EEL 6764 Principles of Computer Architecture
Homework #2

1 Problems

B1 a. $1 + 0.03 \times 110 = 4.3$

b. Because of the randomness of the accesses, the probability an access will be a hit is equal to the size of the cache divided by the size of the array.

$$\text{Hit rate } = 64\ KB/1\ GB = 2^{-14} \approx= 6 \times 10^{-5}$$

Therefore

$$\text{average access time } = 1 + (1 - 6 \times 10^{-5}) \times 110 \approx 110$$

c. The access time when the cache is disabled is 105 cycles, which is less than the average access time when the cache is enabled and almost all the accesses are misses. If there is no locality at all in the data stream, then the cache memory will not only be useless, but it will also be a liability.

d. Let that missing rate be x. Suppose $1 + x \times 110 = 105$. Solving that equation leads to $x = 104/110$. When the miss rate exceeds 104/110, using cache leads to worse performance.

B2 a. Cache is direct-mapped.

Cache Block	**Memory blocks that can reside in cache block**
0	$M0, M8, \ldots, M24$
1	$M1, M9, \ldots, M25$
2	$M2, M10, \ldots, M26$
3	$M3, M11, \ldots, M27$
4	$M4, M12, \ldots, M28$
5	$M5, M13, \ldots, M29$
6	$M6, M14, \ldots, M30$
7	$M7, M15, \ldots, M31$
cache block	

b. Cache is 4-way set associative.

Cache Block	**Set**	**Memory blocks that can reside in cache block**
0	0	$M0, M2, \ldots, M30$
1	0	$M0, M2, \ldots, M30$
2	0	$M0, M2, \ldots, M30$
3	0	$M0, M2, \ldots, M30$
4	1	$M1, M3, \ldots, M31$
5	1	$M1, M3, \ldots, M31$
6	1	$M1, M3, \ldots, M31$
7	1	$M1, M3, \ldots, M31$

B5 It would be better to summarize all information given in the problem description. Note that nor all information is necessary.

- CPU: 1.1 GHz, or 0.909 ns equivalent, CPI of 1.35 excluding memory access.

- Instruction mix: 70% non-memory-access instruction, loads 20%, and stores 10% of all instructions.
- L1-L2 bus: 16B wide at 266 MHz, bus cycle time is $3.75ns$
- Caches: split L1 caches with no hit penalty.
  - I-cache: miss rate = 2%, 32B blocks, which requires 2 bus cycles to fill, miss penalty = 15ns + 2 L1-L2 bus cycles = $15 + 2 \times 3.75 = 22.5ns$.
  - D-cache: miss rate = 5%, write-through no write allocate, 95% of all writes do not stall because of a write buffer, 16B blocks requiring 1 bus cycle to fill, miss penalty $= 15ns + 1\ L1 - L2\ bus\ cycle = 18.75ns$.
- L2-Mem bus: 16B wide at 133Mhz, bus cycle is 7.5ns
- L2 unified cache, 512 KB, write-back with write-allocate, hit rate = 80%, 50% of replaced blocks are dirty (must go to the main memory), 64B blocks requiring 4 bus cycles to fill, miss penalty $= 60 + 4 * 7.5 = 90ns$. (four cycles for transferring 64B on L2-memory bus with 7.5 ns/per cycle, which is equivalent to one 133 MHz memory bus cycle). Its hit time is 15 ns.

a. The AMAT for instruction accesses is

$$\text{L1 hit time} + \text{L1 miss rate} \times \text{L1 miss penalty}$$

where

$$\text{L1 miss penalty} = \text{L2 hit time} + \text{L2 miss rate} \times \text{L2 miss penalty}.$$

- L2 hit time is 15 ns plus 2 cycles at 266 Mhz, which is $15 + 2 \times 3.75 = 22.5$ ns. (3.75 ns is equivalent to one 266 MHz L1-L2 bus cycle).
- L2 miss rate is 0.2. For each L2 miss, there is 50% chance that a block is written back to the main memory. Combining all the results above,

$$\text{AMAT for instruction access} = 0.02 \times (22.5 + 0.2 \times 1.5 \times 90) = 0.99ns$$

which is about 1.09 CPU cycles. Since L1 hit does not lead to penalty, it is ignored.

b. The average memory access time for data reads − Similar to the above formula with one difference: the data cache block size is 16 bytes which takes one L1-L2 bus cycle transfer (versus two for the inst. cache), so

- L1 (read) miss time in L2 $= 15 + 3.75 = 18.75ns$
- L2 miss time in memory: $90ns$
- AAMAT for read $= 0.05 \times (18.75 + (1 - 0.8) \times (90 + 0.5 \times 90)) = 2.29ns$

c. The average memory access time for data writes − Assume that writes misses are not allocated in L1, hence, all writes use the write buffer. Also, assume the write buffer is as wide as the L1 data cache.

- L1 (write) time to L2 $= 15 + 3.75 = 18.75ns$
- L2 miss time in memory $= 90ns$
- AMAT for data writes $= 0.05 \times (18.75 + (1 - 0.8) \times (90 + 0.5 * 90)) = 2.29ns$

d. The overall CPI, including memory accesses includes base CPI, Inst fetch CPI, read CPI or write CPI, inst fetch time is added to data read or write time (for load/store instructions).

$$\text{Overall CPI } = 1.35 + \frac{0.99}{0.909} + 0.2 \times \frac{2.29}{0.909} + 0.1 \times \frac{2.29}{0.909} = 3.1$$

B9 Assume DM cache with 2 blocks, and a 2-way associative cache with 1 set.

Construct a trace of memory addresses from CPU $A1, A2, A3, A1, A2, A3, A1, A2, A3, \ldots$, such that all the three addresses map to the same set in the two-way associative cache. Because of the LRU policy, every access will evict a block and the miss rate will be 100%.

If the addresses are set such that in the DM cache $A1$ maps to one block while $A2$ and $A3$ map to another block, then all $A1$ accesses will be hits, while all $A2/A3$ accesses will be all misses, yielding a 66% miss rate.

2.18 From 2.8(c), we know that misses per instruction are 0.022, 0.012, 0.0033, and 0.0009 for 1-way, 2-way, 4-way, and 8-way set associative caches, respectively. The average data references per instruction is 0.3. Miss penalty is 20 cycles for all models.

a. Consider the 4-way associative cache. Its hit time is 3 cycles. Its AMAT is

$$3 + 0.0033 \times 20 = 3.066 \text{ cycles}$$

For the same cache with way prediction, it is modeled as a direct mapped cache, so its miss rate remains 0.0033. The hit time has two components: hits with correct prediction, and hits with incorrect prediction. Since the prediction accuracy is 0.8, the hit time is

$$0.8 * 2 + 0.2 * 3 = 2.2 \text{ cycles}$$

Its AMAT is

$$2.2 + 0.0033 \times 20 = 2.266 \text{ cycles}$$

Therefore, cache with way-prediction is $\frac{3.066}{2.266} = 1.353$ times faster than the 4-way associative cache.

c. In D-cache case, if there is a mis-prediction, the penalty is 15 cycles. So, the overall hit time is

$$0.8 * 2 + 0.2 * 15 = 4.6 \text{ cycles}$$

And its AMAT is

$$4.6 + 0.0033 \times 20 = 4.666 \text{ cycles}$$

If we consider 4-way associative cache used in (a) for comparison, the performance of D-cache with way-prediction is 3.066/4.666 = 66% of 4-way associative cache without way-prediction.

2.20 a. Without the critical word first, the number of cycles required would be $120+3\times16 = 168$ cycles. Assume that the first 16B block from the memory contains the requested data. With the critical word first and early restart, the number of cycles required to service a L2 miss is 120 cycles.

b. To decide whether critical word first is more important to L1 or L2, its contribution to AMAT of L1 or L2 should be evaluated.

2.21 a. 16B, to match the L2 write data bus.

b. The stated assumption implies a perfect condition where 8B writes are issued without interruption. Assume that 1 cycle is needed to fill a buffer entry, and no overlap between two consecutive L2 writes from the L1 write buffer.

Without write merging, it takes 1 cycle to fill a buffer entry and 4 cycles to write 8B to L2. For 16B, it would take a total of 10 cycles. With write merging, it takes 2 cycles to fill buffer entry, and 4 cycles to write 16B to L2. So, the speed up is 10/6 = 1.67.

c. Since L1 is write-through, the write misses do not exist. Therefore, only read misses are considered. For blocking cache, a miss stalls the processor, and future writes are suspended until the miss is resolved. The misses do not change the required write buffer entries.

For non-blocking cache, writes can be processed from the write buffer during a miss. Even though the processor may still issue writes to L1 cache in the presence of a miss, the chances of some future writes depending on the current miss is higher. Therefore, the bandwidth of writes to the buffer could be less, which means that fewer entries may be needed.