CMPEN 331 – Computer Organization and Design,
Exam 2 Review Questions

1. We need to calculate the performance of cache design for a byte-addressed memory system. The type of the cache design is a direct-mapped cache (DM) with four blocks, each block holding one four-byte word.

For the following sequences of memory read accesses to the cache, calculate the performance of the cache. Assume that all blocks are invalid initially, and that each address sequence is repeated as shown below. All addresses are given in decimal.

i. Memory Byte Accesses: 0, 4, 0, 4, (repeats). The Miss Rate is:
(a) 0%
(b) 25%
(c) 40%
(d) 50%
(e) 100%
(f) none of the above

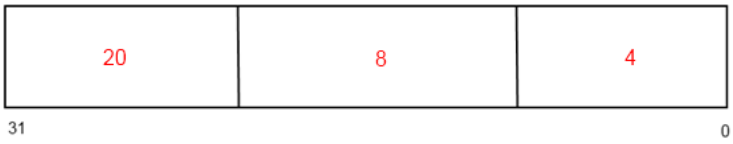
ii. Memory Byte Accesses: 0, 1, 2, 3, 5, 2, 3, 5, 3, 2 (not repeatable) The Miss Rate is:
(a) 0%
(b) 25%
(c) 50%
(d) 75%
(e) 100%
(f) none of the above

iii. Memory Byte Accesses: 0, 0, 4, 0 The Miss Rate is:

(a) 0%
(b) 25%
(c) 50%
(d) 75%
(e) 100%
(f) none of the above

2. For a different data cache, also byte--addressed with 32--bit addresses, with a total data capacity of 4KB and 4x32b word blocks.

i. label the indicated fields as tag, index, and offset, and indicate the number of bits for each.



In addition to data storage, each block of cache requires some additional state storage to enable the hardware to function correctly.

ii. If this cache were write--through, each cache block would require _____bits of storage total.

iii. If this cache were write--back, each cache block would require _____ bits of storage total.

3. Compare the performance of two cache designs for a byte-addressed memory system. The first cache design is a direct-mapped cache (DM) with four blocks, each block holding one four-byte word. The second cache has the same capacity and block size but is fully associative (FA) with a least-recently used replacement policy. **(6 points)**

For the following sequences of memory read accesses to the cache, compare the relative performance of the two caches. Assume that all blocks are invalid initially, and that each address sequence is repeated a large number of times. Ignore compulsory misses when calculating miss rates. All addresses are given in decimal.

Fully associative: allow a given block to go in any cache entry
Compulsory miss: This occurs when a process starts, or restarts, or touches new data
Least-recently used: Choose the one unused for the longest time

iv. Memory Accesses: 0, 4, 0, 4, (repeats). The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	0%	0%
(b)	0%	100%
(c)	100%	0%
(d)	100%	50%
(e)	100%	100%

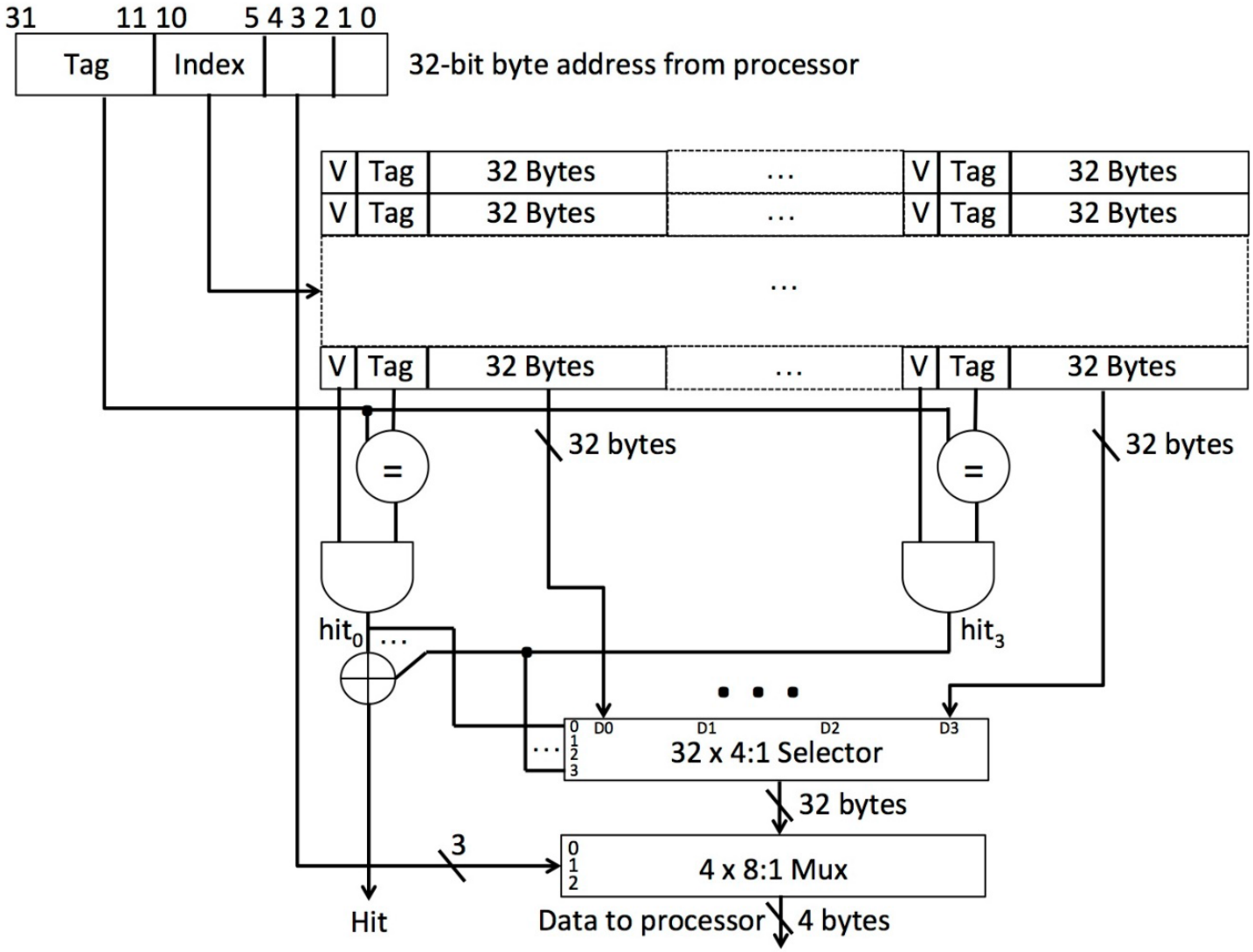
v. Memory Accesses: 0, 4, 8, 12, 16, 0, 4, 8, 12, 16, (repeats) The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	20%	0%
(b)	40%	0%
(c)	20%	20%
(d)	40%	100%
(e)	100%	100%

vi. Memory Accesses: 0, 4, 8, 12, 16, 12, 8, 4, 0, 4, 8, 12, 16, 12, 8, 4, The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	25%	0%
(b)	25%	25%
(c)	50%	0%
(d)	50%	100%
(e)	100%	100%

4. You are given the sketch of a cache design below:



Note: a 4:1 Selector has four control inputs labeled 0, 1, 2, 3 and four data inputs D0, D1, D2, D3. It connects D_i to the output when the ith control signal is true. Assume that at most one control input is true at any time. The selector function is undefined when none of the control inputs are true.

Answer the following questions about the cache above.

i. What is the block size of the cache in bytes?
ii. What is the number of blocks in this cache?
iii. What is the total data capacity of the cache in bytes?
iv. What is the total number of valid bits in the cache?
v. What is the total number of tag bits in the cache?

5. You are given two programs that read in n files (where n is very large), perform some small operation on each file (like counting the occurrence of a word, etc.), and then generate some output files. Listed below are the high--level pseudo--code implementations of each program:

Program A:	**Program B:**
Read file1 Do--something Output out1 Read file2 Do--something Output out2 … (and so on)	Read file1 Read file2 … (read all the files) Do--something Do--something … (do something to all the files) Output out1 Output out2 … (output all the results)

i. Which program would you expect to better utilize a data cache? **(1 point)**
(circle one) A B

i. Which program would you expect to better utilize an instruction cache? **(1 point)**
(circle one) A B

6. Rank the following types of memories in terms of average memory access time: **(2 point)**

Disk storage, Register, Main Memory, L1 Cache, L2 Cache

Fastest , _____ Slowest

Solution

1. We need to calculate the performance of cache design for a byte-addressed memory system. The type of the cache design is a direct-mapped cache (DM) with four blocks, each block holding one four-byte word.

For the following sequences of memory read accesses to the cache, calculate the performance of the cache. Assume that all blocks are invalid initially, and that each address sequence is repeated as shown below. All addresses are given in decimal.

vii. Memory Accesses: 0, 4, 0, 4, (repeats). The Miss Rate is:

(a) 0%
(b) 25%
(c) 40%
(d) 50%
(e) 100%
(f) none of the above

a

viii. Memory Accesses: 0, 1, 2, 3, 5, 2, 3, 5, 3, 2 The Miss Rate is: (not repeatable)

(a) 0%
(b) 25%
(c) 50%
(d) 75%
(e) 100%
(f) none of the above

f

ix. Memory Accesses: 0, 0, 4, 0 The Miss Rate is:

(a) 0%
(b) 25%
(c) 50%
(d) 75%
(e) 100%
(f) none of the above

'c' but it will be 'a' with large number of times

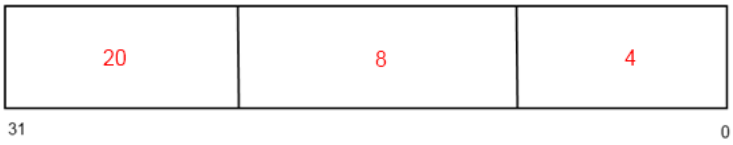
2. For a different data cache, also byte--addressed with 32--bit addresses, with a total data capacity of 4KB and 4x32b word blocks.

iv. label the indicated fields as tag, index, and offset, and indicate the number of bits for each.

We have (4 x 32 /8) = 16 bytes per slot
4096/16 bytes = 256

index = $\log_2 256$ = 8 bits

Byte address = 4 bits since we have 16 bytes per slot.
Tag = 32 – (8+4) = 20 bits



In addition to data storage, each block of cache requires some additional state storage to enable the hardware to function correctly.

v. If this cache were write--through, each cache block would require 149 bits of storage total.
(4*32)(bits / block) + 20 + 1 (validity bit) = 149 bits

vi. If this cache were write--back, each cache block would require 150 bits of storage total.
(4*32) (bits / block) + 20 + 1 (validity bit) +1 (dirty bit because or write back) = 150 bits

3. Compare the performance of two cache designs for a byte-addressed memory system. The first cache design is a direct-mapped cache (DM) with four blocks, each block holding one four-byte word. The second cache has the same capacity and block size but is fully associative (FA) with a least-recently used replacement policy. **(6 points)**

For the following sequences of memory read accesses to the cache, compare the relative performance of the two caches. Assume that all blocks are invalid initially, and that each address sequence is repeated a large number of times. Ignore compulsory misses when calculating miss rates. All addresses are given in decimal.

Fully associative: allow a given block to go in any cache entry
Compulsory miss: This occurs when a process starts, or restarts, or touches new data
Least-recently used: Choose the one unused for the longest time

In Direct-Mapped organization, the block offset for these two accesses is two bits, and we have 2 bits for the index because the cache has 4 blocks. As a result, block offset will be 0 for both of them and index is 0 and 1, respectively for these two accesses, and they will map to the different rows (sets) in the cache. Except for the first two accesses that are cold misses, all the other accesses will be hit. So, miss rate will be 0 percent in the direct-mapped organization.

If the cache is fully-associative, the cache has only one row(set), and as a result, the number of index bits will be 0 and all the accesses will map to the same row. However, we have four blocks in this row and except the first two accesses, all other accesses will be hit because these two cache blocks of address 0 and 4 will not be evicted from the cache and the miss rate will be 0 again.

x. **(2 points)** Memory Accesses: 0, 4, 0, 4, (repeats). The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	0%	0%
(b)	0%	100%

(c)	100%	0%
(d)	100%	50%
(e)	100%	100%

a

xi. **(2 points)** Memory Accesses: 0, 4, 8, 12, 16, 0, 4, 8, 12, 16, (repeats) The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	20%	0%
(b)	40%	0%
(c)	20%	20%
(d)	40%	100%
(e)	100%	100%

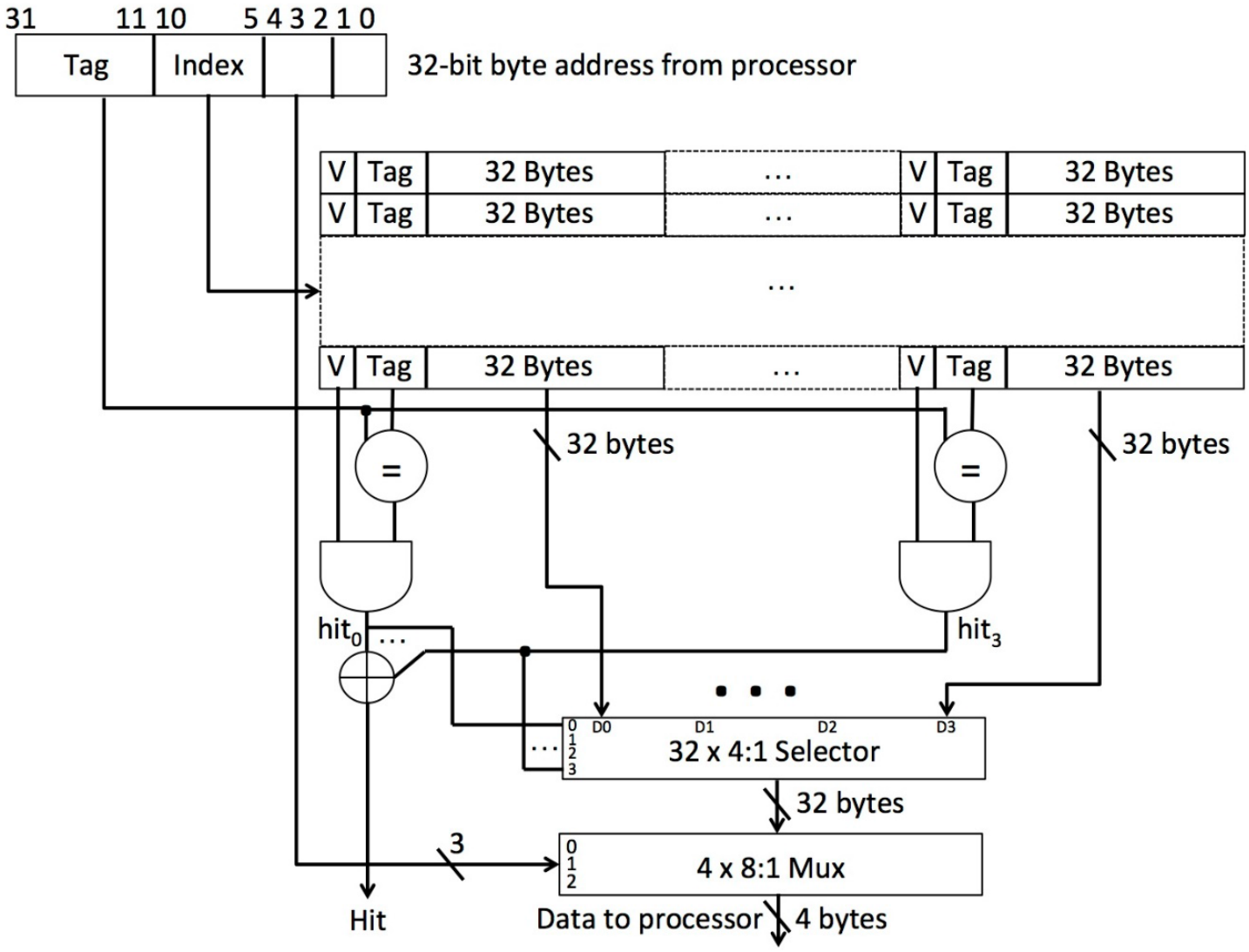
d

xii. **(2 points)** Memory Accesses: 0, 4, 8, 12, 16, 12, 8, 4, 0, 4, 8, 12, 16, 12, 8, 4, The Miss Rate is:

	DM Miss Rate	FA Miss Rate
(a)	25%	0%
(b)	25%	25%
(c)	50%	0%
(d)	50%	100%
(e)	100%	100%

b

4. You are given the sketch of a cache design below:



Note: a 4:1 Selector has four control inputs labeled 0, 1, 2, 3 and four data inputs D0, D1, D2, D3. It connects D_i to the output when the ith control signal is true. Assume that at most one control input is true at any time. The selector function is undefined when none of the control inputs are true.

Answer the following questions about the cache above.

vi. What is the block size of the cache in bytes?
32

vii. What is the number of blocks in this cache?

$2^6 \times 4 = 256$

viii. What is the total data capacity of the cache in bytes?

2^6 x 32 x 4 = 8192 Bytes

ix. What is the total number of valid bits in the cache?

256

x. What is the total number of tag bits in the cache?

256 * 21 = 5376

5. You are given two programs that read in n files (where n is very large), perform some small operation on each file (like counting the occurrence of a word, etc.), and then generate some output files. Listed below are the high--level pseudo--code implementations of each program:

Program A:	**Program B:**
Read file1 Do--something Output out1 Read file2 Do--something Output out2 … (and so on)	Read file1 Read file2 … (read all the files) Do--something Do--something … (do something to all the files) Output out1 Output out2 … (output all the results)

ii. Which program would you expect to better utilize a data cache?
(circle one) A B

A

ii. Which program would you expect to better utilize an instruction cache?
(circle one) A B

B

6. Rank the following types of memories in terms of average memory access time:

Disk storage, Register, Main Memory, L1 Cache, L2 Cache

Fastest Register , L1 Cache, L2 Cache, Main Memory, Disk storage Slowest