

Assignment 6

Due: 1:25pm, Wed Nov 2nd, 2022

Note: Make reasonable assumptions where necessary and clearly state them. Feel free to discuss problems with classmates, but the only written material that you may consult while writing your solutions are the textbook and lecture slides/videos. Solutions should be uploaded as a single pdf file on Canvas. **Show your solution steps** so you receive partial credit for incorrect answers and we know you have understood the material. Don't just show us the final answer.

This homework has an automatic penalty-free **1.5 day extension** to accommodate any covid/family-related disruptions. In other words, try to finish your homework by Wednesday 1:25pm to keep up with the lecture content, but if necessary, you may take until **Thursday 11:59pm**.

1. LSQ (30 points)

The table below lists a sequence of loads and stores in the LSQ, when their one/two input operands are made available, and their computed effective addresses. Estimate when the address calculation happens for each ld/st and when each ld/st accesses the data memory. Assume that the processor does *no* memory dependence prediction to speculatively issue loads.

| LD/ST | The register for the address calculation is made available | The register that must be stored into memory is made available | The calculated effective address | Data memory access time |
|-------|--|--|----------------------------------|-------------------------|
| LD | 4 | - | abce | |
| ST | 9 | 3 | abdd | |
| LD | 2 | - | abcd | |
| LD | 5 | - | abdd | |
| ST | 2 | 3 | abdd | |
| LD | 6 | - | abdd | |
| LD | 1 | - | abce | |

2. Memory access times (30 points)

Consider a processor and a program that would have an IPC of 1 with a perfect 1-cycle L1 cache. Assume that each additional cycle for cache/memory access causes program execution time to increase by one cycle. Assume the following MPKIs and latencies for the following caches:

- L1: 16 KB: 1-cycle: 40 MPKI
- L2: 1 MB: 8-cycle: 25 MPKI
- L3: 16 MB: 30-cycle: 10 MPKI
- L4: 64 MB: 60-cycle: 5 MPKI
- Memory: 250 cycles

Estimate the program execution times for the following cache hierarchy configurations. Which cache hierarchy is the best, and (in one sentence) can you reason about why it emerges as the best design point?

1. L1-L2-L3-L4-memory
2. L1-L2-L3-memory
3. L1-L2-L4-memory

3. Cache Organization (20 points)

A 48 MB L3 cache has a 128 byte block (line) size and is 12-way set-associative. How many sets does the cache have? How many bits are used for the offset, index, and tag, assuming that the CPU provides 40-bit addresses? How large is the tag array? (If you do not explain your steps, you will not receive partial credit for an incorrect answer.)

4. Cache Miss Rates (20 points)

For the following access pattern: (i) Indicate if each access is a hit or miss. (ii) What is the hit rate? Assume that the cache has 2 sets and is 2-way set-associative. Assume that block A maps to set 0, B to set 1, C to set 0, D to set 1, E to set 0, F to set 1. Assume an LRU replacement policy.

Does the hit rate improve if you assume a fully-associative cache of the same size, i.e., 1 set and 4 ways? Again, indicate if each access is a hit or a miss.

Access pattern: A B C D E A C E A C E
