Cptr350 Homework #9 Winter, 2020

Direct-Mapped and Set-Associative Caches

DUE: Wednesday, February 26, Start of class

Objective

Gain an understanding of the issues involved in cache memory systems.

To Do

1. This question covers cache and pipeline performance analysis.
   A. Write the formula for the average memory access time assuming one level of cache memory.
   B. For a data cache with a 92% hit rate and a 2-cycle hit latency, calculate the average memory access latency. Assume that latency to memory and the cache miss penalty together is 124 cycles. Note: The cache must be accessed after memory returns the data.
   C. Calculate the performance of a processor taking into account stalls due to data cache and instruction cache misses. The data cache (for loads and stores) is the same as described in Part B and 30% of instructions are loads and stores. The instruction cache has a hit rate of 90% with a miss penalty of 50 cycles. Assume the base CPI using a perfect memory system is 1.0 Calculate the CPI of the pipeline, assuming everything else is working perfectly. Assume the load never stalls a dependent instruction and assume the processor must wait for stores to finish when they miss the cache. Finally, assume that instruction cache misses and data cache misses never occur at the same time.
   D. Calculate the additional CPI due to the icache stalls.
   E. Calculate the additional CPI due to the dcache stalls.
   F. Calculate the overall CPI for the machine.
2. This question covers general issues related to cache systems.
   A. What are the two characteristics of program memory accesses that caches exploit?
   B. What are three types of cache misses?
   C. Design a 128KB direct-mapped data cache that uses a 32-bit address and 16 bytes per block. Calculate the following:
      1. How many bits are used for the byte offset?
      2. How many bits are used for the set (index) field?
      3. How many bits are used for the tag?
   G. Design an 8-way set associative cache that has 16 blocks and 32 bytes per block. Assume a 32 bit address. Calculate the following:
      1. How many bits are used for the byte offset?
      2. How many bits are used for the set (index) field?
      3. How many bits are used for the tag?

To Turn In

- **This page** stapled to your solutions, which are to be done in accordance with the School of Engineering homework guidelines found on the course web page. Use minimal, but sufficient, problem statements. Show all work.