Tutorial 5
Caches

Syed Asad Alam
### Document History

<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Comment</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>28-11-2020</td>
<td>Initial Draft</td>
<td>SAA</td>
</tr>
<tr>
<td>0.2</td>
<td>29-11-2020</td>
<td>Feedback from demonstrator</td>
<td>SAA</td>
</tr>
<tr>
<td>0.3</td>
<td>01-12-2020</td>
<td>New problems added</td>
<td>SAA</td>
</tr>
<tr>
<td>1.0</td>
<td>03-12-2020</td>
<td>Tutorial 5 released</td>
<td>SAA</td>
</tr>
</tbody>
</table>
1 Learning Outcomes

This lab satisfies the following learning outcomes of the course:

**LO7** Explain the use of a memory hierarchy to reduce effective memory access times, describe the organization and operation of a cache, evaluate the hit rate of a cache given an address trace, develop a C/C++ cache model and know how to apply address trace analysis optimizations

2 Questions

**Question 1:**
Compute the number of hits and misses if the following list of hexadecimal addresses is applied to caches with the following organizations.

1. 128 byte 1-way cache with 16 bytes per line (direct mapped)
2. 128 byte 2-way set associative cache with 16 bytes per line
3. 128 byte 4-way set associative cache with 16 bytes per line
4. 128 byte 8-way associative cache with 16 bytes per line (fully associative)

0 × 0000 → 0 × 0004 → 0 × 000d0 → 0 × 000e0 → 0 × 1130 → 0 × 0028 →
0 × 113c → 0 × 2204 → 0 × 0010 → 0 × 0004 → 0 × 0040 → 0 × 2208 → 0 × 0008 →
0 × 00d0 → 0 × 0004 → 0 × 1104 → 0 × 000c → 0 × 0084 → 0 × 000c → 0 × 3390 →
0 × 0060 → 0 × 1100 → 0 × 0028 → 0 × 0070 → 0 × 00d0 → 0 × 0008 → 0 × 3394 →

Assume that the first 4 bits of the address is used as the offset within the cache line, the next \( \log_2(N) \) bits select the set and the remaining bits form the tag. Furthermore, assume that the all cache lines are initially invalid and that a LRU replacement policy is used.

**Question 2:**
Assume an instruction cache miss rate for a program is 2% and a data cache miss rate is 4%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. The instructions which access memory, i.e., the load/store instructions, make up 36% of all instructions.

**Question 3:**
Consider a single-level cache with an access time of 2.5 ns, a line size of 64 bytes and a hit ratio of \( H = 0.95 \). Main memory uses a block transfer capability that has a first-word (4 bytes) access time of 50 ns and an access time of 5 ns for each word thereafter.

1. What is the access time when there is a cache miss? Assume that the cache waits until the line has been fetched from main memory and then re-executes for a hit.
2. Suppose that increasing the line size to 128 bytes increases \( H \) to 0.97. Does this reduce the average memory access time.
Question 4:
Assume the following performance characteristics on a cache read miss: one clock cycle to send an address to main memory and four clock cycles to access a 32-bit word from main memory and transfer it to the processor and cache.

1. If the cache line size is one word, what is the miss penalty (i.e., additional time required for a read in the event of a read miss)?

2. What is the miss penalty if the cache line size is four words and a multiple, non-burst transfer is executed?

3. What is the miss penalty if the cache line size is four words and a transfer is executed in burst mode, with one clock cycle per word transfer after the initial transfer?

Submission
For submission, you need to submit a document containing answers to all questions. The document must be named properly to identify yourself. Preferably submit the document in pdf format.

Marks Distribution
This coursework will be marked out of 100. Details are:

- Q1: 40
- Q2 – Q4: 20 marks each

Deadline
The deadline is: 9 pm, January 05, 2021.