broadcast and the resulting actions are complete.

Problem 6. [7 points] Branch Prediction

A single-issue, statically-scheduled, deeply pipelined processor is executing a program. During the entire program run, the processor retires (commits) a total of 1,000,000 instructions. With a perfect BTB (which correctly predicts both the target and the direction every time), there would be no stalls in the pipeline and the program would complete in exactly one millisecond. However, the BTB is not perfect. Of the 1,000,000 dynamic instructions, 200,000 are branches. Only 170,000 of those are BTB hits (the BTB has an entry for them). Of the 170,000 BTB hits, 150,000 are correctly predicted, while for the other 20,000 the BTB incorrectly predicts the direction, target, or both. Of the 30,000 BTB misses (those branches that the BTB did not have an entry for), 10,000 are not taken and 20,000 are taken. The BTB is the processor's only means of predicting branches, there are no delay slots, and the processor makes no attempt to correct a mispredicted branch until the branch goes through the last stage of the pipeline. At the end of that last pipeline stage the correct direction and target are known and, if needed, in the next cycle the fetch restarts from the first instruction that should execute after the branch. With the imperfect BTB, the overall execution time of the program is 1.8 milliseconds. How many stages are there in the processor's pipeline? Show your work.