Homework 8 – Caching & Virtual Memory

Out: 4.14.21
Due: 4.28.21

1. [Cache]
   Assume a processor executing a program with the following properties:
   - Instructions complete, on average, every 7 cycles, when the memory references are all found in cache.
   - A 2.4GHz clock.
   - The miss time is 40 ns.
   - The data miss rate is 5% and the instruction miss rate is 2%.
   - 30% of the instructions are loads and stores.

   a) How many instructions per second are executed?
   b) How many instructions per second would be executed if there were no cache misses?

2. [Cache]
   Here is a series of address references given as word addresses: 1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17. For the following cache designs, label each reference as a hit or a miss and show the final contents of the cache. Assume the cache is initially empty.
   a) Direct mapped with 16 1-word blocks
   b) Direct mapped with 4 4-word blocks

3. [Cache]
   The following matrix computation code is written in C, where elements within the same row are stored contiguously. Assume each word is a 32-bit integer.

```
for (I=0; I<8; I++)
  for (J=0; J<8000; J++)
    A[I][J]=B[I][0]+A[J][I]
```

   1. How many 32-bit integers can be stored in a 16-byte cache block?
   2. References to which variables exhibit temporal locality?
   3. References to which variables exhibit spatial locality?

   Locality is affected by both the reference order and data layout. The same computation can also be written below in Matlab, which differs from C by storing matrix elements within the same column contiguously in memory.

```
for I=1:8
  for J=1:8000
    A(I,J)=B(I,0)+A(J,I);
  end
end
```

[part 4 was skipped]
5. References to which variables exhibit temporal locality?
6. References to which variables exhibit spatial locality?

4. [Cache]
Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 32-bit memory address references, given as word addresses.

3, 180, 43, 2, 191, 88, 190, 14, 181, 44, 186, 253

1. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with 16 one-word blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.
2. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with two-word blocks and a total size of 8 blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.
3. You are asked to optimize a cache design for the given references. There are three direct-mapped cache designs possible, all with a total of 8 words of data: C1 has 1-word blocks, C2 has 2-word blocks, and C3 has 4-word blocks. In terms of miss rate, which cache design is the best? If the miss stall time is 25 cycles, and C1 has an access time of 2 cycles, C2 take s3 cycles, and C3 takes 5 cycles, which is the best cache design?

5. [Cache – associativity] Optional
Refer to the list of memory accesses from problem 4.
1. Show the final cache content for a 3-way set associative cache with two-word blocks and a total size of 24 words. Use LRU replacement. For each reference identify the index bits, the tag bits, the block offset bits, and if it is a hit or a miss.
2. Show the final cache content for a fully associative cache with one-word blocks and a total size of 8 words. Use LRU replacement. For each reference identify the index bits, the tag bits, and if it is a hit or a miss.

6. [Virtual Memory]
Let the Page Size = 64KB; Address Space = 28 bits; Physical Memory = 20MB. Let the process occupy 256 pages of memory. Let the first ten entries of the page table be in HEX 0x05, 0x99, 0x30, 0x01, 0x11, 0xFF, 0x07, 0x00, 0xAA, 0x20. For the following
virtual addresses, give the physical address. If you don't have enough information, say so.
a) 0xFFFFFF
b) 0x012345
c) 0x054321
d) 0x900000

7. [Virtual Memory]
Virtual memory uses a page table to track the mapping of virtual addresses to physical addresses. This exercise shows how this table must be updated as addresses are accessed. The following data constitutes a stream of virtual addresses as seen on a system. Assume 4 KB pages, a 4-entry fully associative TLS, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number.

Address stream: 4669, 2227, 13916, 34587, 48870, 12608, 49225

TLB:

Valid	Tag	Physical Page Number
1	11	12
1	7	4
1	3	6
0	4	9

Page Table:

Valid	Physical Page or in Disk
1	5
0	Disk
0	Disk
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

1. Given the address stream above, and the initial TLB and page table states provided above, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.
2. Repeat part (1), but this time use 16KB pages instead of 4KB pages. What would be some of the advantages of having a larger page size? What are some of the disadvantages?