Name(Pinyin):
Email(Prefix):

Computer Architecture Homework 6

2020 Spring Apr. 21

Instructions:

Homework 6 is due in May. 6, covers the content of caches and float-points, please refer to the lecture slides. You can print it out and write on it, and scan it into a pdf, or you can take photos or write Latex if you want, just remember: you must create a **PDF** and upload to the **Gradescope**, please assign the questions properly on Gradescope, otherwise you will lose 25% of points.

Question Set 1. Direct Mapped Cache

[30 points] In a 16-bit byte-addresses machine, the clock frequency is 2GHz. We have a cache with properties as follows:

1. Cache size is 4 KiB;
2. Block size is 8 Bytes;
3. Cache hit time is 2 cycles;
4. Cache miss penalty is 200 cycles;

1-A. What the width of each field of following address bit assignment:

TAG:4	Set index:9	Block offset:3

Please show the procedure that your solutions derive from.

Answer 6pt + Analysis 4pt

1-B. We will access the data of addresses as follows. Fill in the blanks. It is about the index, tag (in decimal) and whether there is a hit or miss. If there is a miss, then give what type is the miss (either compulsory or replace). (Here we define replace as either conflict or capacity that causes a miss.)

Addresses (serially access)	Tag/Index	Hit, Compulsory or Replace
0x0000	0/0	Compulsory
0x0004	0/0	Hit
0x0008	0/1	Compulsory
0x000c	0/1	Hit
0x1000	1/0	Replace
0x1004	1/0	Hit
0x1008	1/1	Replace
0x100c	1/1	Hit
0x0000	0/0	Replace
0x0004	0/0	Hit

1-C. Calculations. (Step-by-step, worth 50% pts)

1-C-i: Miss rate: (4 pt.)

5/10 = 50%

1-C-ii: AMAT (ns): (3 pt.)

Period = 1/clock frequency = 1/2GHz = 0.5ns

Hit time = 1ns

Miss time = 100ns

AMAT = hit time + miss time * miss rate = 1ns+ (100*0.5) = 51ns

1-C-iii: AMAT if we don't have this cache (ns): (3 pt.)

AMAT = miss time * 1 = 100ns

Question Set 2. Two-Way Set Associative Cache

From QS 1. We change the block size to 4 Bytes and implemented a two-way set associative cache. The parameters are shown as follows:

1. Cache size is 4 KiB;
2. 16-bit byte-addresses machine;
3. Block size is 4 Bytes;

2-A. What is the width of each field of following address bit assignment? :

TAG: 5	Set index: 9	Block offset: 2

Please show the procedure that your solutions derive from.

Answer 6pt + Analysis 4pt

2-B. We will access the data of the addresses as follows. Fill in the blanks. It is about the index, tag (in decimal) and whether there is a hit or miss. If there is a miss, then give what type is the miss (either compulsory or replace). Use LRU policy to decide which block to replace. (Here we define replace as either conflict or capacity that causes a miss.)

Addresses (serially access)	Tag/Index	Hit, Compulsory or Replace
0x0000	0/0	Compulsory
0x0004	0/1	Compulsory
0x0008	0/2	Compulsory
0x000c	0/3	Compulsory
0x1000	2/0	Compulsory
0x1004	2/1	Compulsory
0x0000	0/0	Hit
0x2000	4/0	Replace
0x0000	0/0	Hit
0x1000	2/0	Replace

2-C. Calculations.

2-C-i. Miss rate: (5 pt.)

8/10 = 80%

2-C-ii. Assume the new cache miss time is 300 cycles and hit time is 3 cycles. Calculate the AMAT in ns. Round to the nearest tenth. (5 pt.)

Period = 1/clock frequency = 1/2GHz = 0.5ns

Hit time = 1.5ns

Miss time = 150ns

AMAT = 1.5 + 150*0.8 = 121.5ns

Question Set 3. Floating Point Numbers

We consider the IEEE 32-bit floating point representation except with a 6-bit exponent (bias of 31) and a denorm implicit exponent of -30.

3-A. Convert `-95.2` to that form. In hexadecimal.

0xcaf99999

3-B. Convert floating point number `0x49480000` from binary to decimal, specify infinities as `+inf` and `-inf`, and not a number as `NaN`.

52.5(decimal)

3-C. What is the next smallest positive number larger than 2 that can be represented completely? (Not every number can be represented perfectly using floating point. For example, $\frac{1}{3}$ can only be approximated with floating point). Please explain why.

Increment the number by the smallest amount possible. We increment significand by 1 in the rightmost location 0b00…01, so (1+2^-25)*2 = 2 + 2^-24

3-D. What's the smallest positive value it can represent that is not a denorm? Leave your answer as a power of 2. Please explain why.

2^-30

This number is not a denorm, so exponent can not be zero and it is 0b000001 when we want the smallest value. We make significands all zeros to get the smallest: 1.0*2^(1-31)

3-E. What's the smallest positive value it can represent? Leave your answer as a power of 2. Please explain why.

2^-30*2^-25 = 2^-55

This number is denorm with Exponent all zeros and Significands 0b000…01 which is 2^-25. So the total is 2^-55. Note that denorm has an implicit exponent of -30 which is different from norm.