Introduction to Computer Architecture
CSE-3666

Practice Final Exam-- closed notes, closed book

Materials Allowed: Scrap paper, Calculator

Read problems carefully and budget your time wisely. Make your handwriting legible.

Do not use pseudoinstructions in your answers.

Question 1: (14 Points) _____

Question 2: (10 Points) _____

Question 3: (10 Points) _____

Question 4: (10 Points) _____

Question 5: (15 Points) _____

Question 6: (15 Points) _____

Question 7: (16 Points) _____

Question 8: (10 Points) _____

Total _____

I pledge my honor that I have not violated and will not violate the exam policy of this course and the Student Conduct Code during this examination.

Signature: _____

Name: _____

PeoplesoftID: _____

Note:

- **You need to show your work to get a full credit. you will receive partial credit if your final answers are not correct.**

1. (1) True of False.
   a. A pipelined datapath must have separate instruction memory and data memory since instruction format is different from data format. **(F)**
   b. In the RISC-V 5-stage pipeline, not all RAW data hazards can be eliminated by forwarding. **(T)**
   c. Pipelining reduces the latency of individual instructions **(F)**
   d. From generation to generation, the number of the cache blocks inside a CPU increases exponentially as Moore's law predicted. **(F)**

   (2) Select the correct answer(s) for the following questions. (One or multiple answers may be correct.)

   a) How many memory addresses are there with a 32-bit address line?

   A. 2^{32}
   B. 32
   C. 32^2
   D. This depends on how much memory has been installed in the computer.

   b) Which one of the following technology trend is NOT true:

   A. The CPU clock is running faster.
   B. The CPU performance keeps on improving.
   C. The CPU supply voltage is rising higher.
   D. The CPU power consumption is going up.

   c) Given the Million Instructions per Second (MIPS) rate of a machine, we need _____ to calculate its average CPI.

   A. Clock frequency
   B. CPU Time
   C. Instruction count
   D. Both A and B

   Hint: MIPS = Clock rate/(CPI * 10^6)

2. Short answers:
   a) Write the assembly language statement(s) that will reverse the values of each bit in register t0 and put the result in register s1.

   Nor s1, t0,0

   b) What decimal number does the bit pattern 0×0C000000 represent if it is a floating point number? Use the IEEE 754 standard.

   Answer: 0x0C000000 = 0000 1100 0000 0000 0000 0000 0000 0000
   The first bit is 0 which will represent (-1)0. The next 8 bits represent the exponent, which in this case is 24. The rest of the bits are 0s so this bit pattern represents $1 \times 1.0 \times 2^{(24-127)} = 1.0 \times 2^{-103}$

c) Using three RISC-V instructions to multiply the value in S0 by 10 and save the result in S1.

<<1
<<3
+

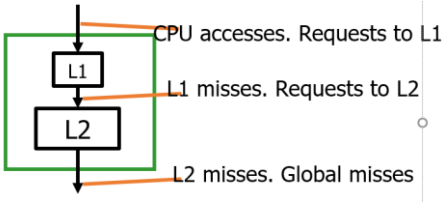
d) why do we need to save the instruction in a pipeline register multiple times?

We need to save the instruction in a register multiple times because each pipeline stage needs to receive the right control signals for the instruction currently in that stage.

e) Write two cache optimizations that are used to improve the miss rate.

Larger block, Set associativity

3. A processor with a CPIideal of 2, a 100 cycle miss penalty, 36% load/store instr's, and 2% I cache and 4% D cache miss rates.



Memory-stall cycles = 0.02 × 100 + 0.36 × 0.04 × 100 = 3.44
CPI_{stalls} = 2 + 3.44 = **5.44**

Suppose we add a unified L2 cache. L2 access time: 25 cycles, for both hit and miss
Miss rate of the entire cache: 0.5%. What is the new CPI_{stalls}? What is speedup?

With L2 cache.
Memory-stall cycles = 0.02×25 + 0.36×0.04×25 + (1+0.36)×0.005×100 = 1.54
New CPI_{stalls} = 2 + 1.54 = 3.54
Speedup = 5.44 / 3.54 = 1.55

4. Translate the lines of C code to RISC-V code that accomplishes the same thing. Assume a and b are stored in s0 and s1, respectively. **Do not use pseudoinstructions in your answers**.

```
Int a=5,b=10;
If (a+a==b){
        a=0;
   }else{
        b=a-1;
   }
```

```
    addi s0, x0, 5
    addi s1, x0, 10
    add  t0, s0, s0
    bne  t0, s1, else
    xor  s0, x0, x0
    jal  x0,  exit
else:
    addi s1, s0, -1
exit:
```

5. Compilers can have a profound impact on the performance of an application. Assume that for a program, compiler A results in a dynamic instruction count of 1.0E9 and has an execution time of 1.1s, while compiler B results in a dynamic instruction count of 1.2E9 and an execution time of 1.5s.

   a.) Find the average CPI for each program given that the processor has a clock cycle time of 1 ns.

CPI units = cycles per instruction
Clock cycle time units = seconds/cycle
Execution time units = seconds

$$CPI = \frac{Execution\ time}{clock\ cycle \times instructions} = \frac{cycles \times seconds}{seconds \times instructions} = \frac{cycles}{instruction}$$

For compiler A:

$$\frac{1.1\ seconds}{(1.0 \times 10^9\ instructions) \times (1.0 \times 10^{-9})} = \mathbf{1.1\ CPI}$$

For compiler B:

$$\frac{1.5\ seconds}{(1.2 \times 10^9\ instructions) \times (1.0 \times 10^{-9})} = \mathbf{1.25\ CPI}$$

   b.) Assume the compiled programs run on two different processors. If the execution times on the two processors are the same, how much faster is the clock of the processor running compiler A's code versus the clock of the processor running compiler B's code?

Since the execution times are the same, we can utilize the equation

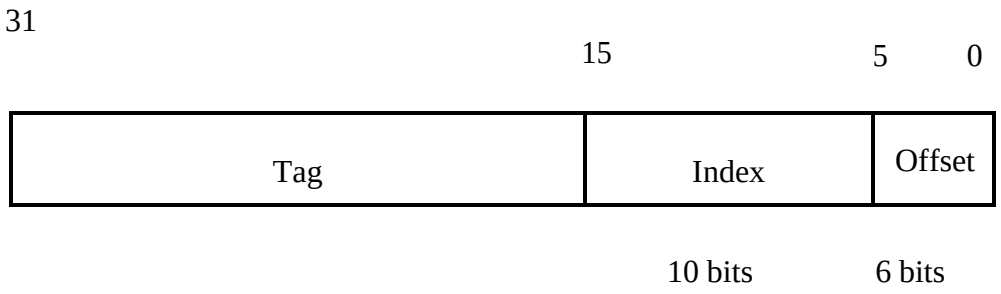
$$Execution\ time = \frac{seconds \times cycles \times instructions}{cycles \times instructions} = clock\ cycle \times CPI \times instructions$$

$$Execution\ time_A = (clock\ cycle_A) \times (1.1\ CPI_A) \times (1.0 \times 10^9\ instructions)$$
$$Execution\ time_B = (clock\ cycle_B) \times (1.25\ CPI_B) \times (1.2 \times 10^9\ instructions)$$

$$Execution\ time_A = Execution\ time_B$$
$$(clock\ cycle_A) \times (1.1\ CPI_A) \times (1.0 \times 10^9\ instructions) =$$
$$= (clock\ cycle_B) \times (1.25\ CPI_B) \times (1.2 \times 10^9\ instructions)$$
$$\frac{clock\ cycle_A}{clock\ cycle_B} = \frac{(1.25\ CPI_B) \times (1.2 \times 10^9\ instructions)}{(1.1\ CPI_A) \times (1.0 \times 10^9\ instructions)} = 1.36$$

So clock A is 1.36 times faster than clock B

6. 32 bit address of a direct-mapped data cache that has 6 bits for the offset field and 10 bits for the index field as shown bellow



(a) How many words per block does the cache have?

M+2 =6 → m=4

2^4 =16

16 words per block

(b) How many bytes per block does the cache have?

16*4=64→ 2^6 bytes per block

(c) what is the data size of the cache in KB?

bytes = #block * #bytes per block

block =2^10

#bytes =2^10 * 2^6= 64 KB

Hint:
If asked: the total number of bits in a direct-mapped cache is,
then:
2^n × (block size + tag field size + valid field size)

If asked: Cache (data) size in KB, then: # bytes = #block * #bytes per block

7. Consider the following sequence of instructions:

OR x5, x6, x7

OR x6, x5, x1

OR x5, x5, x6

Also, assume the following cycle times for each of the options related to forwarding:

Without Forwarding	With Full Forwarding	With ALU-ALU Forwarding Only
250ps	300ps	290ps

(a) Assume there is no forwarding in this pipelined processor. Indicate hazards (what types of hazard on which register from which line).

OR x5, x6, x7

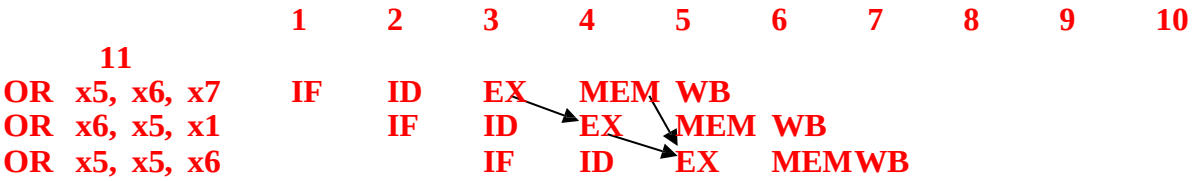
OR x6, x5, x1 -- RAW hazard on x5 from line 1

OR x5, x5, x6 -- RAW hazards on x5 from line 1 and on x6 from line 2

(b) If the hardware has no hazard detection unit, the hazards have to be eliminated by adding *NOP* instructions explicitly into the instruction sequence. Modify the instruction sequence to eliminate hazard.

OR x5, x6,
x7
NOP
NOP
OR x6, x5,
x1
NOP
NOP
OR x5, x5, x6

(c) If the hardware has full forwarding unit. Draw the pipeline execution diagram. Indicate data forwarding on the diagram. In each clock cycle, indicate the value of forward A and forward B signals.



(d) What is the total execution time of this instruction sequence with full forwarding? What is the speedup achieved by adding full forwarding?

-With full forwarding, execution time = 7 cycles * 300ps = 2100ps speed up = 2750/2100 = 1.31 times

8. A processor has a Miss Rate of 5% for both instruction cache and data cache and a Hit Time of 0.2 ns. Assume that main memory accesses take 40 ns and that memory accesses are 40% of all instructions. (use appropriate units to get full credit)

a) Assuming that the hit time determines the cycle time for the processor, what is its respective clock rate?

Solution:

$$Clock\ Rate\ (GHz) = \frac{1}{seconds\ (ns)}$$

So, for P1

$$clock\ rate\ \ = \frac{1}{0.2\ ns} = 5\ GHz$$

b) What is the average memory access time for the processor?

Solution:

$$AMAT = (hit\ time) + (miss\ rate) \times (miss\ time)$$
$$AMAT\ \ = 0.2\ ns + (0.05 \times 40ns) = 2.2\ ns$$

c) If the processor has a CPI of 1 without any memory stalls, determine and state how much faster a processor would run with a perfect cache that never missed.

Solution:

CPI = CPI-ideal + Memory Stall cycles per instruction
= CPI-ideal +frequency of memory access * miss rate * miss penalty
Miss penalty cycles= 40 ns/0.2 ns=200 cycles
I_Memory stall cycles per instruction = 1 x 0.05 x 200 = 10
D_memory stall cycles per instruction = 0.4 x 0.05 x 200 = 4
Total CPI= 1 + 10 + 4 = 15

CPUTime -real/CPUTime-ideal=15/1 → 15 times faster