CSE3666

Homework 8

Due Date: By the end of Friday, 4/29/2022.

Total Points: 100

Enter your answers in HuskyCT.

Round answers to the nearest hundredths *if necessary*. For example, enter 0.5 for a half, 0.33 for a third, and 0.67 for two thirds. For percentages, enter numbers only. For examples, enter 1.23 for 1.234% and 1.24 for 1.235%.

Assume 32-bit addresses in all problems.

1. Cache organization.

Consider a direct-mapped cache of 256 KiB. The block size is 64 bytes. Each block has two status bits. Find out the following numbers.

a) The number of cache blocks

b) The number of bits in the offset

c) The number of bits in the cache index

d) The number of bits in the tag field

e) The number of bits for each block (including tag, data, and status bits)

f) The total number of bits in the cache

g) Percentage of bits that are not used for storing data

2. Cache organization.

Consider a direct-mapped cache. The bits in an address are used in the following way when accessing the cache. The number bits in cache index is 8 and the number of bits in the block offset is 5.

One status bit for each block indicates if a block is valid.

a) The number of bytes in a block

b) The number of blocks in the cache

c) The number of bits in the tag field

d) The size of cache in KiB

e) The total number of bits in the cache

3. Cache operation.

A sequence of memory references is provided in the spreadsheet. Assume the higher 20 bits of the addresses are the same (so you can ignore them in this problem) and the cache is empty at the beginning. For each address, find out the tag, the cache index, whether the access is a miss or a hit (0 for miss and 1 for hit), and lowest and highest addresses in the same block. The answers for the first memory reference are provided as an example.

In addition, answer the following questions.

a) What is the block size of the cache in bytes?

b) What is the size of the cache in bytes?

c) How many misses are in total?

4. Cache performance.

Suppose a computer system has a 5-stage RISC-V processor, and separate data and instruction caches. The hit time of both caches is one cycle. The miss penalty of both caches is 80 cycles.

When the processor runs an application, the miss rate of the data cache is 10% and the miss rate of the instruction cache is 5%.

35% of the instruction executed in an application are memory accesses.

The CPI of the application is 1.6 without memory stalls.

A system that has a perfect memory does not have stalls due to memory accesses.

a) What is the average memory access time of the data cache?

b) What is the average memory access time of the instruction cache?

c) What is the CPI overhead of the application due to data memory accesses?

d) What is the CPI overhead of the application due to instruction memory accesses?

e) What is the average CPI with memory stalls?

f) What is the speedup if the system has a perfect memory system?

5. Application.

Applications that have "streaming" workloads bring in large amounts of data but do not reuse much of it. Consider a streaming application that mainly use an array of 1M (2^{20}) half words as working data set and accesses the elements in the array sequentially, starting from 0.

```
short int D[1024*1024]; // a short int is a half word
// access pattern is D[0], D[1], D[2], and so on
```

Assume a 64KiB direct-mapped cache with a 32-byte block is in the system. The starting address of the array is a multiple of 256. Ignore other memory references.

a) When the application accesses D[0], it is loaded into cache. What are the elements in D that are also loaded into cache?

b) After the miss for accessing D[0], how many hits are there before the next miss?

c) Which element in D, when it is accessed, will evict D[0] from the cache?

d) What is the miss rate of this application accessing the working data set?

e) What is the miss rate of this application assessing the working data set if the cache block size is of 64 bytes?