CS161 – Fall 2016
Homework Assignment 4

Due: Friday, December 2nd @ 11:59pm

This assignment may be completed in pairs. Please Indicate both names when submitting the assignment and upload in PDF format to iLearn

1. In this exercise we look at memory locality properties of matrix computation. The following 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];
```

a) How many 32-bit integers can be stored in a 16-byte cache block?

b) References to which variables exhibit temporal locality?

c) 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
```

d) How many 16-byte cache blocks are needed to store all 32-bit matrix elements being referenced?

e) References to which variables exhibit temporal locality?

f) References to which variables exhibit spatial locality?

\

2. 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

a) 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.

b) 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.

c) 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 takes 3 cycles, and C3 takes 5 cycles, which is the best cache design?

There are many different design parameters that are important to a cache's overall performance. Below are listed parameters for different direct-mapped cache designs.

Cache Data Size: 32 KiB
Cache Block Size: 2 words
Cache Access Time: 1 cycle

d) Calculate the total number of bits required for the cache listed above, assuming a 32-bit address. Given that total size, find the total size of the closest direct-mapped cache with 16-word blocks of equal size or greater. Explain why the second cache, despite its larger data size, might provide slower performance than the first cache.

e) Generate a series of read requests that have a lower miss rate on a 2 KiB 2-way set associative cache than the cache listed above. Identify one possible solution that would make the cache listed have an equal or lower miss rate than the 2 KiB cache. Discuss the advantages and disadvantages of such a solution.

f) A simple formula to index a direct-mapped cache is: (Block address) modulo (Number of blocks in the cache). Assuming a 32-bit address and 1024 blocks in the cache, consider a different indexing function, specifically (Block address[31:27] XOR Block address[26:22]). Is it possible to use this to index a direct-mapped cache? If so, explain why and discuss any changes that might need to be made to the cache. If it is not possible, explain why.

3. For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache.

Tag	Index	Offset
31–10	9–5	4–0

a) What is the cache block size (in words)?

b) How many entries does the cache have?

c) What is the ratio between total bits required for such a cache implementation over the data storage bits?

Starting from power on, the following byte-addressed cache references are recorded.

Address											
0	4	16	132	232	160	1024	30	140	3100	180	2180

d) How many blocks are replaced?

e) What is the hit ratio?

f) List the final state of the cache, with each valid entry represented as a record of <index, tag, data>.

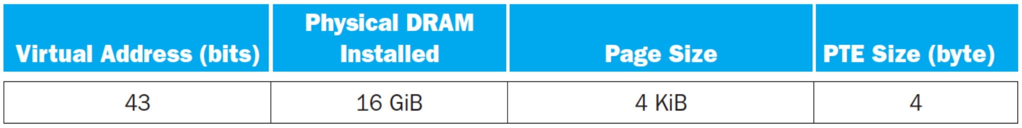
4. This exercise examines the impact of different cache designs, specifically comparing associative caches to the direct-mapped caches. For these exercises, refer to the address stream shown in Question (2).

a) Using the sequence of references from Question (2), show the final cache contents for a three-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 off set bits, and if it is a hit or a miss.

b) Using the references from Question (2), show the final cache contents 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.

c) Using the references from Question (2), what is the miss rate for a fully associative cache with two-word blocks and a total size of 8 words, using LRU replacement? What is the miss rate using MRU (most recently used) replacement? Finally what is the best possible miss rate for this cache, given any replacement policy?

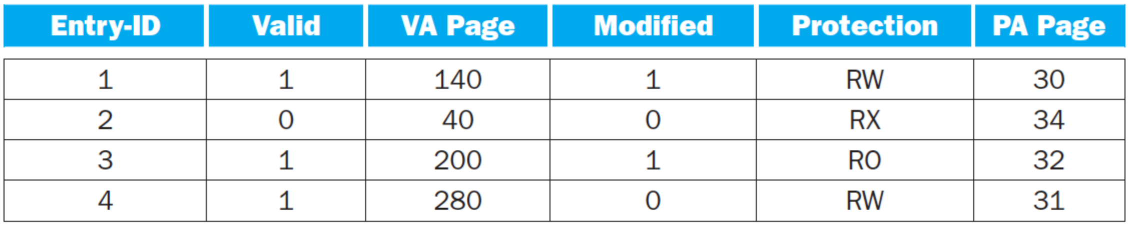
5. In this exercise, we will examine space/time optimizations for page tables. The following list provides parameters of a virtual memory system.



Virtual Address (bits)	Physical DRAM Installed	Page Size	PTE Size (byte)
43	16 GiB	4 KiB	4

a) For a single-level page table, how many page table entries (PTEs) are needed? How much physical memory is needed for storing the page table?

The following table shows the contents of a 4-entry TLB.



Entry-ID	Valid	VA Page	Modified	Protection	PA Page
1	1	140	1	RW	30
2	0	40	0	RX	34
3	1	200	1	RO	32
4	1	280	0	RW	31

b) Under what scenarios would entry 2's valid bit be set to zero?

c) What happens when an instruction writes to VA page 30? When would a software managed TLB be faster than a hardware managed TLB?

d) What happens when an instruction writes to VA page 200?

6. Consider a system with physically-addressed caches, and assume that 40-bit virtual addresses and 32-bit physical addresses are used, and the memory is byte-addressable. Further assume that the cache is 4-way set-associative, the cache line size is 64 Bytes and the total size of the cache is 64 KBytes. Answer the following questions, providing adequate explanations in all cases:

a) What should be the minimum page size in this system to allow for the overlap of the TLB access and the cache access?

b) Repeat part (a) assuming that the cache associativity is increased to 8. Assume that the total cache size and the cache line size remain the same.

c) Assuming that the memory page size in this system is as calculated in your answer to Part (b), compute the total size of the page table in bytes. Assume that a simple linear page table is used and only the page translation information is stored in each entry, with no additional bits.