IN1006 Systems Architecture (PRD1 A 2022/23)

Started on	Thursday, 8 December 2022, 5:47 PM
State	Finished
Completed on	Thursday, 8 December 2022, 6:00 PM
Time taken	12 mins 44 secs
Grade	10.00 out of 10.00 (100%)

Question 1

Correct

Mark 1.00 out of 1.00

Which of the following is a pair of non-volatile memories?

Select one:

- a. DRAM and Flash
- b. EPROM and DRAM
- c. DRAM and SRAM
- d. ROM and Flash ✔
- e. SRAM and EEPROM

The volatile memories in the question are SRAM, VRAM and DRAM, The only option that does not contain any of them is the answer.

The correct answer is: ROM and Flash

Question 2

Correct

Mark 1.00 out of 1.00

Which of the following statements about the different types of memory that can be found in a computer is correct?

- a. A computer can operate without a RAM.
- b. A computer can operate without a ROM.
- c. A computer cannot operate without virtual memory.
- d. A computer must always have a full memory hierarchy with L1 cache, L2 cache, RAM and secondary memory.
- e. A computer can operate without a cache and secondary memory. ✔

Your answer is correct.

The correct answer is:
A computer can operate without a cache and secondary memory.

Question 3

Correct

Mark 1.00 out of 1.00

Assume a computer having a main memory of 2^{32} bytes and a direct mapped cache of 1024 blocks each of which containing 32 bytes. To which cache block the main memory address $000063FA_{16}$ map?

Select one:

- a. Block 1024
- b. Block 573
- c. Block 1
- d. Block 799 ✔
- e. Block 64
- f. Don't know/no answer

As the computer has a memory with 2^{32} bytes, it will need 32 bits to identify the addresses of all of them. Thus, the total size of the tag, block and offset (TBO) fields is 32.

As the cache has 1024 blocks, it can be indexed with 10 bits ($1024 = 2^{10}$), so the size of the block (B) field is 10 bits.

As each block has 32 bytes, so it can be indexed with 5 bits ($32 = 2^5$) or, equivalently, the offset (O) field must be 5 bits long.

Thus , the tag (T) field is: TBO - B - O = 32 - 10 - 5 = 17.

Consequently the memory address format will be

TAG (32nd to 16th) BLOCK (15th to 6th bit) OFFSET (5th to 1st bit)

The hexadecimal address $000063FA_{16}$ corresponds to the binary address:

0000 0000 0000 0000 0110 0011 1111 1010

So the block part in this address consists of the bold bits below:

0000 0000 0000 0000 0110 0011 1111 1010

And if we convert $110\ 0011\ 111_{bin}$ to decimal, we get 799_{dec} So the given address will map to block 799 in cache.

The correct answer is: Block 799

Question 4

Correct

Mark 1.00 out of 1.00

Assume a computer having a main memory of 2^{32} bytes and a direct mapped cache of 512 blocks each of which containing 64 bytes. To which cache block the main memory address $000063FA_{16}$ map?

Select one:

- a. Block 399 ✔
- b. Block 512
- c. Block 201
- d. Don't know/no answer
- e. Block 124
- f. Block 58

As the computer has a memory with 2^{32} bytes, it will need 32 bits to identify the addresses of all of them. Thus, the total size of the tag, block and offset (TBO) fields is 32.

As the cache has 512 blocks, it can be indexed with 9 bits ($512 = 2^9$), so the size of the block (B) field is 9 bits.

As each block has 64 bytes, so it can be indexed with 64 bits ($64 = 2^6$) or, equivalently, the offset (O) field must be 6 bits long.

Thus , the tag (T) field is: TBO - B - O = 32 - 9 - 6 = 17.

Consequently the memory address format will be

TAG (32 nd to 16 th) BLOCK (15 th to 7 th bit) OFFSET (6 th to 1 st bit)

The hexadecimal address $000063FA_{16}$ corresponds to the binary address:

0000 0000 0000 0000 0110 0011 1111 1010

So the block part in this address consists of the bold bits below:

0000 0000 0000 0000 0110 0011 1111 1010

And if we convert 110001111_{bin} to decimal, we get 399_{dec} So the given address will map to block 399 in cache.

The correct answer is: Block 399

Question 5

Correct

Mark 1.00 out of 1.00

Suppose we have a byte-addressable computer which has a memory of 4,294,967,296 addresses, a cache with 64 blocks where each block contains 16 bytes. What is the size of the block (B), offset (O) and the tag (T) fields in the memory addresses used by this computer.

Select one:

- a. T= 16 , O= 8 and B = 8
- b. T= 28 , O= 2 and B = 2
- c. T= 22 , O= 4 and B = 6 ✔
- d. Don't know/no answer
- e. T= 8 , O= 16 and B = 8
- f. T= 24 , O= 4 and B = 4

As the computer has a memory with 4,294,967,296 bytes, it will need 32 bits to identify the addresses of all of them (4,294,967,296 = 2^{32}). Thus, the total size of the tag, block and offset (TBO) fields is 32.

As the cache has 64 blocks, it can be indexed with 6 bits (64 = 2^6). So the size of the block field (B) is 6.

As each block has 16 bytes, it can be indexed with 4 bits (16 = 2^4). So the size of the offset field (B) is 4.

Thus , the tag (T) field is: TBO - B - O = 32 - 6 - 4 = 22.

The correct answer is: T= 22 , O= 4 and B = 6

Question 6

Correct

Mark 1.00 out of 1.00

Which of the following *best* describes why a memory hierarchy is needed?

Select one:

- a. SRAM is cheaper than DRAM per unit of memory.
- b. Don't know/no answer
- c. It is useful to be able to exceed physical memory limits.
- d. Processor speeds have increased faster than SRAM speeds.
- e. DRAM is cheaper than SRAM per unit of memory.
- f. Processor speeds have increased faster than memory speeds and there are different technologies of memory with different speeds and costs. ✔

The need for a memory hierarchy is driven by the increasing gap between the speed of the processor and the speed of main memory (though it has other useful applications, e.g. in allowing physical memory) and the different technologies of memory.

The correct answer is: Processor speeds have increased faster than memory speeds and there are different technologies of memory with different speeds and costs.

Question 7

Correct

Mark 1.00 out of 1.00

Assume a computer having a cache whose block size is 1024 bytes. Which of the following programs would lead the most effective use of the sequential locality?

Select one:

- a. A program with 800 instructions executed within a loop that can be executed several times processing an array with 200 data points. ✔
- b. A program that reads 2000 data points from a file and prints them to the computer terminal.
- c. A program with that computes the sum of 9999 numbers stored in a file.
- d. A program with 900 instructions that processes an array with 300 data points.
- e. A program with 50 instructions that reads and transforms data in a file which can grow dynamically.
- f. Do know the answer.

Sequential locality refers to the common feature of programs that instructions and data tend to be accessed sequentially.

As the block size of the computer is 1024 bytes only the program with the 800 instructions processing the array of 200 data points would be possible to fit in one block and thus execute as many times as required without the need to transfer further data and/or instructions from the main memory to the cache. The existence of a loop in this program would not affect sequential locality either since the jump from the last to the first instruction of the loop in the program would not cause any transfers to and from the main memory because all instructions of the program can fit in the cache and would be loaded at the start of the program's execution.

All other options given may lead to data/instructions transfers between cache and the main memory thus violating sequential locality.

The correct answer is: A program with 800 instructions executed within a loop that can be executed several times processing an array with 200 data points.

Question 8

Correct

Mark 1.00 out of 1.00

Suppose we have a byte-addressable computer which has a memory of 4,294,967,296 addresses, a cache with 32 blocks where each block contains 32 bytes. What is the size of the block (B), offset (O) and the tag (T) fields in the memory addresses used by this computer.

Select one:

- a. T= 28 , O= 2 and B = 2
- b. T= 16 , O= 8 and B = 8
- c. T= 22 , O= 5 and B = 5 ✔
- d. Don't know/no answer
- e. T= 8 , O= 16 and B = 8
- f. T= 24 , O= 4 and B = 4

As the computer has a memory with 4,294,967,296 bytes, it will need 32 bits to identify the addresses of all of them (4,294,967,296 = 2^{32}). Thus, the total size of the tag, block and offset (TBO) fields is 32.

As the cache has 32 blocks, it can be indexed with 5 bits (32 = 2^5), so the size of the block (B) field is 5 bits.

As each block has 32 bytes, so it can be indexed with 5 bits (32 = 2^5) or, equivalently, the offset (O) field must be 5 bits long.

Thus , the tag (T) field is: TBO - B - O = 32 - 5 - 5 = 22.

The correct answer is: T= 22 , O= 5 and B = 5

Question 9

Correct

Mark 1.00 out of 1.00

Suppose we have a byte-addressable computer which has a memory of 4,294,967,296 addresses, a cache with 128 blocks where each block contains 32 bytes. What is the size of the block (B), offset (O) and the tag (T) fields in the memory addresses used by this computer.

Select one:

- a. T= 8 , O= 16 and B = 8
- b. T= 24 , O= 4 and B = 4
- c. T= 16 , O= 8 and B = 8
- d. T= 20 , O= 5 and B = 7 ✔
- e. Don't know/no answer
- f. T= 28 , O= 2 and B = 2

As the computer has a memory with 4,294,967,296 bytes, it will need 32 bits to identify the addresses of all of them (4,294,967,296 = 2^{32}). Thus, the total size of the tag, block and offset (TBO) fields is 32.

As the cache has 128 blocks, it can be indexed with 7 bits (128 = 2^7), so the size of the block (B) field is 7 bits.

As each block has 32 bytes, so it can be indexed with 5 bits (32 = 2^5) or, equivalently, the offset (O) field must be 5 bits long.

Thus , the tag (T) field is: TBO - B - O = 32 - 7 - 5 = 20.

The correct answer is: T= 20 , O= 5 and B = 7

Question 10

Correct

Mark 1.00 out of 1.00

Suppose we have a byte-addressable computer using direct mapping with 16-bit main memory addresses and 16 blocks of cache. If each block contains 8 bytes, determine the size of the tag field.

Select one:

- a. 9 ✔
- b. 8
- c. Don't know/no answer
- d. 4
- e. 16
- f. 5

The total size of the tag, block and offset fields is 16.
16 blocks of cache can be indexed with 4 bits (32 = 2^4), so block needs 4 bits.

Each block has 8 bytes so it can be indexed with 3 bits (8=2^3); thus 3 bits are needed for the offset to determine the address of each byte within a block.

Finally, the tag field is: 16 - 4 - 3 = 9.

The correct answer is: 9

◄ Quiz 5 _ Weekly Assessed Quiz 2022

Jump to...

Quiz 7 _ Weekly Assessed Quiz 2022 ►

Quiz navigation

1 2 3 4 5 6 7 8 9 10

Show one page at a time

Finish review