Faculty of Media Engineering and Technology
Dept. of Computer Science and Engineering
Assoc. prof. Milad Ghantous

CSEN 702: Microprocessors
Winter 2024
Quiz 2 version 2 (35 min)

Name	ID	Tutorial	Total /15

Memory hierarchy

CPU execution time = (CPU clock cycles + Memory stall cycles) x Clock cycle time

$$\begin{aligned}\text{Memory stall cycles} &= \text{Number of misses} \times \text{Miss penalty}\\ &= \text{IC} \times \frac{\text{Misses}}{\text{Instruction}} \times \text{Miss penalty}\\ &= \text{IC} \times \frac{\text{Memory accesses}}{\text{Instruction}} \times \text{Miss rate} \times \text{Miss penalty}\end{aligned}$$

$$\frac{\text{Misses}}{\text{Instruction}} = \frac{\text{Miss rate} \times \text{Memory accesses}}{\text{Instruction count}} = \text{Miss rate} \times \frac{\text{Memory accesses}}{\text{Instruction}}$$

Average memory access time = Hit time + Miss rate × Miss penalty

Consider two memory systems M1 and M2.

M1 has the following characteristics:

- A 800 KB direct-mapped cache with 80-bytes blocks.
- 50 misses per 1000 instructions
- 0.45 ns clock cycle
- Miss penalty is 2 cycles-per-block-byte. Hit time is 1 cycle.
- No branch prediction is used. Branch penalty is 1 cycle.

M2 has the following characteristics:

- A 800 KB direct-mapped cache with 40-bytes blocks.
- Miss rate is 6%
- 0.55 ns clock cycle
- Miss penalty is 25 ns, plus 10 extra cycles per miss.
- Hit time is 1 cycle.
- Static always-taken branch prediction is used. Penalty is 2 cycles.

Both systems are tested against the same kind of programs having 50% ALU operations, 10% branches, 30% loads and 10% stores. 70% of branches are found to be taken. Consider 1 KB to be 1000 bytes.

A) Compute the miss rate of M1. (2 pts)
Miss rate = (misses/instructions) / (memory access per instruction) = 0.05/1.4 = 0.035

B) Compare the average memory access time for both systems (in ns). (4 pts)

M1: hit time + miss rate x miss penalty = 0.45 + (0.035)x (160 x 0.45 ns) = 2.97 ns
M2: 0.55 + 0.06x(25 + 10x0.55) = 2.38 ns
M2 is better

C) Compare the 2 systems in terms of CPU execution time. (4 pts)

M1: IC x (CPI + control hazard + miss rate x miss penalty) x clock cycle
= IC x (1 + 0.1x1 + 0.035 x 160) x 0.45 = 3.015 IC

M2: IC x (1 + 0.1x0.3x2 + 0.06x45.45 + 0.06x10) x 0.55 = 2.41 IC

M2 is better

D) Now assume that we know our programs access the data in this pattern: Read block #2000, Write block #4000 and repeat. If M1 uses a **no-write allocate** strategy with a write buffer that handles the writes, while M2 uses the **write-allocate** strategy, which system would be better? Why? (5 pts)

M1 has 10000 blocks so address 1000 and 4000 map to different locations.
The first read will miss, then the write won't need to be brought to cache, so all next accesses to read are hits but all writes are misses.
M2 has 20000 blocks so address 2000 maps to 0 and 4000 map to different locations too.
First read will miss, then write will miss but brought to the cache, then read will miss and then write will hit all the way. M2 is better.