1. (20 pts) A byte-addressable computer has a small data cache capable of holding eight

32-bit words. Each cache block consists of one 32-bit word. When a given program is

executed, the processor reads data from the following sequence of hex addresses:

Pass 1 200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4 (green is the hits)

Pass 2 200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4

Pass 3 200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4

Pass 4 200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4

This pattern is repeated four times.

(a) Show the contents of the cache at the end of each pass through this loop if a

direct-mapped cache is used. Compute the hit rate for this example. Assume that

the cache is initially empty.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Block | Pass1 | Pass2 | Pass3 | Pass4 |
| 0 | 200 | 200 | 200 | 200 |
| 1 | 204 | 204 | 204 | 204 |
| 2 | 208->24C | 24C->208->24C | 24C->208->24C | 24C->208->24C |
| 3 | 20C | 20C | 20C | 20C |
| 4 | 2F4 | 2F4 | 2F4 | 2F4 |
| 5 | 2F0 | 2F0 | 2F0 | 2F0 |
| 6 | 218 | 218 | 218 | 218 |
| 7 | 21C | 21C | 21C | 21C |

First pass in a different table. This repeats for the next three but slightly different

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| address | index | hit/miss | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 200 | 0 | miss | 200 |  |  |  |  |  |  |  |
| 204 | 1 | miss | 200 | 204 |  |  |  |  |  |  |
| 208 | 2 | miss | 200 | 204 | 208 |  |  |  |  |  |
| 20C | 3 | miss | 200 | 204 | 208 | 20C |  |  |  |  |
| 2F4 | 4 | miss | 200 | 204 | 208 | 20C | 2F4 |  |  |  |
| 2F0 | 5 | miss | 200 | 204 | 208 | 20C | 2F4 | 2F0 |  |  |
| 200 | 0 | hit | 200 | 204 | 208 | 20C | 2F4 | 2F0 |  |  |
| 204 | 1 | hit | 200 | 204 | 208 | 20C | 2F4 | 2F0 |  |  |
| 218 | 6 | miss | 200 | 204 | 208 | 20C | 2F4 | 2F0 | 218 |  |
| 21C | 7 | miss | 200 | 204 | 208 | 20C | 2F4 | 2F0 | 218 | 21C |
| 24C | 2 | miss | 200 | 204 | 24C | 20C | 2F4 | 2F0 | 218 | 21C |
| 2F4 | 4 | hit | 200 | 204 | 24C | 20C | 2F4 | 2F0 | 218 | 21C |

With that there are 9 misses on the first pass and two misses on the second, third, and fourth. This makes miss rate 15/48 = 0.3125 \* 100 = 31.25% and hit rate 33/48 = 0.6875 \*100 = 68.75%

(b) Repeated part (a) for a 2-way associative-mapped cache that uses the LRU

replacement algorithm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Block | Pass1 | Pass2 | Pass3 | Pass4 |
| 0 | ->200 | 200 | 200 | 200 |
| 1 | ->204 | 204 | 204 | 204 |
| 2 | ->208->24C | 24C->21C | 21C->218 | 218->2F0 |
| 3 | ->20C | 20C->208->24C | 24C->21C | 21C->218 |
| 4 | ->2F4 | 2F4 | 2F4 | 2F4 |
| 5 | ->2F0 | 2F0->20C | 20C->208->24C | 24C->21C |
| 6 | ->218 | 218->2F0 | 2F0->20C | 20C->208->24C |
| 7 | ->21C | 21C->218 | 218->2F0 | 2F0->20C |
|  | 9 misses | 6 misses | 6 misses | 6 misses |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Block | Pass1 | Pass2 | Pass3 | Pass4 |
| 0 | 7->200 | -19>200 | -31>200 | -42>200 |
| 1 | 8->204 | -20>204 | -32>204 | -43>204 |
| 2 | 3->208-11>24C | 24C-22>21C | 21C-33>218 | 218-41>2F0 |
| 3 | 4->20C | 20C-15>208-23>24C | 24C-34>21C | 21C-44>218 |
| 4 | 12->2F4 | -24>2F4 | -35>2F4 | -40>2F4 |
| 5 | 6->2F0 | 2F0-16>20C | 20C-27>208-34>24C | 24C-45>21C |
| 6 | 9->218 | 218-18>2F0 | 2F0-28>20C | 20C-38>208-46>24C |
| 7 | 10->21C | 21C-21>218 | 218-30>2F0 | 2F0-39>20C |
|  | 9 misses | 6 misses | 6 misses | 6 misses |

The miss rate 27/48 = 56.25% The hit rate was 100-56.25 = 43.75%.

(c) Repeated part (a) for a four-way set-associative cache.

200, 204, 208, 20C, 2F4, 2F0, 200, 204, 218, 21C, 24C, 2F4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Block | Pass1 | Pass2 | Pass3 | Pass4 |
| 0 | ->200 | 200 | 200 | 200 |
| 1 | ->208 | 208 | 208 | 208 |
| 2 | ->2F0 | 2F0 | 2F0 | 2F0 |
| 3 | ->218 | 218 | 218 | 218 |
| 4 | ->204 | 204 | 204 | 204 |
| 5 | ->20C->24C | 24C->21C | 21C->20C->24C | 24C->21C |
| 6 | ->2F4 | 2F4 | 2F4 | 2F4 |
| 7 | ->21C | 21C->20C->24C | 24C->21C | 21C->20C->24C |
|  | 9 misses | 3 misses | 3 misses | 3 misses |

The miss rate 18/48 = 37.5% The hit rate was 100-37.5 = 62.5%.

2. (20 pts) In many computers the cache block size is in the range of 32 to 128 bytes.

What would be the main advantages and disadvantages of making the size of cache

blocks larger or smaller?

A larger cache block size is good for spatial locality. It will decrease compulsory misses which is a good thing and will increase the hit rate. It could also get rid of collision conflicts in memory locations mapped to the same cache location.

A disadvantage would be it might also increase the miss penalty which could make missing waste more time. For very large block sizes it may increase miss rate due to pollution as well.

3. (20 pts) Consider the following page-address trace generated by a two-level cache-

main-memory scheme that uses demand paging and has a cache capacity of four pages.

1 6 4 5 1 4 3 2 1 2 1 4 6 7 4 1 3 1 3 1 7

Assume the cache initially contains pages 1, 2, 3 and 4. Which of the page-replacement

policies FIFO or LRU is more suitable in this case? Show your calculations, and give a

short intuitive justification of your answer.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIFO: | 1 | 2 | 3 | 4 |
| 1 | 1 | 2 | 3 | 4 |
| 6: f | 6 | 2 | 3 | 4 |
| 4 | 1 | 2 | 3 | 4 |
| 5: f | 1 | 5 | 3 | 4 |
| 1 | 1 | 5 | 3 | 4 |
| 4 | 1 | 5 | 3 | 4 |
| 3 | 1 | 5 | 3 | 4 |
| 2: f | 1 | 5 | 2 | 4 |
| 1 | 1 | 5 | 2 | 4 |
| 2 | 1 | 5 | 2 | 4 |
| 1 | 1 | 5 | 2 | 4 |
| 4 | 1 | 5 | 2 | 4 |
| 6: f | 1 | 5 | 2 | 6 |
| 7: f | 7 | 5 | 2 | 6 |
| 4: f | 7 | 4 | 2 | 6 |
| 1: f | 7 | 4 | 1 | 6 |
| 3: f | 7 | 4 | 1 | 3 |
| 1 | 7 | 4 | 1 | 3 |
| 3 | 7 | 4 | 1 | 3 |
| 1 | 7 | 4 | 1 | 3 |
| 7 | 7 | 4 | 1 | 3 |

FIFO had 8 page faults

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| LRU: | 1 | 2 | 3 | 4 |
| 1 | **1** | 2 | 3 | 4 |
| 6: f | 1 | **6** | 3 | 4 |
| 4 | 1 | 6 | 3 | **4** |
| 5: f | 1 | 6 | **5** | 4 |
| 1 | **1** | 6 | 5 | 4 |
| 4 | 1 | 6 | 5 | **4** |
| 3: f | 1 | 3 | 5 | 4 |
| 2: f | 1 | 3 | 2 | 4 |
| 1 | 1 | 3 | 2 | 4 |
| 2 | 1 | 3 | 2 | 4 |
| 1 | 1 | 3 | 2 | 4 |
| 4 | 1 | 3 | 2 | 4 |
| 6: f | 1 | 6 | 2 | 4 |
| 7: f | 1 | 6 | 7 | 4 |
| 4 | 1 | 6 | 7 | 4 |
| 1 | 1 | 6 | 7 | 4 |
| 3: f | 1 | 3 | 7 | 4 |
| 1 | 1 | 3 | 7 | 4 |
| 3 | 1 | 3 | 7 | 4 |
| 1 | 1 | 3 | 7 | 4 |
| 7 | 1 | 3 | 7 | 4 |

LRU had 7 page faults

From the two methods LRU had the least page faults by 1 so it is the better page-replacement policy in this example.

4. (20 pts) Consider a cache with 32 blocks and a block size of 16 bytes. What block

number does byte address 1200 map to? Note that 1200 is a decimal number, not a

hexadecimal number.

1200/16 = 75

75%32 = 11

Cache block number 11

5. (20 pts) You have learned techniques to reduce miss penalties in class. Similarly, it is

also important to reduce the hit time of memory access. List (at least) three approaches

that can be used to reduce the hit time.

Lower associativity, add L1 and L2 caches.

Reduce miss penalty, can do this using faster RAM or adding a write buffer between cache and memory.

Design the memory system to support caches, either using one-word wide memory organization, wide memory organization or interleaved memory architecture.

Use aliasing to hit faster.

Use pipelined writes, this would help because writing takes longer than reading for hits because of the tag check before the data is written. Can be sped up by pipelining this write process.

v