15-213: Introduction to Computer Systems
Written Assignment #5

This written homework covers Cache, Design, and Debugging

Directions

Complete the question(s) on the following pages with single paragraph answers. These questions are not meant to be particularly long! Once you are done, submit this assignment on Canvas.

Below is an example question and answer.

Q: Please describe benefits of two's-complement signed integers versus other approaches.

A: Other representations of signed integers (ones-complement and sign-and-magnitude) have two representations of zero (+0 and -0), which makes testing for a zero result more difficult. Also, addition and subtraction of two's complement signed numbers are done exactly the same as addition and subtraction of unsigned numbers (with wraparound on overflow), which means a CPU can use the same hardware and machine instructions for both.

Grading

Each assignment will be graded in two parts:

1. Does this work indicate any effort? (e.g. it's not copied from a homework for another class or from the book)
2. Three peers will provide short, constructive feedback.

Due Date

This assignment is due on March 15 by 11:59 PM Pittsburgh time (currently UTC-4). Remember to convert this time to the timezone you currently reside in.

Question 1

Consider a situation as described below:

L1 cache:	5 cycles/access, 96% hit rate
L2 cache:	10 cycles/access, 97% hit rate
L3 cache:	25 cycles/access, 98% hit rate
Main memory:	100 cycles/access

If you could wave a magic wand and increase the hit rate of any level of cache by 1%, for example making the L1 go from 96% to 97%,, which cache would you improve and why? You don't have to do math, but your answer should show intuition that could be demonstrated by setting up and solving the equations.

Solution: L1's hit rate

Increasing L1's hit rate implies lesser access to higher levels of cache, which dramatically increases the average access cyclesbecause

Average_access_cycles = L1_hitrate*L1_cycles + L1_missrate*L2_average_access_cycles

L2_average_access_cycles = L2_hitrate*(L2_cycles + L1_cycles) + L2_missrate*(L3_avaerage_access_cycles)

Increasing anything other than L1 is going to get multiplied by the miss rate of the previous level which is a very small number and thus leads to no significant increase in the Average access cycles.

Given a cache of size 8192 bytes with 8 sets and 4 lines per set, answer the following questions (you can assume that the addresses are shorts, aka 16 bits) :

a) How many bits represent the set?
b) How many bytes are there per block?
c) How many bits are necessary to represent the block?
d) Given the address 0x338B what are the tag bits? The set bits? The block bits?

Identifies all values correctly:

3
256
8
Tag = 00110, set = 011, block = 10001011