ECEn 528
Study Guide - Basic caching

- Read Sections B.1-B.3 of H&P
  - Things to focus on
    - You need to know all 36 terms on page B-2. They will all be defined in today's reading.
    - The four memory hierarchy questions
    - Equations for average memory access time and processor performance. The example on p. B-18 to B-19 is very good.
    - Only skim the "Miss penalty and out-of-order execution" section
    - Know the three types of misses from B.3
    - The sixth optimization in B.3 won't make as much sense until you read B.4

  - Clarifications
    - Beware: the number of sets is not the associativity; rather:

      $$(Number\ of\ blocks) = (Number\ of\ sets) * (Associativity)$$

    - Note that the number of sets and block size are almost always a power of 2, but the associativity and number of blocks need not be.
    - *Block address* equals (*address* MOD *blockSize*) where *address* is the actual address referred to by an instruction.
    - Remember that $(Capacity) = (Number\ of\ sets) * (Associativity) * (Block\ size)$

  - Answer the following questions:

1. In a 3-way associative cache with a 384Kbyte capacity and a block size of 32 bytes, in which set would the address 0xab876543 be stored? What would the tag field be for that address?

blocks = 384k/32 = 12k; # sets = 12k/3 = 4k -> # index bits = 12 -> set address = 0x543
tag width = 32 - 12 = 20 bits -> tag field = 0xab876

2. How many bits of storage would a direct-mapped cache with 64-byte blocks, a capacity of 32K, and 32-bit addresses use? How many bits of storage would it use if 64-bit addresses were used?

blocks = 32K / 64 = 512
sets = # blocks / 1 = 512 (9 bits)
status width = 1 bit

32-bit addresses:
tag width = 32 - 9 - 6 = 17 bits
bits = 512*1*(64*8+17+1)
= 271,360 bits

64-bit addresses:
tag width = 64 - 9 - 6 = 49 bits
bits = 512*1*(64*8+49+1)
= 287,744 bits

3. What advantages and disadvantages does a writeback cache have with respect to a writethrough cache?

Writeback: writes occur at the speed of the cache hierarchy, multiple writes within a block require only one write to the lower-level memory, less memory bandwidth is used, and less power is used

Write-through: easier to implement; cache is always clean, next lower level has the most current copy of data

4. What are the three kinds of misses?

Compulsory (cold-start/first-reference): the first access to a block can't be in cache

Capacity: not every block can fit in the cache, so some are thrown out as more appear

Conflict: the same block gets thrown out and retrieved repeatedly if too many blocks map to its set

5. For each cache optimization, indicate whether the following elements of cache performance improve or disimprove; for miss rate, indicate which kinds of misses are affected:

Optimization	***Miss Penalty***	***Hit Latency***	***Miss Rate***
Larger block size	Disimproves		Improves compulsory misses
Larger cache		Disimproves	Improves capacity misses
Higher associativity		Disimproves	Improves
Multilevel caches	Improves		Improves
Giving read misses priority over writes	Improves		
Avoiding address translation		Improves	